



Controlled query evaluation in description logics through consistent query answering [☆]

Gianluca Cima ^a, Domenico Lembo ^a, Riccardo Rosati ^{a,*}, Domenico Fabio Savo ^b

^a Sapienza Università di Roma, Italy

^b Università degli Studi di Bergamo, Italy

ARTICLE INFO

Keywords:

Description logics

Ontologies

Confidentiality preservation

Query answering

Data complexity

ABSTRACT

Controlled Query Evaluation (CQE) is a framework for the protection of confidential data, where a *policy* given in terms of logic formulae indicates which information must be kept private. Functions called *censors* filter query answering so that no answers are returned that may lead a user to infer data protected by the policy. The preferred censors, called *optimal* censors, are the ones that conceal only what is necessary, thus maximizing the returned answers. Typically, given a policy over a data or knowledge base, several optimal censors exist.

Our research on CQE is based on the following intuition: confidential data are those that violate the logical assertions specifying the policy, and thus censoring them in query answering is similar to processing queries in the presence of inconsistent data as studied in Consistent Query Answering (CQA). In this paper, we investigate the relationship between CQE and CQA in the context of Description Logic ontologies. We borrow the idea from CQA that query answering is a form of skeptical reasoning that takes into account all possible optimal censors. This approach leads to a revised notion of CQE, which allows us to avoid making an arbitrary choice on the censor to be selected, as done by previous research on the topic.

We then study the data complexity of query answering in our CQE framework, for conjunctive queries issued over ontologies specified in the popular Description Logics $DL\text{-}Lite_{\mathcal{R}}$ and \mathcal{EL}_{\perp} . In our analysis, we consider some variants of the censor language, which is the language used by the censor to enforce the policy. Whereas the problem is in general intractable for simple censor languages, we show that for $DL\text{-}Lite_{\mathcal{R}}$ ontologies it is first-order rewritable, and thus in AC^0 in data complexity, for the most expressive censor language we propose.

1. Introduction

Preserving confidentiality in information systems involves devising query answering mechanisms that protect sensible data from unauthorized access. This problem has been extensively studied in the fields of databases and Artificial Intelligence, specifically in knowledge representation, with various works exploring it in the context of Description Logic (DL) ontologies (e.g. in [2–4]). In this paper, we refer to a logic-based approach to confidentiality preservation known as Controlled Query Evaluation (CQE), originally

[☆] This paper is an extended version of [1].

* Corresponding author.

E-mail addresses: cima@diag.uniroma1.it (G. Cima), lembo@diag.uniroma1.it (D. Lembo), rosati@diag.uniroma1.it (R. Rosati), domenicofabio.savo@unibg.it (D.F. Savo).

<https://doi.org/10.1016/j.artint.2024.104176>

Received 2 December 2022; Received in revised form 19 June 2024; Accepted 24 June 2024

Available online 2 July 2024

0004-3702/© 2024 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>).

introduced for propositional databases [5,6] and more recently studied over ontologies [7,8]. In CQE a declarative *policy*, typically expressed as a set of logic formulae, specifies the information that is considered sensitive, and a function called *sensor* modifies the query answers so that users do not get data that may lead them to infer secrets protected by the policy. Sensors that operate according to some minimality criterion are called *optimal*, and are usually the preferred ones, for their ability to maximize answers to queries. For example, if a policy prohibits the disclosure of both facts $\text{hasName}(01,\text{John})$ and $\text{hasSalary}(01,2000)$, an optimal sensor would reveal just one of these facts (and retain the other confidential), rather than concealing both pieces of information. It is clear that in general several (typically incomparable) optimal sensors exist (for instance, in the previous example, there are two such sensors).

Under a purely logical perspective, data protected by the policy in CQE are those that result inconsistent with respect to the logical formulae specifying the policy. Thus, if we interpret the policy as a set of integrity constraints over the information schema, censoring confidential data in query answering is similar to processing queries in the presence of data that contradict these constraints. In this paper we move from the above observation to study, in the context of DL ontologies, the relationship existing between CQE and Consistent Query Answering (CQA), a declarative framework aiming at guaranteeing meaningful query answering over inconsistent logical theories, where otherwise inference becomes trivial, according to the “ex falso quod libet” principle of classical first-order logic [9–14].

In our investigation, we start from the CQE framework proposed in [8]. This framework generalizes the one proposed in [15], which in turn extends the one for propositional open databases given in [16] to DL ontologies (see also Section 8). The paper [8] carries CQE over DLs through an elegant and simple declarative formalization based on the classical semantics of first-order logic, which makes use of a definition of sensor effective enough to protect sensitive information from the ability of a user of making (classical) inferences over the ontology and the answers that the system returns to queries, even in the presence of expressive formulae in the policy. These characteristics led us to devise a quite natural correspondence with the problem of CQA. More precisely, we define a CQE instance as a triple consisting of a DL TBox (i.e. the intensional component of the ontology), an ABox (i.e. the extensional component of the ontology) and a policy, expressed as a set of conjunctive queries (CQs), whose answers must never be inferred from the system by a user asking queries over the TBox. We then consider the prototypical CQA framework based on the so-called ABox-Repair (AR) semantics [14,17,11]. In this latter framework, given an ontology composed of a TBox \mathcal{T} and an ABox \mathcal{A} , possibly contradicting \mathcal{T} , a repair is an inclusion-maximal subset of \mathcal{A} that is consistent with \mathcal{T} , and processing a query q amounts to computing the answers to q inferred by all (consistent) ontologies constituted by \mathcal{T} and one such repair.

We thus start considering sensors that return inclusion-maximal subsets of the ABox that satisfy the policy seen as a set of constraints.

Example 1. Let us formalize the previous example. Assume for simplicity an empty TBox and consider the policy $\exists x, y, z(\text{hasName}(x, y) \wedge \text{hasSalary}(x, z))$, saying that for a certain individual x no both name and salary can be disclosed. Let us turn it into a constraint, i.e. $\forall x, y, z(\text{hasName}(x, y) \wedge \text{hasSalary}(x, z) \rightarrow \perp)$. Given the ABox $\mathcal{A} = \{\text{hasName}(01,\text{John}), \text{hasSalary}(01,2000)\}$, the two possible optimal sensors produce the two sets $\{\text{hasName}(01,\text{John})\}$ and $\{\text{hasSalary}(01,2000)\}$, which correspond to two repairs of \mathcal{A} with respect to the TBox consisting of the above constraint. \square

We note that, for more complex scenarios, e.g. with non-empty TBox, the form of sensor we have considered above may result quite restrictive, since it returns only subsets of the ABox. We solve this problem through a parameterization of sensors with respect to a language \mathcal{L}_c , i.e. a set of sentences in some fragment of first-order logic. Intuitively, under this perspective, an (optimal) sensor is a function returning a (maximal) subset C of the sentences in \mathcal{L}_c implied by the TBox and the ABox of the CQE instance, such that C satisfies the policy. We call \mathcal{L}_c the *sensor language*.

We then borrow the idea from CQA that query answering is a form of skeptical reasoning that takes into account all possible sensors, that is, all possible sets of formulae they return, as CQA reasons over all possible repairs. We remark that this form of query answering does not make a specific choice on the (optimal) sensor to select for preserving confidentiality, which might be discretionary or unfair in the absence of metadata suggesting a selection criterion. Our approach is thus different from the one in [8], as well as from other previous works on CQE, where the focus is on the construction of a single sensor, even though an idea similar to ours has been also previously discussed in [15].

Once defined this comprehensive (and novel) framework for CQE, we study its general properties, consider sensor languages of increasing expressiveness, and investigate the relationship with CQA, providing a general property that establishes the conditions under which CQE can be reduced to CQA. We then characterize the data complexity of answering conjunctive queries (CQs) under the various sensor languages we consider, for CQE instances whose TBox is expressed in the popular DLs *DL-Lite_R* [18] and \mathcal{EL}_{\perp} [19], which are, respectively, the logical counterparts of OWL 2 QL and OWL 2 EL, two tractable profiles of OWL 2 [20]. More precisely, we provide data complexity results for the cases when: (i) \mathcal{L}_c is the ABox of the CQE instance, i.e. the sensor can enforce the policy only by selecting facts in the ABox, as in Example 1 (ABox sensor language); (ii) \mathcal{L}_c coincides with the set of facts expressed over the signature of the TBox and ABox of the CQE instance (GA sensor language); (iii) \mathcal{L}_c is the language of Boolean CQs (BCQs) expressed over the signature of the TBox and ABox of the CQE instance and whose length is at most k , for every integer $k \geq 1$ (CQ_k sensor language); and (iv) \mathcal{L}_c is as at point (iii), but with no limits on the maximum length of BCQs (CQ sensor language). Some of our findings follow from the correspondence between CQA and CQE, whereas for the cases in which the CQE problem does not have a CQA counterpart we devise tailored techniques. The complexity results proved in this paper are shown in Fig. 1. Note that besides entailment of BCQs, we also consider the special cases of entailment of ground atoms, a classical reasoning task in DL called *instance checking*, and entailment of *purely existential BCQs* (BCQ_∃), i.e. BCQs not admitting constants in their atoms. We also point out that all

\mathcal{L}_c	$DL\text{-}Lite_{\mathcal{R}}$			$\mathcal{E}\mathcal{L}_{\perp}$		
	Instance checking	BCQ_{\exists} entailment	BCQ entailment	Instance checking	BCQ_{\exists} entailment	BCQ entailment
ABox	= coNP [4, 5]	= coNP [4, 5]	= coNP [4, 5]	= coNP [4, 5]	= coNP [4, 5]	= coNP [4, 5]
GA	$\leq AC^0$ [9]	= coNP [6, 8]	= coNP [6, 8]	= coNP [6, 7]	= coNP [6, 7]	= coNP [6, 7]
CQ_k †	$\leq AC^0$ [13]	= coNP [10, 12]	= coNP [10, 12]	= coNP [10, 11]	= coNP [10, 11]	= coNP [10, 11]
CQ	$\leq AC^0$ [14]	$\leq AC^0$ [14]	$\leq AC^0$ [14]	$\leq coNP$ [16] $\geq PTIME$ ‡	= PTIME[15]	$\leq coNP$ [16] $\geq PTIME$ ‡

Fig. 1. Data complexity of query answering in CQE: corresponding theorems are given between squared brackets. † : The results in this row hold for every fixed integer k such that $k \geq 1$. ‡: The PTIME lower bounds directly follow from [21, Theorem 4.3].

results on entailment of BCQs can be extended to arbitrary (i.e. non-Boolean) CQs in the standard way, and thus also hold for open CQs.

Interestingly, the complexity of entailment never increases when moving from less expressive censor languages to more expressive ones, and decreases in some cases, for both $DL\text{-}Lite_{\mathcal{R}}$ and $\mathcal{E}\mathcal{L}_{\perp}$. In particular, BCQ entailment for $DL\text{-}Lite_{\mathcal{R}}$ is intractable for all censor languages but **CQ**, the most expressive language we consider for censors. For the case of $\mathcal{E}\mathcal{L}_{\perp}$, we prove a similar behavior for BCQ_{\exists} entailment, which is PTIME-complete when $\mathcal{L}_c = \mathbf{CQ}$ and coNP-complete for all the other less expressive censor languages. For BCQ entailment (and instance checking) when $\mathcal{L}_c = \mathbf{CQ}$ we could not establish exact complexity, leaving it open whether membership in coNP can be further refined. Beyond their theoretical connotation, tractability results are interesting for practical applicability. In particular, we believe that the $DL\text{-}Lite_{\mathcal{R}}$ case is very relevant for real-world applications, since it is characterized by an ontology language suited for data-intensive scenarios and ontology-mediated query answering, as well as expressive policy and censor languages, providing a setting that enables a designer great expressiveness for declaratively protecting data. It is worthwhile remarking that our results show that data complexity of CQE for $DL\text{-}Lite_{\mathcal{R}}$ under **CQ** censors is the same as standard query answering in this logic.

The rest of the paper is organized as follows. In Section 2 we provide some preliminaries on first-order logic and Description Logics. In Section 3 we introduce our CQE framework and show its general properties. In Section 4 we investigate the relationship between CQA and CQE. We then focus on CQE instances whose TBox is expressed in the DLs $DL\text{-}Lite_{\mathcal{R}}$ and $\mathcal{E}\mathcal{L}_{\perp}$. For these cases, we study data complexity of conjunctive query answering (instance checking, entailment of BCQs and of BCQ_{\exists} s) under the restricted censor languages ABox and GA in Section 5, under CQ_k censor language in Section 6, and under **CQ** censor language (i.e. the most expressive language we consider for censors), in Section 7. We finally discuss some related work in Section 8 and then conclude the paper in Section 9.

This paper is an extended version of [1].

2. Preliminaries

We make use of standard notions of function-free first-order (FO) logic, and, in particular, we consider DLs that are decidable fragments of FO. As customary, we focus on DLs that use only unary and binary predicates, called (atomic) concepts and roles, respectively [22].

We assume to have the pairwise disjoint countably infinite sets Σ_O , Σ_I , and Σ_V , for (unary and binary) predicates, constants (also known as individuals), and variables, respectively. Σ_O is in turn partitioned in two pairwise disjoint sets Σ_C and Σ_R for atomic concepts and atomic roles, respectively.

We use **FO** to indicate the language of all FO sentences over Σ_O , Σ_I , and Σ_V . Every language considered in this paper is a subset of **FO**. Given a language $\mathcal{L} \subseteq \mathbf{FO}$, a *theory* in \mathcal{L} is a set of sentences in \mathcal{L} . Given a theory \mathcal{K} , the set of predicates and constants occurring in \mathcal{K} is called the *signature* of \mathcal{K} and it is denoted by $sig(\mathcal{K})$. Moreover, given a language $\mathcal{L} \subseteq \mathbf{FO}$, we denote by $\mathcal{L}(\mathcal{K})$ the subset of formulae in \mathcal{L} over $sig(\mathcal{K}) \cup \Sigma_V$.

A *Boolean conjunctive query* (BCQ) q is a sentence in **FO** of the form $q = \exists \vec{x} cq(\vec{x})$, where $cq(\vec{x}) = \alpha_1(\vec{x}_1) \wedge \dots \wedge \alpha_n(\vec{x}_n)$, with $n \geq 1$ and \vec{x} being a sequence of variables from Σ_V such that $\vec{x} = \bigcup_{i=1}^n \vec{x}_i$ ¹ and, for each $i = 1, \dots, n$, $\alpha_i(\vec{x}_i)$ is an atom such that $\alpha_i \in \Sigma_O$ and each of its arguments belongs to $\Sigma_I \cup \vec{x}$. The *length* of q is the number of its atoms, denoted by $length(q)$. Furthermore, we say that q is a BCQ_{\exists} if it does not mention any constants.

In the following, we denote by **CQ** the language of BCQs, by CQ_{\exists} the language of BCQ_{\exists} s, by CQ_k the language of BCQs from **CQ** whose maximum length is k , and by **GA** the language of single-atom queries with no variables, i.e. *ground atoms* or *facts*. Notice that $CQ_{\exists} \subset \mathbf{CQ}$, $\mathbf{GA} \subset CQ_k \subset CQ_{k+1} \subset \mathbf{CQ}$, for every integer $k \geq 1$, CQ_{\exists} and **GA** are incomparable languages, and CQ_{\exists} and CQ_k are incomparable languages, for every integer $k \geq 1$.

A DL ontology \mathcal{O} is a finite theory $\mathcal{T} \cup \mathcal{A}$, where \mathcal{T} is the *TBox* (i.e. “Terminological Box”) and \mathcal{A} is the *ABox*, (i.e. “Assertional Box”), that is, finite sets of assertions (i.e. sentences) specifying intensional and extensional knowledge, respectively.

Different DLs allow for different kinds of TBox and/or ABox assertions. In this paper, an ABox is always a finite set of ground atoms, i.e. atoms of the form $A(a)$, $P(a, b)$, where $A \in \Sigma_C$, $P \in \Sigma_R$, and $a, b \in \Sigma_I$. We mainly consider the DLs $DL\text{-}Lite_{\mathcal{R}}$ [18] and $\mathcal{E}\mathcal{L}_{\perp}$. This latter extends $\mathcal{E}\mathcal{L}$ [19] with the empty concept \perp . Such DLs are the logical counterpart of the OWL 2 profiles OWL 2 QL and OWL 2 EL, respectively [20].

¹ With a little abuse of notation we treat here sequences of variables as sets.

$DL\text{-Lite}_{\mathcal{R}}$ expressions are constructed according to the following syntax:

$$B := A \mid \exists R \quad R := P \mid P^{-}$$

where R is called *basic role*, which can be an atomic role $P \in \Sigma_R$, or its inverse P^{-} , B is called *basic concept*, which can be an atomic concept $A \in \Sigma_C$, or a concept of the form $\exists R$, called *existential restriction*, denoting the set of objects occurring as first argument of a basic role R .

A $DL\text{-Lite}_{\mathcal{R}}$ TBox is a finite set of assertions of the form

$$\begin{array}{ll} B_1 \sqsubseteq B_2 & B_1 \sqsubseteq \neg B_2 \\ R_1 \sqsubseteq R_2 & R_1 \sqsubseteq \neg R_2 \end{array}$$

Assertions of the first row above are called *concept inclusions*, whereas those of the second row are called *role inclusions*. Moreover, assertions of the first column are also called *positive inclusions*, whereas those in the second column are called *negative inclusions*.

\mathcal{EL}_{\perp} expressions are constructed according to the following syntax:

$$C := A \mid \exists P \mid \exists P.C \mid C_1 \sqcap C_2 \mid \perp \mid \top$$

where C is called *general concept*, which can be an atomic concept $A \in \Sigma_C$, an existential restriction of the form $\exists P$, with $P \in \Sigma_R$, a concept of the form $\exists P.C$, with $P \in \Sigma_R$, called *qualified existential restriction* and denoting the set of objects that the atomic role P relates to some instance of the general concept C , a concept of the form $C_1 \sqcap C_2$, i.e. a conjunction of two general concepts, \perp , i.e. the *empty concept*, or \top , i.e. the *top concept*. Note that $\exists P$ is equivalent to $\exists P.\top$, and thus it can be considered syntactic sugar, which we however prefer to have in the \mathcal{EL}_{\perp} syntax to simplify comparison with $DL\text{-Lite}_{\mathcal{R}}$ syntax.

An \mathcal{EL}_{\perp} TBox is a finite set of assertions of the form

$$C_1 \sqsubseteq C_2$$

called *general concept inclusions*.

Besides $DL\text{-Lite}_{\mathcal{R}}$ and \mathcal{EL}_{\perp} assertions, we also consider *denial assertions* (or simply *denials*) over atomic concepts and roles, i.e. sentences of the form

$$\forall \vec{x}(cq(\vec{x}) \rightarrow \perp)$$

where $cq(\vec{x})$ is such that $\exists \vec{x}(cq(\vec{x}))$ is a BCQ. The *length* of the denial is the length of such a query. We will use denials to specify the policy in CQE. In our technical treatment, we will also consider the DL $DL\text{-Lite}_{\mathcal{R},den}$ [14], which extends $DL\text{-Lite}_{\mathcal{R}}$ with denials.

Example 2. Suppose we want to use an ontology to model the domain of a pharmacy, i.e. its customers, medicines, and so on. A TBox \mathcal{T} in $DL\text{-Lite}_{\mathcal{R}}$ for this purpose might be as follows:

$$\mathcal{T} = \{ \begin{array}{l} \exists buy \sqsubseteq \text{Customer}, \\ \exists buy^{-} \sqsubseteq \text{Medicine}, \\ \text{Medicine} \sqsubseteq \exists \text{treats}, \\ \exists \text{treats}^{-} \sqsubseteq \text{Condition} \end{array} \}$$

From top to bottom, the above TBox assertions specify that: who buys something is a customer; what is bought is a medicine; each medicine is used in at least a treatment (cf. role *treats*) the subject of a treatment is always a medical condition.

Let us now consider the language \mathcal{EL}_{\perp} . The expressive power of this DL is incomparable with that of $DL\text{-Lite}_{\mathcal{R}}$. For instance, the second and fourth assertions of \mathcal{T} cannot be specified in \mathcal{EL}_{\perp} , which instead allows a designer to use constructs not available in $DL\text{-Lite}_{\mathcal{R}}$. The following TBox \mathcal{T}' is an example of \mathcal{EL}_{\perp} TBox (incomparable with \mathcal{T}):

$$\mathcal{T}' = \{ \begin{array}{l} \exists buy \sqsubseteq \text{Customer}, \\ \text{Medicine} \sqsubseteq \exists \text{treats}, \\ \text{Customer} \sqsubseteq \exists buy.\text{Medicine}, \\ \exists \text{treats}.\text{Condition} \sqsubseteq \text{Medicine} \end{array} \}$$

\mathcal{T}' specifies that anyone who buys something is a customer, each medicine is used in at least a treatment, and that a customer must buy at least one medicine. It also models the fact that what treats a condition is a medicine.

An ABox suitable for both TBoxes given above could be the following, which states that *bob* bought some *insulin* that treats *diabetes*.

$$\mathcal{A} = \{ buy(bob, insulin), \text{treats}(insulin, diabetes) \}. \quad \square$$

The semantics of a theory \mathcal{K} in **FO** is given in terms of FO interpretations $I = \langle \Delta^I, \cdot^I \rangle$, where Δ^I is the interpretation domain, i.e. a non-empty set of objects, and \cdot^I is the interpretation function, which assigns to each unary predicate $A \in sig(\mathcal{K})$ a subset $A^I \subseteq \Delta^I$, to each binary predicate $P \in sig(\mathcal{K})$ a subset $P^I \subseteq \Delta^I \times \Delta^I$, and to each constant c occurring in $sig(\mathcal{K})$ an object $c^I \in \Delta^I$. An interpretation I is a *model* of \mathcal{K} if I satisfies all the sentences in \mathcal{K} (i.e. all such sentences evaluate to true in I). We denote by

$Mod(\mathcal{K})$ the set of models of \mathcal{K} . A theory \mathcal{K} is consistent if it has at least one model, i.e. if $Mod(\mathcal{K}) \neq \emptyset$, inconsistent otherwise, and \mathcal{K} entails an FO sentence $\phi \in \mathbf{FO}(\mathcal{K})$, denoted $\mathcal{K} \models \phi$, if ϕ is satisfied by every $I \in Mod(\mathcal{K})$. From now on, when we consider entailment of an FO sentence ϕ from a theory \mathcal{K} we always assume that $\phi \in \mathbf{FO}(\mathcal{K})$, and thus we omit to explicitly say it. When $\phi \in \mathbf{GA}$, the above entailment is also called *instance checking*. For non-entailment, we use the notation $\not\models$. As usual, we may also use $I \models \mathcal{K}$ (resp. $I \models \phi$, with ϕ a sentence) to indicate that the interpretation I is a model of \mathcal{K} (resp. satisfies the sentence ϕ). Analogously, $I \not\models \mathcal{K}$ (resp. $I \not\models \phi$) denotes the fact that I is not a model of \mathcal{K} (resp. I does not satisfy ϕ). Moreover, given an ABox \mathcal{A} , we denote with $I_{\mathcal{A}}$ the interpretation “isomorphic” to \mathcal{A} , i.e., $I_{\mathcal{A}} = \langle \Delta^{I_{\mathcal{A}}}, \cdot^{I_{\mathcal{A}}} \rangle$ where $\Delta^{I_{\mathcal{A}}}$ is the set of all the constants appearing in \mathcal{A} , and $\cdot^{I_{\mathcal{A}}}$ is such that $a^{I_{\mathcal{A}}} = a$ for each constant a appearing in \mathcal{A} , $a^{I_{\mathcal{A}}} \in C^{I_{\mathcal{A}}}$ for each $C(a) \in \mathcal{A}$, and $(a^{I_{\mathcal{A}}}, b^{I_{\mathcal{A}}}) \in P^{I_{\mathcal{A}}}$ for each $P(a, b) \in \mathcal{A}$. Furthermore, given a language $\mathcal{L} \subseteq \mathbf{FO}$ and a theory $\mathcal{K} \subseteq \mathbf{FO}$, we denote by $\mathcal{L}^{Ent}(\mathcal{K})$ the set $\{\phi \in \mathcal{L}(\mathcal{K}) \mid \mathcal{K} \models \phi\}$. Obviously $\mathcal{L}^{Ent}(\mathcal{K}) \subseteq \mathcal{L}(\mathcal{K})$.

We then observe that a denial $\delta = \forall \vec{x}(cq(\vec{x}) \rightarrow \perp)$ is satisfied by an interpretation I , if $\exists \vec{x}(cq(\vec{x}))$ is not satisfied by I (i.e. it evaluates to false in I). As a consequence, $\mathcal{O} \cup \{\delta\}$ is consistent if and only if $\mathcal{O} \not\models \exists \vec{x}(cq(\vec{x}))$.

For the sake of completeness we report below the semantics of $DL\text{-}Lite_{\mathcal{R}}$ and \mathcal{EL}_{\perp} , i.e. say when an interpretation satisfies assertions in these languages. For the constructs of $DL\text{-}Lite_{\mathcal{R}}$ and \mathcal{EL}_{\perp} , the interpretation function is extended to non-atomic concept and role expressions as follows:

$$\begin{aligned} \top^I &= \Delta^I \\ \perp^I &= \emptyset \\ (P^-)^I &= \{ (o, o') \mid (o', o) \in P^I \} \\ (\exists R)^I &= \{ o \mid \exists o' (o, o') \in R^I \} \\ (\exists R.C)^I &= \{ o \mid \exists o' (o, o') \in R^I \wedge o' \in C^I \} \\ (C_1 \sqcap C_2)^I &= C_1^I \cap C_2^I \\ (\neg R)^I &= (\Delta^I \times \Delta^I) \setminus R^I \\ (\neg B)^I &= \Delta^I \setminus B^I \end{aligned}$$

An interpretation I satisfies a positive concept inclusion $B_1 \sqsubseteq B_2$ (resp., a general concept inclusion $C_1 \sqsubseteq C_2$, a positive role inclusion $R_1 \sqsubseteq R_2$) if $B_1^I \subseteq B_2^I$ (resp., $C_1^I \subseteq C_2^I$, $R_1^I \subseteq R_2^I$). Furthermore, I satisfies a negative concept inclusion $B_1 \sqsubseteq \neg B_2$ (resp., a negative role inclusion $R_1 \sqsubseteq \neg R_2$) if $B_1^I \cap B_2^I = \emptyset$ (resp., $R_1^I \cap R_2^I = \emptyset$). Finally, I satisfies ABox assertions $A(a)$ and $P(a, b)$ if $a^I \in A^I$ and $(a^I, b^I) \in P^I$, respectively.

In the following, given a TBox \mathcal{T} and an ABox \mathcal{A} , we denote by $\mathcal{A}_{\mathcal{T}}$ the *ground closure of \mathcal{A} with respect to \mathcal{T}* , i.e. the set of ABox assertions α such that the DL ontology $\mathcal{T} \cup \mathcal{A} \models \alpha$. Notice that $\mathcal{A}_{\mathcal{T}} = \mathbf{GA}^{Ent}(\mathcal{T} \cup \mathcal{A})$.

We then recall that entailment of BCQs in $DL\text{-}Lite_{\mathcal{R}}$ is FO rewritable, i.e. for every $DL\text{-}Lite_{\mathcal{R}}$ TBox \mathcal{T} and BCQ q , it is possible to effectively compute an FO query q_r , called the *perfect reformulation of q with respect to \mathcal{T}* , such that, for each ABox \mathcal{A} , $\mathcal{T} \cup \mathcal{A} \models q$ if and only if $I_{\mathcal{A}} \models q_r$. In this paper, we will make use of the algorithm PerfectRef presented in [18]. This algorithm computes a perfect reformulation of q with respect to \mathcal{T} by using only positive inclusions in \mathcal{T} as rewriting rules (we refer the reader to [18] for additional details). The following proposition establishes correctness of PerfectRef.

Proposition 1 ([18]). *Let $\mathcal{T} \cup \mathcal{A}$ be a consistent $DL\text{-}Lite_{\mathcal{R}}$ ontology and let q be a BCQ. Then, $\mathcal{T} \cup \mathcal{A} \models q$ if and only if $I_{\mathcal{A}} \models \text{PerfectRef}(q, \mathcal{T})$.*

We also point out that all the complexity results given in this paper are concerned with *data complexity* [23], that in our framework is the complexity computed only with respect to the size of the ABox. Specifically, we will refer to the following complexity classes [24]:

- NP, which is the set of decision problems decidable by a non-deterministic Turing machine in polynomial time;
- coNP, which is the set of decision problems whose complement is in NP;
- PTIME, which is the set of decision problems decidable by a deterministic Turing machine in polynomial time;
- LOGSPACE, which is the set of decision problems decidable by a deterministic Turing machine requiring a logarithmic amount of writable memory space;
- AC^0 , which is the set of decision problems decidable by a uniform family of circuits of constant depth and polynomial size, with unlimited fan-in AND gates and OR gates.

As already remarked in the introduction, although we limit our technical treatment to languages containing only closed formulae, our results also hold for open formulae.

3. CQE framework

Our framework for CQE is adapted from the one presented in [8].

An \mathcal{L} CQE instance \mathcal{E} is a triple $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where \mathcal{T} is a TBox in the DL \mathcal{L} , \mathcal{A} is an ABox such that $\mathcal{T} \cup \mathcal{A}$ is consistent, and \mathcal{P} is the *policy*, i.e. a set of denial assertions over the signature of $\mathcal{T} \cup \mathcal{A}$, such that $\mathcal{T} \cup \mathcal{P}$ is consistent. Intuitively, \mathcal{T} is the schema a user interacts with to pose her queries; \mathcal{A} is the dataset underlying the schema; \mathcal{P} specifies the knowledge that cannot be disclosed for

confidentiality reasons, in the sense that the user will never get, through query answers, sufficient knowledge to violate the denials in \mathcal{P} .² In the following, when the language \mathcal{L} used for the TBox is not specified we intend $\mathcal{L} = \mathbf{FO}$.

We then define a censor for a CQE instance.

Definition 1. Given a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ and a language $\mathcal{L}_c \subseteq \mathbf{FO}(\mathcal{T} \cup \mathcal{A})$, a *censor* for \mathcal{E} in \mathcal{L}_c (also called \mathcal{L}_c censor for \mathcal{E}) is a Boolean-valued function $\text{cens} : \mathcal{L}_c \rightarrow \{\text{true}, \text{false}\}$ such that:

- (i) for each $\phi \in \mathcal{L}_c$, if $\text{cens}(\phi) = \text{true}$ then $\mathcal{T} \cup \mathcal{A} \models \phi$, and
- (ii) $\mathcal{T} \cup \mathcal{P} \cup \text{Th}(\text{cens})$ is consistent, where $\text{Th}(\text{cens}) = \{\phi \in \mathcal{L}_c \mid \text{cens}(\phi) = \text{true}\}$ and is called the *theory of the censor* cens .

The set of theories of all the censors in \mathcal{L}_c for a CQE instance \mathcal{E} is denoted by $\text{ThS}_{\mathcal{L}_c}(\mathcal{E})$. The language \mathcal{L}_c is called the *censor language*.

Intuitively, the censor establishes if a sentence in \mathcal{L}_c can be divulged to the user while preserving the policy. More precisely, the sentences from \mathcal{L}_c that the censor discloses are the ones that are entailed by the TBox and the theory of the censor.

Among all censors, optimal ones are the most important, since they maximize the answers returned to users.

Definition 2. Given a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, a language $\mathcal{L}_c \subseteq \mathbf{FO}(\mathcal{T} \cup \mathcal{A})$, and a censor cens for \mathcal{E} in \mathcal{L}_c , we say that cens is *optimal* for \mathcal{E} in \mathcal{L}_c if there does not exist a censor cens' for \mathcal{E} in \mathcal{L}_c such that $\text{Th}(\text{cens}) \subset \text{Th}(\text{cens}')$.

The set of theories of all the optimal censors in \mathcal{L}_c for a CQE instance \mathcal{E} is denoted by $\text{OTHS}_{\mathcal{L}_c}(\mathcal{E})$.

Example 3. Let us consider again the pharmacy domain used in Example 2, but for the sake of simplicity do not bring TBox assertions into play. We define the CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where $\mathcal{T} = \emptyset$, $\mathcal{A} = \{\text{buy}(c_1, m_A), \text{buy}(c_1, m_B), \text{buy}(c_2, m_A)\}$, and $\mathcal{P} = \{\forall x(\text{buy}(x, m_A) \wedge \text{buy}(x, m_B) \rightarrow \perp)\}$. The policy specifies as confidential the fact that a customer buys both medicine A and medicine B (this may reveal an embarrassing disease). The optimal censors for \mathcal{E} in $\mathbf{CQ}(\mathcal{T} \cup \mathcal{A})$ are only cens_1 and cens_2 , where:

- $\text{Th}(\text{cens}_1)$ consists of the BCQs $\exists x(\text{buy}(x, m_B))$, $\text{buy}(c_1, m_A)$, $\text{buy}(c_2, m_A)$, as well as all the queries in $\mathbf{CQ}(\mathcal{T} \cup \mathcal{A})$ that these BCQs infer;
- $\text{Th}(\text{cens}_2)$ consists of the BCQs $\text{buy}(c_1, m_B)$, $\text{buy}(c_2, m_A)$, as well as all the queries in $\mathbf{CQ}(\mathcal{T} \cup \mathcal{A})$ that these BCQs infer;

If we instead restrict the censor language to \mathcal{A} (i.e. censor theories can only contain facts of the ABox), we still have only two optimal censors, which are as follows:

- $\text{Th}(\text{cens}_3) = \{\text{buy}(c_1, m_A), \text{buy}(c_2, m_A)\}$;
- $\text{Th}(\text{cens}_4) = \{\text{buy}(c_1, m_B), \text{buy}(c_2, m_A)\}$.

Notice that, since in this example the TBox is empty, the above censors cens_3 and cens_4 are also optimal censors for \mathcal{E} in **GA**. \square

We introduce below a notation which is useful for formalizing the entailment problems studied in this paper.

Definition 3. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance, $\mathcal{L}_c \subseteq \mathbf{FO}(\mathcal{T} \cup \mathcal{A})$, and $\phi \in \mathbf{FO}(\mathcal{T} \cup \mathcal{A})$. We write $\mathcal{E} \models_{\mathcal{L}_c}^{cqe} \phi$, read as \mathcal{E} entails ϕ under \mathcal{L}_c censors, if $\mathcal{T} \cup \text{Th} \models \phi$ for every $\text{Th} \in \text{OTHS}_{\mathcal{L}_c}(\mathcal{E})$.

We are now ready to define entailment decision problems in CQE for the censor languages $\mathbf{CQ}(\mathcal{T} \cup \mathcal{A})$, $\mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A})$ (for every fixed integer $k \geq 1$), **GA**($\mathcal{T} \cup \mathcal{A}$), and \mathcal{A} , where \mathcal{T} and \mathcal{A} are respectively the TBox and the ABox of the CQE instance at the hand. We will refer to these problems as entailment under **CQ** censors, entailment under \mathbf{CQ}_k censors, entailment under **GA** censors, and entailment under ABox censors, respectively. The definitions are parametric with respect to the ontology language (i.e. the language used to express the TBox) and the query language (i.e. the language used to express the user query).

PROBLEM:	CQ-Cens-Entailment ($\mathcal{L}_{\mathcal{T}}, \mathcal{Q}$)
INPUT:	A $\mathcal{L}_{\mathcal{T}}$ CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, a Boolean query $\phi \in \mathcal{Q}$
QUESTION:	Does $\mathcal{E} \models_{\mathbf{CQ}(\mathcal{T} \cup \mathcal{A})}^{cqe} \phi$?

² Our notion of policy generalizes the one given in [8], where \mathcal{P} is a single CQ.

PROBLEM:	CQ_k-Cens-Entailment($\mathcal{L}_{\mathcal{T}}, \mathcal{Q}$)
INPUT:	A $\mathcal{L}_{\mathcal{T}}$ CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, a Boolean query $\phi \in \mathcal{Q}$
QUESTION:	Does $\mathcal{E} \models_{\text{CQ}_k(\mathcal{T} \cup \mathcal{A})}^{cqe} \phi$?

PROBLEM:	GA-Cens-Entailment($\mathcal{L}_{\mathcal{T}}, \mathcal{Q}$)
INPUT:	A $\mathcal{L}_{\mathcal{T}}$ CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, a Boolean query $\phi \in \mathcal{Q}$
QUESTION:	Does $\mathcal{E} \models_{\text{GA}(\mathcal{T} \cup \mathcal{A})}^{cqe} \phi$?

PROBLEM:	ABox-Cens-Entailment($\mathcal{L}_{\mathcal{T}}, \mathcal{Q}$)
INPUT:	A $\mathcal{L}_{\mathcal{T}}$ CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, a Boolean query $\phi \in \mathcal{Q}$
QUESTION:	Does $\mathcal{E} \models_{\mathcal{A}}^{cqe} \phi$?

We remark that all the above mentioned problems collapse to standard query entailment in $\mathcal{T} \cup \mathcal{A}$ in the case when $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A}$ is consistent, that is, if the original ontology does not contain confidential data with respect to the policy.

Hereinafter, to simplify the notation, we will sometimes omit to specify that a censor language is limited to the signature of $\mathcal{T} \cup \mathcal{A}$, e.g. we will use **CQ** instead of **CQ**($\mathcal{T} \cup \mathcal{A}$), when the signature is clear from the context.

The following example shows that, in general, the entailment problems above defined are incomparable to each other, when $\mathcal{Q} = \text{CQ}$. Moreover, it also shows that, if we consider as censor languages the ABox and **GA** or the ABox and **CQ**, then the incomparability holds even for instance checking.³

Example 4. Consider the following CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where:

$$\begin{aligned} \mathcal{T} &= \{ B \sqsubseteq C, B \sqsubseteq D, E \sqsubseteq \exists R \} \\ \mathcal{A} &= \{ A(o_1), B(o_1), E(o_2), F(o_2) \} \\ \mathcal{P} &= \{ \forall x(A(x) \wedge C(x) \rightarrow \perp), \quad \forall x, y(F(x) \wedge R(x, y) \rightarrow \perp), \\ &\quad \forall x(B(x) \wedge C(x) \rightarrow \perp), \quad \forall x, y(E(x) \wedge R(x, y) \rightarrow \perp) \} \end{aligned}$$

If the censor language is the CQE instance ABox \mathcal{A} , then we have only one optimal censor cens_1 for \mathcal{E} whose theory is:

$$\text{Th}(\text{cens}_1) = \{ A(o_1), F(o_2) \}$$

Differently, if we consider **GA** as censor language, then we have the two optimal censors cens_2 and cens_3 for \mathcal{E} whose theories are:

$$\begin{aligned} \text{Th}(\text{cens}_2) &= \{ A(o_1), D(o_1), F(o_2) \} \\ \text{Th}(\text{cens}_3) &= \{ C(o_1), D(o_1), F(o_2) \} \end{aligned}$$

In case the censor language is **CQ**, we have four optimal censors for \mathcal{E} that are as follows:

- $\text{Th}(\text{cens}_4)$ contains the sentences $A(o_1), D(o_1), F(o_2), \exists x(C(x)), \exists x, y(R(x, y))$, and all the sentences in **CQ** inferred by them;
- $\text{Th}(\text{cens}_5)$ contains the sentences $A(o_1), D(o_1), \exists y(R(o_2, y)), \exists x(C(x)), \exists x(F(x))$, and all the sentences in **CQ** inferred by them;
- $\text{Th}(\text{cens}_6)$ contains the sentences $C(o_1), D(o_1), F(o_2), \exists x(A(x)), \exists x, y(R(x, y))$, and all the sentences in **CQ** inferred by them;
- $\text{Th}(\text{cens}_7)$ contains the sentences $C(o_1), D(o_1), \exists y(R(o_2, y)), \exists x(A(x)), \exists x(F(x))$, and all the sentences in **CQ** inferred by them.

Now, consider the following sentences:

$$\phi_1 = A(o_1), \quad \phi_2 = D(o_1), \quad \phi_3 = F(o_2), \quad \phi_4 = \exists x, y(R(x, y)).$$

We have that:

- $\mathcal{E} \models_{\mathcal{A}}^{cqe} \phi_1$, whereas $\mathcal{E} \not\models_{\text{GA}}^{cqe} \phi_1$ and $\mathcal{E} \not\models_{\text{CQ}}^{cqe} \phi_1$;
- $\mathcal{E} \models_{\text{GA}}^{cqe} \phi_2$ and $\mathcal{E} \models_{\text{CQ}}^{cqe} \phi_2$, whereas $\mathcal{E} \not\models_{\mathcal{A}}^{cqe} \phi_2$;

³ In the example, we show the pairwise incomparability between entailment under ABox censors, **GA** censors, and **CQ** censors. Moreover, it is easy to verify that the CQE instance considered here also shows the pairwise incomparability between entailment under ABox censors, **GA** censors, and **CQ**₁ censors. Finally, the incomparability between entailment under **CQ**_k censors and **CQ** censors will be illustrated in Section 6 (see in particular the proof of Theorem 12).

- $\mathcal{E} \vDash_{\mathbf{GA}}^{cqe} \phi_3$ and $\mathcal{E} \vDash_{\mathbf{A}}^{cqe} \phi_3$, whereas $\mathcal{E} \not\vDash_{\mathbf{CQ}}^{cqe} \phi_3$;
- $\mathcal{E} \vDash_{\mathbf{CQ}}^{cqe} \phi_4$, whereas $\mathcal{E} \not\vDash_{\mathbf{A}}^{cqe} \phi_4$ and $\mathcal{E} \not\vDash_{\mathbf{GA}}^{cqe} \phi_4$. \square

Surprisingly, we can show that instance checking under **CQ** censors is a sound approximation of instance checking under **GA** censors. Indeed, this property is a consequence of a more general result that we establish for censor languages of increasing expressiveness. To prove this result, we need the following lemma (which will be also useful for other results of the next sections).

Lemma 1. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance, $\mathcal{L} \subseteq \mathbf{FO}$ be a language, $\phi \in \mathcal{L}$ be a sentence, and $\text{Th} \in \text{OThS}_{\mathcal{L}}(\mathcal{E})$ be the theory of an optimal censor for \mathcal{E} in \mathcal{L} . We have that $\mathcal{T} \cup \text{Th} \vDash \phi$ if and only if $\phi \in \text{Th}$.*

Proof. The if-part is immediate. Indeed, if $\phi \in \text{Th}$, then obviously $\mathcal{T} \cup \text{Th} \vDash \phi$.

Suppose now that $\mathcal{T} \cup \text{Th} \vDash \phi$. By way of contradiction, assume that $\phi \notin \text{Th}$. We now prove that $\text{Th}' = \text{Th} \cup \{\phi\}$ is such that $\text{Th}' \in \text{ThS}_{\mathcal{L}}(\mathcal{E})$, thus as per Definition 2 contradicting the fact that $\text{Th} \in \text{OThS}_{\mathcal{L}}(\mathcal{E})$ because $\text{Th} \subset \text{Th}'$.

By condition (i) of Definition 1, we know that Th can contain only sentences entailed by $\mathcal{T} \cup \mathcal{A}$. So, since $\mathcal{T} \cup \text{Th} \vDash \phi$ by assumption, due to the monotonicity of first-order logic we derive that $\mathcal{T} \cup \mathcal{A} \vDash \phi$ as well. Together with the fact that $\text{Th} \in \text{OThS}_{\mathcal{L}}(\mathcal{E})$, this implies that the function cens' over \mathcal{L} such that $\text{cens}'(\phi) = \text{true}$ if and only if $\phi \in \text{Th}'$ satisfies condition (i) of Definition 1. Furthermore, by condition (ii) of Definition 1 we know that $\mathcal{T} \cup \mathcal{P} \cup \text{Th}$ is consistent. So, since $\mathcal{T} \cup \text{Th} \vDash \phi$ by assumption, we derive that $\mathcal{T} \cup \mathcal{P} \cup \text{Th} \cup \{\phi\}$ is consistent as well. This implies that the function cens' also satisfies condition (ii) of Definition 1, and therefore cens' is a censor for \mathcal{E} in \mathcal{L} and Th' is its theory. Thus, $\text{Th}' \in \text{ThS}_{\mathcal{L}}(\mathcal{E})$, as required. \square

We are now able to prove the property we mentioned above.

Proposition 2. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance, \mathcal{L}_1 and \mathcal{L}_2 be two languages such that $\mathcal{L}_1 \subseteq \mathcal{L}_2 \subseteq \mathbf{FO}$, and $\phi \in \mathcal{L}_1$ be a sentence. We have that if $\mathcal{E} \vDash_{\mathcal{L}_2}^{cqe} \phi$, then $\mathcal{E} \vDash_{\mathcal{L}_1}^{cqe} \phi$.*

Proof. Assume $\mathcal{E} \not\vDash_{\mathcal{L}_1}^{cqe} \phi$, i.e. there exists $\text{Th}_{\mathcal{L}_1} \in \text{OThS}_{\mathcal{L}_1}(\mathcal{E})$ such that $\mathcal{T} \cup \text{Th}_{\mathcal{L}_1} \not\vDash \phi$. Since $\mathcal{L}_1 \subseteq \mathcal{L}_2$, one can easily obtain a theory $\text{Th}_{\mathcal{L}_2} \in \text{OThS}_{\mathcal{L}_2}(\mathcal{E})$ starting from $\text{Th}_{\mathcal{L}_1}$ as follows: $\text{Th}_{\mathcal{L}_2} = \text{Th}_{\mathcal{L}_1} \cup \Delta$, where Δ is any maximal subset of $\mathcal{A}_{\mathcal{T}}^{\mathcal{L}_2} = \{\alpha \in \mathcal{L}_2 \mid \mathcal{T} \cup \mathcal{A} \vDash \alpha\}$ such that $\mathcal{T} \cup \mathcal{P} \cup \text{Th}_{\mathcal{L}_1} \cup \Delta$ is consistent.

Using the same arguments provided in the proof of Lemma 1, there cannot be an $\alpha \in \mathcal{L}_1$ such that $\alpha \in \Delta$ and $\alpha \notin \text{Th}_{\mathcal{L}_1}$, otherwise this would contradict the optimality of $\text{Th}_{\mathcal{L}_1}$. So $\phi \notin \Delta$ because $\mathcal{T} \cup \text{Th}_{\mathcal{L}_1} \not\vDash \phi$. Since $\phi \in \mathcal{L}_1$ by initial assumption, $\phi \notin \text{Th}_{\mathcal{L}_1}$, and $\phi \notin \Delta$, we derive that $\phi \notin \text{Th}_{\mathcal{L}_2}$.

Due to Lemma 1, we know that $\mathcal{T} \cup \text{Th}_{\mathcal{L}_2} \vDash \alpha$ if and only if $\alpha \in \text{Th}_{\mathcal{L}_2}$ holds for any $\alpha \in \mathcal{L}_2$. Thus, since $\phi \in \mathcal{L}_1$, $\mathcal{L}_1 \subseteq \mathcal{L}_2$, and $\phi \notin \text{Th}_{\mathcal{L}_2}$, we can immediately derive that $\mathcal{T} \cup \text{Th}_{\mathcal{L}_2} \not\vDash \phi$ as well. Hence, it follows that $\mathcal{E} \not\vDash_{\mathcal{L}_2}^{cqe} \phi$. \square

We remark that the converse of the above property does not necessarily hold, as Example 4 shows for **CQ** and **GA**.

Moreover, the following result, whose proof easily follows from Definition 1, shows that censors are independent from the syntax used to express the CQE instance.

Proposition 3. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ and $\mathcal{E}' = \langle \mathcal{T}', \mathcal{A}', \mathcal{P}' \rangle$ be two CQE instances such that $\text{sig}(\mathcal{T} \cup \mathcal{A}) = \text{sig}(\mathcal{T}' \cup \mathcal{A}')$, $\text{Mod}(\mathcal{T} \cup \mathcal{A}) = \text{Mod}(\mathcal{T}' \cup \mathcal{A}')$, and $\text{Mod}(\mathcal{T} \cup \mathcal{P}) = \text{Mod}(\mathcal{T}' \cup \mathcal{P}')$, and let $\mathcal{L}_c \subseteq \mathbf{FO}$ be a language. We have that $\text{ThS}_{\mathcal{L}_c}(\mathcal{E}) = \text{ThS}_{\mathcal{L}_c}(\mathcal{E}')$.*

Given a censor language \mathcal{L}_c and two CQE instances \mathcal{E} and \mathcal{E}' , if $\text{ThS}_{\mathcal{L}_c}(\mathcal{E}) = \text{ThS}_{\mathcal{L}_c}(\mathcal{E}')$, then we say that \mathcal{E} and \mathcal{E}' are *CQE-equivalent* with respect to \mathcal{L}_c .

Proposition 3 immediately implies that GA-Cens-Entailment in a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ coincides with GA-Cens-Entailment in $\mathcal{E}' = \langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$.

Proposition 4. *Let $\mathcal{L} \subseteq \mathbf{FO}$, $\phi \in \mathcal{L}$, and $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance. We have that $\mathcal{E} \vDash_{\mathbf{GA}}^{cqe} \phi$ if and only if $\mathcal{E}' \vDash_{\mathcal{A}_{\mathcal{T}}}^{cqe} \phi$, where $\mathcal{E}' = \langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$.*

We conclude this section by defining the notion of FO rewritability for BCQ entailments in CQE, which will be used in the next two sections.

Let $\mathcal{X} \in \{\text{ABox}, \mathbf{GA}, \mathbf{CQ}_k, \mathbf{CQ}\}$ and $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}, \mathbf{CQ}\}$. We say that \mathcal{X} -Cens-Entailment(\mathcal{L}, \mathcal{Q}) is FO rewritable if for each TBox \mathcal{T} in the language \mathcal{L} and policy \mathcal{P} such that $\mathcal{T} \cup \mathcal{P}$ is consistent, and for each $q \in \mathcal{Q}$, it is possible to effectively compute an FO query q_r such that, for each ABox \mathcal{A} such that $\mathcal{T} \cup \mathcal{A}$ is consistent, $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle \vDash_{\mathcal{X}}^{cqe} q$ if and only if $\mathcal{I}_{\mathcal{A}} \vDash q_r$.

4. Relationship between CQE and CQA

In this section, we discuss the relationship between the CQE framework we have just defined and CQA.

An \mathcal{L} CQA instance \mathcal{J} is a pair $\langle \mathcal{T}, \mathcal{A} \rangle$, where \mathcal{T} is a consistent TBox in the DL \mathcal{L} , \mathcal{A} is an ABox, and $\mathcal{T} \cup \mathcal{A}$ is a possibly inconsistent ontology. The semantics of CQA is based on the notion of *repair*. Below, we provide the prototypical definition of a *repair* for a CQA instance.

Definition 4. A *repair* for a CQA instance $\mathcal{J} = \langle \mathcal{T}, \mathcal{A} \rangle$ is a set \mathcal{R} such that:

- (i) $\mathcal{R} \subseteq \mathcal{A}$;
- (ii) $\mathcal{T} \cup \mathcal{R}$ is consistent;
- (iii) there does not exist any \mathcal{R}' such that $\mathcal{R} \subset \mathcal{R}' \subseteq \mathcal{A}$ and $\mathcal{T} \cup \mathcal{R}'$ is consistent.

We denote by $\text{RepS}_{\mathcal{A}}(\mathcal{J})$ the set of repairs for a CQA instance \mathcal{J} .

Given a CQA instance $\mathcal{J} = \langle \mathcal{T}, \mathcal{A} \rangle$, each repair $\mathcal{R} \in \text{RepS}_{\mathcal{A}}(\mathcal{J})$ aims to restore consistency with \mathcal{T} while preserving as many facts as possible of those belonging to \mathcal{A} . Indeed, in the DL literature, the above semantics for CQA is known with the name of ABox-repair semantics, or simply *AR*-semantics, and is among the ones that have been most thoroughly investigated [14,17,11,25].

Based on the notion of repair given in the above definition, we define entailment in CQA under the *AR*-semantics.

Definition 5. Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a CQA instance and $\phi \in \mathbf{FO}$. We say that \mathcal{J} *AR-entails* ϕ , denoted by $\mathcal{J} \models_{AR} \phi$, if $\mathcal{T} \cup \mathcal{R} \models \phi$ for every $\mathcal{R} \in \text{RepS}_{\mathcal{A}}(\mathcal{J})$.

We now provide some results that allow to establish correspondences between (theories of) censors and repairs.

The following lemma shows that, for the censor language \mathcal{A} , the set of theories of the optimal censors for a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ coincides with the set of repairs for the CQA instance $\mathcal{J} = \langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle$.

Lemma 2. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance. We have that $\text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle) = \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)$.

Proof. We start by showing that $\text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle) \subseteq \text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle)$. Let $\mathcal{R} \in \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)$. From Definition 4, we have that \mathcal{R} is a maximal subset of \mathcal{A} such that $\mathcal{T} \cup \mathcal{P} \cup \mathcal{R}$ is consistent. Therefore, from Definition 1 and Definition 2, there exists a censor \mathcal{C} for \mathcal{E} in \mathcal{A} such that $\mathcal{R} = \text{Th}(\text{cens})$ and $\text{Th}(\text{cens}) \in \text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle)$.

We now show that $\text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle) \subseteq \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)$. We soon notice that $\mathcal{T} \cup \mathcal{P}$ is consistent by definition of CQE instance, and thus $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle$ is indeed a CQA instance. Let us now consider any $\Gamma \in \text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle)$. From Definition 1 we have that $\Gamma \subseteq \mathcal{A}$. Hence, condition (i) of Definition 4 is verified for Γ . Fulfillment of condition (ii) of Definition 4 directly follows from condition (ii) of Definition 1. Now, suppose, by way of contradiction, that $\Gamma \notin \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)$. This may happen only if condition (iii) of Definition 4 is not satisfied. This means that there exists an $\mathcal{R} \in \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)$ such that $\Gamma \subset \mathcal{R}$. As we have shown before, $\text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle) \subseteq \text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle)$, and thus $\mathcal{R} \in \text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle)$, which contradicts the fact that $\Gamma \in \text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle)$. Therefore, condition (iii) of Definition 4 is satisfied, and so $\Gamma \in \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)$. \square

In order to shift the above correspondence between censors and repairs at the level of entailment problems in the CQA and CQE frameworks, we need to introduce an additional definition.

Definition 6. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance and $\mathcal{L} \subseteq \mathbf{FO}$. We say that \mathcal{E} is *policy independent* with respect to \mathcal{L} -entailment, or simply *\mathcal{L} -policy independent*, if for every $\phi \in \mathcal{L}$ and every $\mathcal{A}' \subseteq \mathcal{A}$ such that $\mathcal{T} \cup \mathcal{A}' \cup \mathcal{P}$ is consistent, we have that $\mathcal{T} \cup \mathcal{A}' \cup \mathcal{P} \models \phi$ if and only if $\mathcal{T} \cup \mathcal{A}' \models \phi$.

As we will show later, every *DL-Lite_R* or *$\mathcal{E}\mathcal{L}_{\perp}$* CQE instance is **CQ**-policy independent. As an example of non-policy independence, consider the CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where $\mathcal{T} = \{A \sqsubseteq B \sqcup C\}$ (i.e. the TBox contains an axiom stating that instances of concept A must be also instances of the union of concept B and concept C), $\mathcal{A} = \{A(a)\}$, and $\mathcal{P} = \{\forall x(A(x) \wedge B(x) \rightarrow \perp)\}$. It is not difficult to see that the query $C(a)$ is entailed by $\mathcal{T} \cup \mathcal{A} \cup \mathcal{P}$ but it is not entailed by $\mathcal{T} \cup \mathcal{A}$.

Clearly, by definition, if a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ is \mathcal{L} -policy independent for a language $\mathcal{L} \subseteq \mathbf{FO}$, then \mathcal{E} is also \mathcal{L}' -policy independent for every language \mathcal{L}' such that $\mathcal{L}' \subseteq \mathcal{L}$. Roughly speaking, policy independence with respect to a language \mathcal{L} guarantees that in a CQE instance the sentences in the policy act only as constraints on top of $\mathcal{T} \cup \mathcal{A}$ for the entailment of sentences in \mathcal{L} . Observe that $\mathcal{T} \cup \mathcal{A}$ can contradict denials in \mathcal{P} .

The following theorem establishes the relationship existing between entailment of sentences in \mathcal{L} under ABox censors and under the *AR*-semantics for an \mathcal{L} -policy independent CQE instance.

Theorem 1. Let $\mathcal{L} \subseteq \mathbf{FO}$, $\phi \in \mathcal{L}$, and $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be an \mathcal{L} -policy independent CQE instance. We have that $\mathcal{E} \models_{\mathcal{A}}^{cqe} \phi$ if and only if $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle \models_{AR} \phi$.

Proof. Since \mathcal{E} is \mathcal{L} -policy independent, then, for each $\phi \in \mathcal{L}$ and for each repair $\mathcal{R} \in \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)$, we have that $\mathcal{T} \cup \mathcal{P} \cup \mathcal{R} \models \phi$ if and only if $\mathcal{T} \cup \mathcal{R} \models \phi$ (recall that $\mathcal{R} \subseteq \mathcal{A}$ by definition). Then, the claim directly follows from Lemma 2, since $\text{OThS}_{\mathcal{A}}(\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle) = \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)$. \square

The following lemma is the analog of Lemma 2 for optimal censors in GA. Notice that in this case the AR-repairs are computed for an ontology whose ABox is the ground closure $\mathcal{A}_{\mathcal{T}}$, where \mathcal{A} and \mathcal{T} are the ABox and the TBox of the CQE instance.

Lemma 3. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance. We have that $\text{OThS}_{\text{GA}}(\mathcal{E}) = \text{RepS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle)$.

Proof. Since $\text{Mod}(\mathcal{T} \cup \mathcal{A}) = \text{Mod}(\mathcal{T} \cup \mathcal{A}_{\mathcal{T}})$, then, from Proposition 3, we derive that \mathcal{E} and $\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$ are CQE-equivalent with respect to any censor language $\mathcal{L}_c \subseteq \text{FO}$. So, if we consider $\mathcal{L}_c = \text{GA}$, then we have that $\text{ThS}_{\text{GA}}(\mathcal{E}) = \text{ThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$, which clearly implies that $\text{OThS}_{\text{GA}}(\mathcal{E}) = \text{OThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$.

Now we prove that $\text{OThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle) = \text{RepS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle)$. Let $\Gamma \in \text{ThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$. From Definition 1, it follows that $\Gamma \subseteq \text{GA}$, that $\mathcal{T} \cup \mathcal{A}_{\mathcal{T}} \models \gamma$ for each $\gamma \in \Gamma$, and that $\mathcal{T} \cup \mathcal{P} \cup \Gamma$ is consistent. Now, since for every $\alpha \in \text{GA}$ we have that $\mathcal{T} \cup \mathcal{A}_{\mathcal{T}} \models \alpha$ if and only if $\alpha \in \mathcal{A}_{\mathcal{T}}$, we have that $\Gamma \subseteq \mathcal{A}_{\mathcal{T}}$. So, since $\mathcal{T} \cup \mathcal{P} \cup \Gamma$ is consistent, by exploiting again Definition 1, we have that it is possible to define a censor cens for $\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$ in $\mathcal{A}_{\mathcal{T}}$ such that $\Gamma = \text{Th}(\text{cens})$. So, $\Gamma \in \text{ThS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$. Thus, $\text{ThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle) \subseteq \text{ThS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$. The other way around, i.e. $\text{ThS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle) \subseteq \text{ThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$, can be shown in the same way by observing that $\mathcal{A}_{\mathcal{T}} \subseteq \text{GA}$. We have that $\text{ThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle) = \text{ThS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$, and therefore that $\text{OThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle) = \text{OThS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$.

From the results given so far, we have the following equalities: $\text{OThS}_{\text{GA}}(\mathcal{E}) = \text{OThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$ and $\text{OThS}_{\text{GA}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle) = \text{OThS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$, from which it directly follows that $\text{OThS}_{\text{GA}}(\mathcal{E}) = \text{OThS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle)$. By exploiting Lemma 2, we also have that $\text{OThS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle) = \text{RepS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle)$, from which we have the claim. \square

Lemma 3 soon implies the following result on entailments, when policy independency comes into play.

Theorem 2. Let $\mathcal{L} \subseteq \text{FO}$, $\phi \in \mathcal{L}$, and $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a CQE instance such that $\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$ is \mathcal{L} -policy independent. We have that $\mathcal{E} \models_{\text{GA}}^{cqe} \phi$ if and only if $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \models_{\text{AR}} \phi$.

Proof. From Lemma 3 we have that $\text{OThS}_{\text{GA}}(\mathcal{E}) = \text{RepS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle)$. Since $\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$ is \mathcal{L} -policy independent, from Theorem 1, we derive that $\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle \models_{\mathcal{A}_{\mathcal{T}}}^{cqe} \phi$ if and only if $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \models_{\text{AR}} \phi$ holds for each $\phi \in \mathcal{L}$. Due to Proposition 4, we have $\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle \models_{\mathcal{A}_{\mathcal{T}}}^{cqe} \phi$ if and only if $\mathcal{E} \models_{\text{GA}}^{cqe} \phi$. It follows that, for each $\phi \in \mathcal{L}$, $\mathcal{E} \models_{\text{GA}}^{cqe} \phi$ if and only if $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \models_{\text{AR}} \phi$, as required. \square

We conclude this section by considering another popular semantics, called IAR, originally introduced in [14]. Interestingly, entailment of BCQs under the IAR-semantics is in AC^0 in data complexity. We first provide the formal definition of IAR-entailment, and then discuss its relationship with GA-Cens-Entailment.

Definition 7. Let $\mathcal{J} = \langle \mathcal{T}, \mathcal{A} \rangle$ be a CQA instance and $\phi \in \text{FO}$. We say that \mathcal{J} IAR-entails ϕ , denoted by $\mathcal{J} \models_{\text{IAR}} \phi$, if $\mathcal{T} \cup \mathcal{A}_{\cap} \models \phi$, where $\mathcal{A}_{\cap} = \bigcap_{\mathcal{A}_i \in \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)} \mathcal{A}_i$.

For the case of instance checking, we can establish a precise relationship between IAR-entailment and GA-Cens-Entailment.

Theorem 3. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a GA-policy independent CQE instance and let $\gamma \in \text{GA}$. We have that $\mathcal{E} \models_{\text{GA}}^{cqe} \gamma$ if and only if $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \models_{\text{IAR}} \gamma$.

Proof. Suppose that $\mathcal{E} \models_{\text{GA}}^{cqe} \gamma$. As per Definition 3 we have that $\mathcal{T} \cup \Gamma \models \gamma$ for each $\Gamma \in \text{OThS}_{\text{GA}}(\mathcal{E})$. Thus, by Lemma 1 we immediately derive that $\gamma \in \Gamma$ for each $\Gamma \in \text{OThS}_{\text{GA}}(\mathcal{E})$. Moreover, from Lemma 3 we have that $\text{OThS}_{\text{GA}}(\mathcal{E}) = \text{RepS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle)$. Hence, $\gamma \in \Gamma$ for each $\Gamma \in \text{RepS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle)$. So, we derive that $\gamma \in \mathcal{A}'$ as well, where $\mathcal{A}' = \bigcap_{\mathcal{A}_i \in \text{RepS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle)} \mathcal{A}_i$. From the definition of IAR-entailment (Definition 7), this clearly implies that $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \models_{\text{IAR}} \gamma$, as required.

Conversely, suppose that $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \models_{\text{IAR}} \gamma$. Since IAR-entailment is a sound approximation of AR-entailment [14], we can derive that $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \models_{\text{AR}} \gamma$. Since $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ is a GA-policy independent CQE instance, the CQE instance $\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$ is GA-policy independent as well. Finally, the facts that $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \models_{\text{AR}} \gamma$ and $\langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$ is a GA-policy independent CQE instance, due to Theorem 2 directly imply that $\mathcal{E} \models_{\text{GA}}^{cqe} \gamma$. \square

The following example shows that Theorem 3 does not extend to entailment of BCQs (actually, already of BCQ_{\exists} s) rather than instance checking.

Example 5. Consider the following CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where:

$$\begin{aligned}\mathcal{T} &= \{ B \sqsubseteq \exists P, C \sqsubseteq \exists P \} \\ \mathcal{A} &= \{ B(o), C(o) \} \\ \mathcal{P} &= \{ \forall x(B(x) \wedge C(x) \rightarrow \perp) \}\end{aligned}$$

If we consider **GA** as censor language, then we have the two optimal censors cens_1 and cens_2 for \mathcal{E} whose theories are:

- $\text{Th}(\text{cens}_1) = \{B(o)\}$
- $\text{Th}(\text{cens}_2) = \{C(o)\}$

Consider the CQA instance $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle$. Notice that $\mathcal{A} = \mathcal{A}_{\mathcal{T}}$, and then $\mathcal{A}_{\cap} = \bigcap_{\mathcal{A}_i \in \text{RepS}_{\mathcal{A}_{\mathcal{T}}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle)} \mathcal{A}_i = \bigcap_{\mathcal{A}_i \in \text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle)} \mathcal{A}_i$. Furthermore, we have $\text{RepS}_{\mathcal{A}}(\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle) = \text{OThS}_{\text{GA}}(\mathcal{E}) = \{\text{Th}(\text{cens}_1), \text{Th}(\text{cens}_2)\}$ (cf. Lemma 2), and therefore $\mathcal{A}_{\cap} = \emptyset$.

Consider now the BCQ $_{\exists}$ $\phi = \exists x, y(P(x, y))$. We have that $\mathcal{E} \models_{\text{GA}}^{\text{cqe}} \phi$, whereas $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A}_{\mathcal{T}} \rangle \not\models_{\text{IAR}} \phi$. \square

In this section we have given results about the correspondence between entailment problems in CQA and entailments under either ABox or **GA** censors in CQE. In the next section, we will exploit the established relationships for devising data complexity results for ABox-Cens-Entailment and GA-Cens-Entailment of BCQs in the case of $DL\text{-Lite}_{\mathcal{R}}$ and \mathcal{EL}_{\perp} CQE instances. As for CQ_k -Cens-Entailment and CQ-Cens-Entailment, we notice that there is no semantic counterpart studied in CQA. Computational complexity of these two decision problems will be attacked in Section 6 and Section 7 through a tailored approach.

5. CQE under restricted censor languages

In this section we establish the data complexity of the problems $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$ and $GA\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$, where $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\text{GA}, \text{CQ}_{\exists}, \text{CQ}\}$. These results allow us to clarify the computational properties of query answering in CQE when we adopt restricted censor languages, i.e. languages that can be less expressive than the language used to formulate the queries. We devise such results by exploiting the connection between entailments in CQE and in CQA.

Let $\delta = \forall \bar{x}(\phi(\bar{x}) \rightarrow \perp) \in \mathcal{P}$ be a denial assertion. In the following, we denote by q_{δ} the BCQ $\exists \bar{x}(\phi(\bar{x}))$ (that is equivalent to the negation of δ).

The next proposition shows that CQE instances in the DL languages considered in this paper are **CQ**-policy independent. To this end, we make use of the following folklore lemma.

Lemma 4. *Let \mathcal{T} be either a $DL\text{-Lite}_{\mathcal{R}}$ or \mathcal{EL}_{\perp} TBox, let Φ be a set of BCQs such that $\mathcal{T} \cup \Phi$ is consistent, and let \mathcal{P} be a policy. Then, $\Phi \cup \mathcal{T} \cup \mathcal{P}$ is consistent if and only if $\mathcal{T} \cup \Phi \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$.*

Proof. Suppose $\mathcal{T} \cup \Phi \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$. Now, let \mathcal{A}_{Φ} be the ABox obtained from Φ by taking the set of atoms occurring in Φ and considering every variable symbol as a new individual (assuming that different BCQs in Φ use different variable symbols). It is immediate to verify that, for every TBox \mathcal{T} such that \mathcal{T} is either a $DL\text{-Lite}_{\mathcal{R}}$ or a \mathcal{EL}_{\perp} TBox, and for every FO sentence φ that does not mention the new individuals, $\mathcal{T} \cup \Phi \models \varphi$ if and only if $\mathcal{T} \cup \mathcal{A}_{\Phi} \models \varphi$, which implies that $\mathcal{T} \cup \mathcal{P} \cup \Phi$ is consistent if and only if $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A}_{\Phi}$ is consistent. Now, by hypothesis $\mathcal{T} \cup \Phi$ is consistent, so $\mathcal{T} \cup \mathcal{A}_{\Phi}$ is consistent. Hence, as shown in [26], $\mathcal{T} \cup \mathcal{A}_{\Phi} \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$ iff $\mathcal{U} \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$ (i.e. $\mathcal{U} \models \mathcal{P}$), where \mathcal{U} is the *universal model* of $\mathcal{T} \cup \mathcal{A}_{\Phi}$.⁴ Since $\mathcal{U} \models \mathcal{T} \cup \mathcal{A}_{\Phi}$ and $\mathcal{U} \models \mathcal{P}$, \mathcal{U} is a model of $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A}_{\Phi}$ as well. It follows that $\mathcal{A}_{\Phi} \cup \mathcal{T} \cup \mathcal{P}$ is consistent, which implies that $\Phi \cup \mathcal{T} \cup \mathcal{P}$ is consistent.

Conversely, suppose that $\mathcal{T} \cup \mathcal{P} \cup \Phi$ is consistent, i.e. that such a theory has a model \mathcal{M} . This immediately implies that $\mathcal{M} \models \delta$ for each $\delta \in \mathcal{P}$, and thus $\mathcal{M} \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$. For the monotonicity of first-order logic, \mathcal{M} is a model of $\mathcal{T} \cup \Phi$ as well. Hence, since $\mathcal{M} \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$, we get that $\mathcal{T} \cup \Phi \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$. \square

We now turn our attention to the **CQ**-policy independence property (see Definition 6) and show that all $DL\text{-Lite}_{\mathcal{R}}$ and \mathcal{EL}_{\perp} CQE instances are **CQ**-policy independent. We will exploit this property in the next subsections to transfer from the CQA literature some lower bounds for entailment under ABox and **GA** censors.

Proposition 5. *Let \mathcal{E} be either a $DL\text{-Lite}_{\mathcal{R}}$ or an \mathcal{EL}_{\perp} CQE instance. We have that \mathcal{E} is **CQ**-policy independent.*

Proof. It is enough to prove that, given either a $DL\text{-Lite}_{\mathcal{R}}$ or \mathcal{EL}_{\perp} TBox \mathcal{T} , an ABox \mathcal{A} , and a policy \mathcal{P} , such that $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A}$ is consistent, then $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A} \models q$ if and only if $\mathcal{T} \cup \mathcal{A} \models q$, for each BCQ q .

Let \mathcal{T} , \mathcal{A} , and \mathcal{P} be as above and let q be a BCQ. If $\mathcal{T} \cup \mathcal{A} \models q$, then, due to the monotonicity of first-order logic, we trivially have that $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A} \models q$.

Suppose now that $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A} \models q$. Let \mathcal{U} be a universal model of $\mathcal{T} \cup \mathcal{A}$. Since $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A}$ is consistent by hypothesis, from Lemma 4 we have that $\mathcal{T} \cup \mathcal{A} \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$. As in the proof of Lemma 4, from the results in [26] it follows that $\mathcal{U} \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$ (i.e.

⁴ For a definition of universal model (also known as canonical model) we refer to [26].

$\mathcal{U} \models \mathcal{P}$). Since $\mathcal{U} \models \mathcal{T} \cup \mathcal{A}$ and since $\mathcal{U} \models \mathcal{P}$, we derive that \mathcal{U} is a model of $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A}$ as well. Since $\mathcal{T} \cup \mathcal{P} \cup \mathcal{A} \not\models q$ by assumption, we finally derive, by virtue of the properties of universal models, that $\mathcal{U} \models q$ as well, from which we have that $\mathcal{T} \cup \mathcal{A} \models q$. \square

5.1. ABox as censor language

We start our study by setting the censor language to the assertions in the ABox, and start by establishing lower bounds for the case of instance checking and the case of entailment of BCQ_∃s.

Theorem 4. *For any pair $(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$ such that $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}\}$, we have that $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$ is coNP-hard in data complexity.*

Proof. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be either a $DL\text{-Lite}_{\mathcal{R}}$ or a \mathcal{EL}_{\perp} CQE instance, and let ϕ be either a sentence in \mathbf{GA} or a sentence in \mathbf{CQ}_{\exists} . By combining Theorem 1 and Proposition 5, it follows that $\mathcal{E} \models_{\mathcal{A}}^{cqe} \phi$ if and only if $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle \models_{AR} \phi$.

Since AR-entailment is coNP-hard in data complexity for both $DL\text{-Lite}_{\mathcal{R}}$ and \mathcal{EL}_{\perp} ontologies even for instance checking, as respectively shown in [14, Theorem 3] and [17, Theorem 1], we immediately derive that $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathbf{GA})$ is coNP-hard in data complexity for both $\mathcal{L}_{\mathcal{T}} = DL\text{-Lite}_{\mathcal{R}}$ and $\mathcal{L}_{\mathcal{T}} = \mathcal{EL}_{\perp}$. Furthermore, in [11, Theorem 17] it is shown that AR-entailment is coNP-hard in data complexity for both $DL\text{-Lite}_{\mathcal{R}}$ and \mathcal{EL}_{\perp} ontologies for entailment of BCQ_∃s. Thus, we immediately derive that $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathbf{CQ}_{\exists})$ is coNP-hard in data complexity for both $\mathcal{L}_{\mathcal{T}} = DL\text{-Lite}_{\mathcal{R}}$ and $\mathcal{L}_{\mathcal{T}} = \mathcal{EL}_{\perp}$. \square

We now provide matching upper bounds, and show that they hold even for the case of entailment of BCQs. First, for $DL\text{-Lite}_{\mathcal{R}}$ (resp. \mathcal{EL}_{\perp}) CQE instances, the next lemma characterizes the entailment of BCQs under ABox censors in terms of standard entailment of BCQs in $DL\text{-Lite}_{\mathcal{R}}$ (resp. \mathcal{EL}_{\perp}) ontologies.

Lemma 5. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be either a $DL\text{-Lite}_{\mathcal{R}}$ or an \mathcal{EL}_{\perp} CQE instance and let q be a BCQ. Then, $\mathcal{E} \not\models_{\mathcal{A}}^{cqe} q$ if and only if there exists $\Gamma \subseteq \mathcal{A}$ satisfying the following three conditions: (i) $\mathcal{T} \cup \mathcal{P} \cup \Gamma$ is consistent, (ii) $\mathcal{T} \cup \mathcal{P} \cup \Gamma \cup \{\alpha\}$ is inconsistent for each $\alpha \in \mathcal{A} \setminus \Gamma$, and (iii) $\mathcal{T} \cup \Gamma \not\models q$.*

Proof. According to the definition of optimal censors in \mathcal{A} for \mathcal{E} (see Definition 2), it is straightforward to verify the following: given any $\Gamma \subseteq \mathcal{A}$, conditions (i) and (ii) are satisfied if and only if $\Gamma \in \text{OTH}_{\mathcal{S}_{\mathcal{A}}}(\mathcal{E})$.

Suppose that $\mathcal{E} \models_{\mathcal{A}}^{cqe} q$. Following Definition 3, we have that $\mathcal{T} \cup \text{Th} \models q$ for each $\text{Th} \in \text{OTH}_{\mathcal{S}_{\mathcal{A}}}(\mathcal{E})$. Thus, due to the above observation, no $\Gamma \subseteq \mathcal{A}$ can satisfy conditions (i), (ii), and (iii) simultaneously.

Conversely, suppose that there exists no $\Gamma \subseteq \mathcal{A}$ satisfying (i), (ii), and (iii). Due to the above observation, this means that there exists no $\Gamma \in \text{OTH}_{\mathcal{S}_{\mathcal{A}}}(\mathcal{E})$ such that $\mathcal{T} \cup \Gamma \not\models q$, i.e. it holds that $\mathcal{T} \cup \Gamma \models q$ for each $\Gamma \in \text{OTH}_{\mathcal{S}_{\mathcal{A}}}(\mathcal{E})$. Thus, by Definition 3, it follows that $\mathcal{E} \models_{\mathcal{A}}^{cqe} q$. \square

Notice that, due to Lemma 4, conditions (i) and (ii) in the above lemma can be equivalently reformulated in terms of standard entailment as follows: (i)' $\mathcal{T} \cup \Gamma \not\models q_{\delta}$ for each $\delta \in \mathcal{P}$ and (ii)' for any $\alpha \in \mathcal{A} \setminus \Gamma$, there is a $\delta \in \mathcal{P}$ such that $\mathcal{T} \cup \Gamma \cup \{\alpha\} \models q_{\delta}$. We are now ready to provide the matching upper bounds.

Theorem 5. *Both $ABox\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$ and $ABox\text{-Cens-Entailment}(\mathcal{EL}_{\perp}, \mathbf{CQ})$ are in coNP in data complexity.*

Proof. By Lemma 5, it directly follows that the problem of establishing whether $\mathcal{E} \not\models_{\mathcal{A}}^{cqe} q$, which is the complement of our problem, can be solved in non-deterministic polynomial time in data complexity by first guessing a subset Γ of the ABox \mathcal{A} and then checking conditions (i), (ii), and (iii). Since $\mathcal{T} \cup \mathcal{A}$ is a consistent $DL\text{-Lite}_{\mathcal{R}}$ (resp. \mathcal{EL}_{\perp}) ontology and since BCQ entailment in $DL\text{-Lite}_{\mathcal{R}}$ (resp. \mathcal{EL}_{\perp}) ontologies is in AC^0 (resp. in PTIME) in data complexity [18] (resp. [27]), all the three conditions can be verified in polynomial time with respect to the size of the ABox. This proves a coNP upper bound for both the problems $ABox\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$ and $ABox\text{-Cens-Entailment}(\mathcal{EL}_{\perp}, \mathbf{CQ})$. \square

Note that membership in coNP of $ABox\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$ also follows from Theorem 1, Proposition 5, and from the fact that AR-entailment of BCQs is in coNP for $DL\text{-Lite}_{\mathcal{R},den}$ ontologies, as shown in [14]. Obviously, Theorem 5 implies that $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$ is in coNP for $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}\}$.

With the above theorems, we can establish the precise data complexity of $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$, when $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\mathbf{CQ}, \mathbf{CQ}_{\exists}, \mathbf{GA}\}$.

Corollary 1. *For any pair $(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$ such that $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}, \mathbf{CQ}\}$, we have that $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$ is coNP-complete in data complexity.*

5.2. GA as censor language

We now consider the case in which the censor language coincides with **GA**.

First of all, from the above studied ABox-Cens-Entailment problems and by exploiting Proposition 4, we can immediately derive upper bounds for our GA-Cens-Entailment problems.

Theorem 6. *Both GA-Cens-Entailment(DL-Lite_R, CQ) and GA-Cens-Entailment(ℰL_⊥, CQ) are in coNP in data complexity.*

Proof. For $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{E}\mathcal{L}_{\perp}\}$, we provide a polynomial time reduction from $GA\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathbf{CQ})$ to $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathbf{CQ})$.

From Proposition 4 it follows that, given either a $DL\text{-Lite}_{\mathcal{R}}$ or an $\mathcal{E}\mathcal{L}_{\perp}$ CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ and a BCQ q , we can decide whether $\mathcal{E} \vDash_{\mathbf{GA}}^{cqe} q$ by first constructing a CQE instance $\mathcal{E}' = \langle \mathcal{T}, \mathcal{A}_{\mathcal{T}}, \mathcal{P} \rangle$ and then checking whether $\mathcal{E}' \vDash_{\mathcal{A}_{\mathcal{T}}}^{cqe} q$. Clearly, this reduction can be carried out in polynomial time because $\mathcal{A}_{\mathcal{T}}$ can be always computed in polynomial time for every ABox \mathcal{A} and TBox \mathcal{T} expressed either in $DL\text{-Lite}_{\mathcal{R}}$ or in $\mathcal{E}\mathcal{L}_{\perp}$. The claimed upper bound then follows from Theorem 5, establishing membership in coNP of $ABox\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathbf{CQ})$ for $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{E}\mathcal{L}_{\perp}\}$. \square

We note that the above results for $DL\text{-Lite}_{\mathcal{R}}$ also follow from Theorem 2 and from the fact that AR-entailment of BCQs is in coNP for $DL\text{-Lite}_{\mathcal{R}, den}$ ontologies [14]. Obviously, Theorem 6 implies that $GA\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathbf{Q})$ is in coNP for $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{E}\mathcal{L}_{\perp}\}$ and $\mathbf{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}\}$.

We now provide matching lower bounds for the $\mathcal{E}\mathcal{L}_{\perp}$ cases and for entailment of BCQ_∃s by $DL\text{-Lite}_{\mathcal{R}}$ CQE instances under **GA** censors. For the remaining $GA\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$, instead, we will prove an AC⁰ membership in data complexity of the problem.

We start with the $\mathcal{E}\mathcal{L}_{\perp}$ ontology language, by showing coNP-hardness already for instance checking and for entailment of BCQ_∃s.

We observe that [11, Theorem 29] shows that instance checking under the IAR semantics is coNP-hard for $\mathcal{E}\mathcal{L}_{\perp}$ ontologies. One might thus think that, in order to prove the same lower bound for $GA\text{-Cens-Entailment}(\mathcal{E}\mathcal{L}_{\perp}, \mathbf{GA})$, it is enough to make use of Theorem 3. However, [11, Theorem 29] cannot be directly used for our aims, because it is shown through a reduction that does not work if we replace the ABoxes with their deductive closure with respect to the fixed TBox.⁵

We thus provide a tailored proof, which uses a slight modification of the above mentioned reduction.

Theorem 7. *Both GA-Cens-Entailment(ℰL_⊥, GA) and GA-Cens-Entailment(ℰL_⊥, CQ_∃) are coNP-hard in data complexity.*

Proof. We start with the **GA** case and then provide a slight variation for the **CQ_∃** case. The proof is by a LOGSPACE-reduction from the complement of the satisfiability problem for propositional formulae in Negation Normal Form (NNF), a well-known NP-complete problem.

The fixed $\mathcal{E}\mathcal{L}_{\perp}$ TBox \mathcal{T} , policy \mathcal{P} , and ground atom $\gamma \in \mathbf{GA}$ are as follows:

$$\begin{aligned} \mathcal{T} = \{ & A_{\neg} \sqcap \exists R_1.F \sqsubseteq \exists R_T, \\ & A_{\vee} \sqcap \exists R_1.\exists R_T \sqsubseteq \exists R_T, \\ & A_{\vee} \sqcap \exists R_2.\exists R_T \sqsubseteq \exists R_T, \\ & A_{\wedge} \sqcap \exists R_1.\exists R_T \sqcap \exists R_2.\exists R_T \sqsubseteq \exists R_T \} \\ \mathcal{P} = \{ & \forall x, y(R_T(x, y) \wedge F(x) \rightarrow \perp), \\ & \forall x, y(R_T(x, y) \wedge B(x) \rightarrow \perp) \} \\ \gamma = & B(a_{\varphi}), \text{ where } a_{\varphi} \text{ is an individual.} \end{aligned}$$

Let φ be an NNF propositional formula over the set of propositional variables $\{v_1, \dots, v_n\}$. We now construct in LOGSPACE an ABox \mathcal{A}_{φ} exactly as done in the reduction provided in the proof of [11, Theorem 29]:

$$\begin{aligned} \mathcal{A}_{\varphi} = \{ & A_{\wedge}(a_{\psi}), R_1(a_{\psi}, a_{\chi_1}), R_2(a_{\psi}, a_{\chi_2}) \mid \psi = \chi_1 \wedge \chi_2 \text{ is a subformula of } \varphi \} \cup \\ & \{ A_{\vee}(a_{\psi}), R_1(a_{\psi}, a_{\chi_1}), R_2(a_{\psi}, a_{\chi_2}) \mid \psi = \chi_1 \vee \chi_2 \text{ is a subformula of } \varphi \} \cup \\ & \{ A_{\neg}(a_{\psi}), R_1(a_{\psi}, a_{\chi}) \mid \psi = \neg \chi \text{ is a subformula of } \varphi \} \cup \\ & \{ R_T(a_{v_i}, a'_{v_i}), F(a_{v_i}) \mid 1 \leq i \leq n \} \cup \{ B(a_{\varphi}) \}. \end{aligned}$$

As described in [11], every subformula ξ of φ (including subformulae corresponding to propositional variables) is represented by an individual a_{ξ} . In particular, the individual a_{φ} represents the entire formula φ . Given an ABox \mathcal{A}_{φ} , the first denial in \mathcal{P} forces every optimal censor for the $\mathcal{E}\mathcal{L}_{\perp}$ CQE instance $\mathcal{E}_{\varphi} = \langle \mathcal{T}, \mathcal{A}_{\varphi}, \mathcal{P} \rangle$ to choose between maintaining in its theory either $R_T(a_{v_i}, a'_{v_i})$ or $F(a_{v_i})$, for each individual a_{v_i} corresponding to a propositional variable v_i . Note that each such optimal censor represents a truth

⁵ More precisely, the ground atom $B(a_{\varphi})$ considered for instance checking in the proof of [11, Theorem 29] would always not belong to at least one repair, because the deductive closure of the ABox contains the ground atom $T(a_{\varphi})$ that is inconsistent with $B(a_{\varphi})$. Hence $B(a_{\varphi})$ would never be entailed, independently of the propositional formula.

value assignment to the variables in the formula φ , and that the TBox axioms propagate the truth value up to the whole formula φ . We also remark that, for the way in which the TBox \mathcal{T} is designed, every censor in \mathbf{GA} for \mathcal{E}_φ is a subset of the ABox \mathcal{A}_φ . Indeed, in this case the ground closure of \mathcal{A}_φ with respect to \mathcal{T} is equal to \mathcal{A}_φ .

Clearly, if the formula φ is satisfiable, then using the above argument, there is at least one optimal censor in \mathbf{GA} for \mathcal{E}_φ whose theory, together with the TBox, entails $\exists x(R_T(a_\varphi, x))$, thus discarding $\gamma = B(a_\varphi)$ for satisfying also the second denial in \mathcal{P} .

On the contrary, if the formula φ is unsatisfiable, then there is no assignment that makes true a subformula of φ that in turn implies the truth of φ . According to our construction, this means that there does not exist any optimal censor in \mathbf{GA} for \mathcal{E}_φ that together with the TBox entails $\exists x(R_T(a_\varphi, x))$. It follows that $\gamma = B(a_\varphi)$ belongs to the theory of every optimal censor in \mathbf{GA} for \mathcal{E}_φ . It follows that $\mathcal{E}_\varphi \vDash_{\mathbf{GA}}^{cqe} \gamma$ if and only if φ is unsatisfiable, as required.

We now address the \mathbf{CQ}_\exists case. Let \mathcal{T}' be the fixed \mathcal{EL}_\perp TBox obtained by extending \mathcal{T} with the assertion $B \sqsubseteq \exists R'$, i.e. $\mathcal{T}' = \mathcal{T} \cup \{B \sqsubseteq \exists R'\}$, and let q be the fixed \mathbf{BCQ}_\exists $q = \exists x, y(R'(x, y))$. Given the correctness of the above reduction, it is straightforward to verify that the following property holds: given any NNF propositional formula φ , we have that $\mathcal{E}'_\varphi \vDash_{\mathbf{GA}}^{cqe} q$ if and only if φ is unsatisfiable, where \mathcal{E}'_φ is the CQE instance $\mathcal{E}'_\varphi = \langle \mathcal{T}', \mathcal{A}_\varphi, \mathcal{P} \rangle$ with \mathcal{P} and \mathcal{A}_φ being, respectively, the fixed policy and the ABox constructible in LOGSPACE from φ as illustrated in the above instance checking case. \square

We now turn to the $DL\text{-Lite}_R$ ontology language. We start by providing coNP-hardness for entailment of \mathbf{BCQ}_\exists s.

Theorem 8. *GA-Cens-Entailment($DL\text{-Lite}_R$, \mathbf{CQ}_\exists) is coNP-hard in data complexity.*

Proof. In [11, Theorem 17], it is shown that there are \mathcal{T} , \mathcal{P} , and a \mathbf{BCQ}_\exists q , for which, given an ABox \mathcal{A} , checking whether $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle \vDash_{AR} \phi$ is coNP-hard, where $\mathcal{T} = \emptyset$ and \mathcal{P} is composed of a single $DL\text{-Lite}_R$ disjointness assertion (which can be clearly expressed as a denial assertion).

By combining again Theorem 1 and Proposition 5, given an ABox \mathcal{A} , it follows that $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle \vDash_{AR} q$ if and only if $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle \vDash_{\mathcal{A}}^{cqe} q$. Since in the mentioned reduction $\mathcal{T} = \emptyset$, we have $\mathcal{A}_\mathcal{T} = \mathcal{A}$, and therefore, due to Theorem 2, we can derive that $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle \vDash_{AR} q$ if and only if $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle \vDash_{\mathbf{GA}}^{cqe} q$, for each ABox \mathcal{A} .

It immediately follows that *GA-Cens-Entailment($DL\text{-Lite}_R$, \mathbf{CQ}_\exists)* is coNP-hard in data complexity. \square

Finally, we conclude this section by establishing that *GA-Cens-Entailment($DL\text{-Lite}_R$, \mathbf{GA})* is in \mathbf{AC}^0 in data complexity. We will do this by proving FO-rewritability of the problem, thus improving the upper bound given by Theorem 6.

Theorem 3 actually states that, to solve *GA-Cens-Entailment($DL\text{-Lite}_R$, \mathbf{GA})*, we can resort to the query rewriting techniques used to establish IAR-entailment given in [14], provided that we prevently compute the deductive closure of the input ABox with respect to the input TBox. We recall that GA entailment (in fact, BCQ entailment) under IAR-semantics in a DL \mathcal{L} is FO-rewritable, if for every TBox \mathcal{T} expressed in \mathcal{L} and every ground atom γ , one can effectively compute an FO query q_r such that for every ABox \mathcal{A} , $\langle \mathcal{T}, \mathcal{A} \rangle \vDash_{IAR} \gamma$ if and only if $\mathcal{I}_\mathcal{A} \vDash q_r$. The query q_r is called the *IAR-perfect reformulation* of γ with respect to \mathcal{T} .

To establish FO-rewritability of *GA-Cens-Entailment($DL\text{-Lite}_R$, \mathbf{GA})*, however, we still need to address the above mentioned computation of the mentioned deductive closure, and turn it into an additional query reformulation step. To this aim, we can exploit the fact that, for a $DL\text{-Lite}_{R,den}$ ontology $\langle \mathcal{T}, \mathcal{A} \rangle$, an FO query q evaluates to true over $\mathcal{A}_\mathcal{T}$ if and only if q' evaluates to true over \mathcal{A} , where q' is obtained by suitably rewriting each atom of q according to the positive inclusions of \mathcal{T} . Intuitively, in this way we cast into the query all the possible causes of the facts that are contained in the deductive closure of the ABox with respect to the TBox (similarly to what is done in query rewriting algorithms for $DL\text{-Lite}$ [18]).

To compute such a query q' , we use the function $\text{AtomRewr}(q, \mathcal{T})$, which substitutes each atom α occurring in query q with the formula $\lambda(\alpha)$ defined as follows (where A, B are atomic concepts and R, S are atomic roles):

$$\lambda(A(t)) = \bigvee_{\mathcal{T} \vDash B \sqsubseteq A} B(t) \vee \bigvee_{\mathcal{T} \vDash \exists R \sqsubseteq A} (\exists x(R(t, x))) \vee \bigvee_{\mathcal{T} \vDash \exists R \sqsubseteq A} (\exists x(R(x, t)))$$

$$\lambda(R(t_1, t_2)) = \bigvee_{\mathcal{T} \vDash S \sqsubseteq R} S(t_1, t_2) \vee \bigvee_{\mathcal{T} \vDash S \sqsubseteq R} S(t_2, t_1)$$

For example, if $\mathcal{T} = \{A \sqsubseteq C, B \sqsubseteq C\}$ and $q = \exists x, y(C(x) \wedge P(x, y))$, then $\text{AtomRewr}(q, \mathcal{T})$ returns the query $q = \exists x, y((C(x) \vee A(x) \vee B(x)) \wedge P(x, y))$.

The following lemma states the property we are looking for.

Lemma 6. *Let \mathcal{T} be a $DL\text{-Lite}_{R,den}$ TBox, \mathcal{A} be an ABox such that $\mathcal{T} \cup \mathcal{A}$ is consistent, and q be an FO sentence. We have that $\mathcal{I}_{\mathcal{A}_\mathcal{T}} \vDash q$ if and only if $\mathcal{I}_\mathcal{A} \vDash \text{AtomRewr}(q, \mathcal{T})$.*

Proof. Without loss of generality, we assume that q is in *prenex normal form* (note that every FO sentence is logically equivalent to some FO sentence in prenex normal form [28]), i.e. $q = Q_1 x_1 \dots Q_n x_n \Phi(x_1, \dots, x_n)$, where $\Phi(x_1, \dots, x_n)$ is a quantifier-free FO sentence and Q_i is either \forall or \exists , for each $i = 1, \dots, n$. To prove the claim, it is enough to show that for any assignment a to the variables $\vec{x} = (x_1, \dots, x_n)$ with constants occurring in the ABox \mathcal{A} the following holds: $\mathcal{I}_{\mathcal{A}_\mathcal{T}} \vDash \Phi_a$ if and only if $\mathcal{I}_\mathcal{A} \vDash \text{AtomRewr}(\Phi_a, \mathcal{T})$, where Φ_a denotes the formula obtained from $\Phi(x_1, \dots, x_n)$ by replacing the variable x_i with the constant $a(x_i)$, for each $i = 1, \dots, n$.

So, let a be any assignment as above. By construction of $\text{AtomRwr}(\cdot, \cdot)$, observe that checking whether $\mathcal{I}_{\mathcal{A}\mathcal{T}} \models \Phi_a$ if and only if $\mathcal{I}_{\mathcal{A}} \models \text{AtomRwr}(\Phi_a, \mathcal{T})$ can be equivalently reformulated as follows: $\mathcal{I}_{\mathcal{A}\mathcal{T}} \models \alpha$ if and only if $\mathcal{I}_{\mathcal{A}} \models (\lambda(\alpha), \mathcal{T})$, for each atom $\alpha \in \Phi_a$.

Now, if $\mathcal{I}_{\mathcal{A}} \models (\lambda(\alpha), \mathcal{T})$, then, by construction of $\lambda(\cdot)$, we have that α is entailed by $\mathcal{T} \cup \mathcal{A}$, and therefore $\mathcal{I}_{\mathcal{A}\mathcal{T}} \models \alpha$.

On the contrary, if $\mathcal{I}_{\mathcal{A}\mathcal{T}} \not\models \alpha$, then, by a trivial induction argument on the steps applied to derive α from $\mathcal{T} \cup \mathcal{A}$, it can be immediately verified that $\mathcal{I}_{\mathcal{A}} \not\models (\lambda(\alpha), \mathcal{T})$. \square

We are now able to establish FO-rewritability of $GA\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$.

Theorem 9. $GA\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$ is FO-rewritable, and therefore in AC^0 in data complexity.

Proof. Let \mathcal{T} be a $DL\text{-Lite}_{\mathcal{R}}$ TBox, let \mathcal{P} be a policy for \mathcal{T} , i.e. a set of denial assertions over the signature of \mathcal{T} such that $\mathcal{T} \cup \mathcal{P}$ is consistent, and let $\gamma \in \mathbf{GA}$. We now show how to obtain an FO query q' from \mathcal{T} , \mathcal{P} , and γ such that, for each ABox \mathcal{A} for which $\mathcal{T} \cup \mathcal{A}$ is consistent, we have that $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle \models_{\mathbf{GA}}^{cqe} \gamma$ if and only if $\mathcal{I}_{\mathcal{A}} \models q'$, thus proving FO-rewritability of the problem and consequently its membership in AC^0 .

Let the FO sentence q_r be an IAR-perfect reformulation of γ with respect to the $DL\text{-Lite}_{\mathcal{R},den}$ TBox $\mathcal{T} \cup \mathcal{P}$ [14]. Then, for every ABox \mathcal{A} , $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle \models_{IAR} \gamma$ if and only if $\mathcal{I}_{\mathcal{A}\mathcal{T}} \models q_r$. Now, from Lemma 6, it follows that, for every ABox \mathcal{A} , $\mathcal{I}_{\mathcal{A}\mathcal{T}} \models q_r$ if and only if $\mathcal{I}_{\mathcal{A}} \models \text{AtomRwr}(q_r, \mathcal{T})$. Since by Theorem 3, for every ABox \mathcal{A} such that $\langle \mathcal{T}, \mathcal{A} \rangle$ is consistent, $\langle \mathcal{T} \cup \mathcal{P}, \mathcal{A} \rangle \models_{IAR} \gamma$ if and only if $\mathcal{E} \models_{\mathbf{GA}}^{cqe} \gamma$, it follows that the FO sentence $\text{AtomRwr}(q_r, \mathcal{T})$ is such that $\mathcal{E} \models_{\mathbf{GA}}^{cqe} \gamma$ if and only if $\mathcal{I}_{\mathcal{A}} \models \text{AtomRwr}(q_r, \mathcal{T})$. This proves the FO-rewritability of $GA\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$, which, in turn, implies its membership in AC^0 . \square

The above theorem actually identifies a technique for computing the FO rewriting of a ground atom for the $GA\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$ problem, based on a simple combination of the IAR-perfect reformulation algorithm of [14] and the AtomRwr reformulation defined above.

We can now recall the data complexity of $GA\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$, where $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\mathbf{CQ}, \mathbf{CQ}_{\exists}, \mathbf{GA}\}$.

Corollary 2. $GA\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$ is in AC^0 in data complexity. The following problems are coNP-complete in data complexity:

- $GA\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathcal{Q})$, for every $\mathcal{Q} \in \{\mathbf{CQ}_{\exists}, \mathbf{CQ}\}$;
- $GA\text{-Cens-Entailment}(\mathcal{EL}_{\perp}, \mathcal{Q})$, for every $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}, \mathbf{CQ}\}$.

6. CQE under the \mathbf{CQ}_k censor language

In this section, we establish the data complexity of the problems $\mathbf{CQ}_k\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$, for every fixed integer $k \geq 1$, where $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}, \mathbf{CQ}\}$. More precisely, in Section 6.1 we provide some preliminary properties. In Section 6.2 we prove that entailment of both BCQs and \mathbf{BCQ}_{\exists} s under \mathbf{CQ}_k censors is coNP-complete, for both $DL\text{-Lite}_{\mathcal{R}}$ and \mathcal{EL}_{\perp} CQE instances (the upper bound is given in Theorem 10, the lower bounds are shown in Theorem 11 and Theorem 12). Finally, in Section 6.3 we prove that instance checking under \mathbf{CQ}_k censors for $DL\text{-Lite}_{\mathcal{R}}$ CQE instances is actually tractable, and more precisely in AC^0 in data complexity (Theorem 13). In particular, based on a crucial property provided by Proposition 6, we define a query rewriting technique (Definition 8) that is able to reduce the above instance checking problem to the evaluation of a first-order sentence on the ABox of the CQE instance.

6.1. Preliminary properties

Given a BCQ q of the form $\exists \vec{x}(\alpha_1 \wedge \dots \wedge \alpha_n)$, a *subquery* of q is a BCQ q' of the form $\exists \vec{x}'(\alpha_{i_1} \wedge \dots \wedge \alpha_{i_m})$ such that, for every $j \in \{1, \dots, m\}$, $1 \leq i_j \leq n$, and \vec{x}' are the variables of \vec{x} that occur in some α_{i_j} . Informally, the subquery q' is obtained from q by deleting some of its atoms.

Given a policy \mathcal{P} , we denote by $\text{maxlen}(\mathcal{P})$ the maximum length of a denial in \mathcal{P} . We also denote by $\text{Freeze}(q)$ the ABox obtained from the BCQ q by replacing each variable x with a new individual a_x (i.e. an individual not occurring in \mathcal{P}) and by treating the obtained conjunction of ground atoms as a set of facts, in the obvious way. In the following, we also consider sets of BCQs. For every such set Φ we always assume that different queries in Φ use different variable symbols. Moreover, we use $\text{Freeze}(\Phi)$ to indicate the ABox $\bigcup_{q \in \Phi} \text{Freeze}(q)$. We call *fresh* individuals of $\text{Freeze}(\Phi)$ the new individuals introduced in $\text{Freeze}(\Phi)$. We finally recall that, given a denial δ , q_{δ} indicates the BCQ corresponding to the negation of δ .

The following lemma, easy to verify, shows how the freezing of a set of queries Φ can be used for checking whether $\mathcal{T} \cup \Phi$ entails a BCQ.

Lemma 7. Let \mathcal{T} be either a $DL\text{-Lite}_{\mathcal{R}}$ or an \mathcal{EL}_{\perp} TBox, let Φ be a set of BCQs, and let q be a BCQ that does not mention the fresh individuals of $\text{Freeze}(\Phi)$. Then, $\mathcal{T} \cup \Phi \models q$ if and only if $\mathcal{T} \cup \text{Freeze}(\Phi) \models q$.

The proof of the next lemma immediately follows from Lemma 4 and Lemma 7.

Lemma 8. Let \mathcal{T} be either a $DL\text{-Lite}_{\mathcal{R}}$ or an \mathcal{EL}_{\perp} TBox, let \mathcal{P} be a policy such that $\mathcal{T} \cup \mathcal{P}$ is consistent, and let Φ be a set of BCQs such that $\mathcal{T} \cup \Phi$ is consistent. We have that $\Phi \cup \mathcal{T} \cup \mathcal{P}$ is consistent if and only if $\mathcal{T} \cup \text{Freeze}(\Phi) \not\models q_{\delta}$ for every $\delta \in \mathcal{P}$.

Hereinafter, we consider two BCQs that are equal up to variable renaming as the same BCQ: in this way, for every integer $k \geq 1$, $\mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A})$ is finite and can be computed in polynomial time with respect to the size of the ABox \mathcal{A} . We also recall that $\mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$ denotes the set $\{q \in \mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A}) \mid \mathcal{T} \cup \mathcal{A} \models q\}$.

Similarly to what Lemma 5 does for ABox censors, the following lemma characterizes the entailment of BCQs under \mathbf{CQ}_k censors ($k \geq 1$) in terms of standard entailment of BCQs.

Lemma 9. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be either a $DL\text{-Lite}_{\mathcal{R}}$ or an \mathcal{EL}_{\perp} CQE instance, q be a BCQ, and $k \geq 1$ be an integer. Then, $\mathcal{E} \not\models_{\mathbf{CQ}_k}^{cqc} q$ if and only if there exists a set $\Phi \subseteq \mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$ satisfying the following three conditions: (i) $\mathcal{T} \cup \mathcal{P} \cup \Phi$ is consistent, (ii) $\mathcal{T} \cup \mathcal{P} \cup \Phi \cup \{q\}$ is inconsistent for each $\phi \in \mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A}) \setminus \Phi$, and (iii) $\mathcal{T} \cup \Phi \not\models q$.

Proof. The proof can be obtained immediately by the same reasoning as in the proof Lemma 5 with the following observation, which is trivial to verify: given any $\Phi \subseteq \mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$, conditions (i) and (ii) are satisfied if and only if Φ is equivalent to the theory of an optimal censor in \mathbf{CQ}_k for \mathcal{E} . \square

In the following, given a finite, non-empty set of BCQs $\Phi = \{q_1, \dots, q_n\}$ such that $q_i = \exists \vec{x}_i(cq_i(\vec{x}_i))$ for every $i \in \{1, \dots, n\}$, we denote by $\text{Conj}(\Phi)$ the BCQ corresponding to the conjunction of all the BCQs in Φ , i.e.:

$$\text{Conj}(\Phi) = \exists \vec{x}_1, \dots, \vec{x}_n \left(\bigwedge_{i=1}^n cq_i(\vec{x}_i) \right).$$

Notice that $\text{Conj}(\Phi)$ is a sentence equivalent to Φ .

Example 6. Consider the following set of BCQs: $\Phi = \{\exists x(C(x) \wedge D(x)), \exists y, z(R(y, z) \wedge D(z))\}$. We have that $\text{Conj}(\Phi) = \exists x, y, z(C(x) \wedge D(x) \wedge R(y, z) \wedge D(z))$. \square

A *connected component* of a BCQ q is a BCQ q' such that q' is a subquery of q , and every atom in q but not in q' does not contain any occurrence of the variables of q' . We say that a set of BCQs Φ is a *decomposition* of a BCQ q if every $q' \in \Phi$ is a connected component of q and Φ is equivalent to q . For instance, in Example 6 Φ is a decomposition of the query $\text{Conj}(\Phi)$. Moreover, a *partial instantiation* of a BCQ q is a BCQ obtained from q by replacing some of its variables with constants.

We then say that a set of BCQs Φ is *closed under subqueries* if, for every $q \in \Phi$ and for every subquery q' of q , we have $q' \in \Phi$.

The following property holds for $DL\text{-Lite}_{\mathcal{R}}$ TBoxes.

Lemma 10. Let \mathcal{T} be a $DL\text{-Lite}_{\mathcal{R}}$ TBox, let q be a BCQ, and let Φ be a set of BCQs closed under subqueries, such that $\mathcal{T} \cup \Phi$ is consistent. Then, $\mathcal{T} \cup \Phi \models q$ if and only if there exists a finite subset Φ' of Φ such that $\mathcal{T} \cup \Phi' \models q$ and $\text{length}(\text{Conj}(\Phi')) \leq \text{length}(q)$.

Proof. The proof can be easily obtained by extending the proof of the same, well-known property in the case when Φ is an ABox (see e.g., [29]). \square

6.2. Intractability results

We are now ready to provide a coNP upper bound for the decision problems of interest in this section.

Theorem 10. Let k be any integer such that $k \geq 1$. We have that both the decision problems $CQ_k\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$ and $CQ_k\text{-Cens-Entailment}(\mathcal{EL}_{\perp}, \mathbf{CQ})$ are in coNP in data complexity.

Proof. The proof exploits Lemma 9 in a similar way as the proof of Theorem 5 exploits Lemma 5. More specifically, due to Lemma 9, the problem of checking whether $\mathcal{E} \not\models_{\mathbf{CQ}_k}^{cqc} q$ can be solved in non-deterministic polynomial time in data complexity by first computing $\mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$ in polynomial time with respect to the size of the ABox \mathcal{A} , then by guessing a subset Φ of $\mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$, and finally by checking conditions (i), (ii), and (iii). Since we are dealing either with the $DL\text{-Lite}_{\mathcal{R}}$ ontology language or the \mathcal{EL}_{\perp} ontology language, all three conditions can be verified in polynomial time with respect to the size of the ABox \mathcal{A} . This shows that the complement of our problem is in NP, from which the claim follows. \square

We now give matching lower bounds for both the decision problems $CQ_k\text{-Cens-Entailment}(\mathcal{EL}_{\perp}, \mathbf{GA})$ and $CQ_k\text{-Cens-Entailment}(\mathcal{EL}_{\perp}, \mathbf{CQ}_{\exists})$, as well as for the decision problem $CQ_k\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ}_{\exists})$, for every fixed integer $k \geq 1$. Of course, these results imply that the problem is coNP-hard even when the query language is \mathbf{CQ} , for both $DL\text{-Lite}_{\mathcal{R}}$ and \mathcal{EL}_{\perp} CQE instances. We start with the mentioned lower bounds for \mathcal{EL}_{\perp} CQE instances.

Theorem 11. For every integer $k \geq 1$, we have that both CQ_k -Cens-Entailment(\mathcal{EL}_\perp , **GA**) and CQ_k -Cens-Entailment(\mathcal{EL}_\perp , **CQ₃**) are coNP-hard in data complexity.

Proof. Let k be any integer such that $k \geq 1$. We now show that both CQ_k -Cens-Entailment(\mathcal{EL}_\perp , **GA**) and CQ_k -Cens-Entailment(\mathcal{EL}_\perp , **CQ₃**) are coNP-hard in data complexity.

We start with the **GA** case and then provide a slight variation for the **CQ₃** case. The proof can be obtained by extending the proof of Theorem 7. Specifically, recall the fixed \mathcal{EL}_\perp TBox \mathcal{T} , policy \mathcal{P} , and ground atom $\gamma = B(a_\varphi)$ defined there. We define the fixed \mathcal{EL}_\perp TBox \mathcal{T}_k and policy \mathcal{P}_k as follows:

- \mathcal{T}_k is obtained from \mathcal{T} by replacing each occurrence of the concept $\exists R_T$ with the concept $\exists R_T.\exists P_1 \dots \exists P_k$, where P_1, \dots, P_k are fresh atomic roles in Σ_R ;
- \mathcal{P}_k is obtained from \mathcal{P} by replacing the second denial assertion $\forall x, y(R_T(x, y) \wedge B(x) \rightarrow \perp)$ with the following denial assertion:
 $\forall x, y, z_1, \dots, z_k(R_T(x, y) \wedge P_1(y, z_1) \wedge \dots \wedge P_k(z_{k-1}, z_k) \wedge B(x) \rightarrow \perp)$.

Finally, given any NNF propositional formula φ , we construct in LOGSPACE exactly the same ABox \mathcal{A}_φ illustrated in the proof of Theorem 7.

Importantly, notice that, for every subformula ξ of φ , we cannot have a sentence s of the form $s = \exists y, z_1, \dots, z_k(R_T(a_\xi, y) \wedge P_1(y, z_1) \wedge P_2(z_1, z_2) \wedge \dots \wedge P_k(z_{k-1}, z_k))$ in the theory of a censor in **CQ_k** for $\mathcal{E}_{k,\varphi} = \langle \mathcal{T}_k, \mathcal{A}_\varphi, \mathcal{P}_k \rangle$, since s is a sentence of length $k + 1$. Based on this above observation, and by using analogous arguments as those provided in the proof of Theorem 7 of the correctness of the reduction, one can easily verify that, for any NNF propositional formula φ , we have that φ is satisfiable if and only if $\mathcal{E}_{k,\varphi} \stackrel{cqe}{\not\models} \mathbf{CQ}_k \gamma$, as required.

The variation for the **CQ₃** case is similar to the variation provided at the end of the proof of Theorem 7. Specifically, let \mathcal{T}'_k be the fixed \mathcal{EL}_\perp TBox obtained by extending \mathcal{T} with the assertion $B \sqsubseteq \exists R'.\exists P_1 \dots \exists P_k$, i.e. $\mathcal{T}' = \mathcal{T} \cup \{B \sqsubseteq \exists R'.\exists P_1 \dots \exists P_k\}$, and let q_k be the fixed BCQ₃ $q_k = \exists x, y, z_1, \dots, z_k(R'(x, y) \wedge P_1(y, z_1) \wedge P_2(z_1, z_2) \wedge \dots \wedge P_k(z_{k-1}, z_k))$. Given the correctness of the above reduction, it is straightforward to verify that the following property holds: given any NNF propositional formula φ , we have that $\mathcal{E}'_{k,\varphi} \stackrel{cqe}{\not\models} \mathbf{CQ}_k q_k$ if and only if φ is unsatisfiable, where $\mathcal{E}'_{k,\varphi}$ is the CQE instance $\mathcal{E}'_{k,\varphi} = \langle \mathcal{T}'_k, \mathcal{A}_\varphi, \mathcal{P}_k \rangle$ with \mathcal{P}_k and \mathcal{A}_φ being, respectively, the fixed policy and the ABox constructible in LOGSPACE from φ as illustrated in the above instance checking case. \square

We now turn our attention to $DL\text{-Lite}_{\mathcal{R}}$, and, as announced, provide a matching lower bound for entailment of BCQ₃s under **CQ_k** censors.

Theorem 12. CQ_k -Cens-Entailment($DL\text{-Lite}_{\mathcal{R}}$, **CQ₃**) is coNP-hard in data complexity, for every integer $k \geq 1$.

Proof. Let k be any integer such that $k \geq 1$. We now show that CQ_k -Cens-Entailment($DL\text{-Lite}_{\mathcal{R}}$, **CQ₃**) is coNP-hard in data complexity. The proof is by a LOGSPACE reduction from the complement of 3-SAT, and it can be seen as an adaptation of the proof provided in [14, Theorem 3]. The fixed $DL\text{-Lite}_{\mathcal{R}}$ TBox, policy \mathcal{P} , and BCQ₃ q_k are as follows:

$\mathcal{T} = \emptyset$, i.e. the TBox \mathcal{T} contains no assertions;

$\mathcal{P} = \{ \forall x, y, z(P(x, z) \wedge N(y, z) \rightarrow \perp),$
 $\forall x, y, z(P(x, y) \wedge A(x) \rightarrow \perp),$
 $\forall x, y, z(N(x, y) \wedge A(x) \rightarrow \perp) \};$

$q_k = \exists y(A(y) \wedge A_1(y) \wedge \dots \wedge A_k(y)).$

Let $\varphi = c_1 \wedge \dots \wedge c_m$ be a 3-SAT instance with $c_i = l_i^1 \vee l_i^2 \vee l_i^3$, where each literal l_i^j is either a positive or a negative variable from a set $\{v_1, \dots, v_n\}$. We say that a variable v occurs positively (resp. negatively) in a clause c_i if $l_i^j = v$ (resp. $l_i^j = \neg v$) for some $j = [1, 3]$. We now construct in LOGSPACE an ABox \mathcal{A}_φ^k similarly to as done in the reduction provided in the proof of [14, Theorem 3]:

$$\mathcal{A}_\varphi^k = \{A(c_i) \mid 1 \leq i \leq m\} \cup \bigcup_{i=1}^m \{A_j(c_i) \mid 1 \leq j \leq k\} \cup$$

$$\bigcup_{i=1}^m \{P(c_i, v) \mid v \text{ occurs positively in } c_i\} \cup$$

$$\bigcup_{i=1}^m \{N(c_i, v) \mid v \text{ occurs negatively in } c_i\}.$$

Intuitively, the first denial in \mathcal{P} forces every optimal censor in **CQ_k** for the $DL\text{-Lite}_{\mathcal{R}}$ CQE instance $\mathcal{E}_\varphi^k = \langle \mathcal{T}, \mathcal{A}_\varphi^k, \mathcal{P} \rangle$ to choose between maintaining in its theory either facts of the form $P(c_i, v)$ or facts of the form $N(c_i, v)$, for each possible variable v involved in the 3-SAT instance φ . Note that each such optimal censor represents a truth value assignment to the variables in the instance. Furthermore, we point out that we cannot have a sentence s of the form $s = \exists y(A(y) \wedge A_1(y) \wedge \dots \wedge A_k(y))$ in the theory of a censor in **CQ_k** for \mathcal{E}_φ^k , since s is a sentence of length $k + 1$.

Suppose that φ is unsatisfiable, i.e. for every truth value assigned to the variables in φ there will be at least one clause c_i not satisfied under such an assignment. It follows that in every theory Φ of an optimal censor in **CQ_k** for \mathcal{E}_φ^k there is at least one (constant representing the clause) c_i such that there are no occurrences of the form $P(c_i, v)$ or $N(c_i, v)$ in Φ , implying that $A(c_i) \in \Phi$. As a

consequence, due to the fact that, for every $i \in [1, m]$ and for every $j \in [1, k]$, we trivially have $A_j(c_i) \in \Phi$ for every $\Phi \in \text{OThS}_{\mathbf{CQ}_k}(\mathcal{E}_\varphi^k)$, we immediately get that $\Phi \models q_k$ for every $\Phi \in \text{OThS}_{\mathbf{CQ}_k}(\mathcal{E}_\varphi^k)$, and therefore $\mathcal{E}_\varphi^k \models_{\mathbf{CQ}_k}^{cqe} q_k$.

Suppose now that φ is satisfiable. It follows that there exists at least one theory $\Phi \in \text{OThS}_{\mathbf{CQ}_k}(\mathcal{E}_\varphi^k)$ (corresponding to the assignment that satisfies φ) such that: for every $i \in [1, m]$, there exists a v such that either $P(c_i, v) \in \Phi$ or $N(c_i, v) \in \Phi$ holds. As a consequence, due to the second and third denials in \mathcal{P} , we get that $A(c_i) \notin \Phi$ holds for every $i \in [1, m]$. Thus, we derive that $\Phi \not\models q_k$, and therefore $\mathcal{E}_\varphi^k \not\models_{\mathbf{CQ}_k}^{cqe} q_k$. \square

6.3. A tractable case

We now concentrate on the instance checking case and show that, for every integer $k \geq 1$, \mathbf{CQ}_k -Cens-Entailment(DL-Lite_R , \mathbf{GA}) is in AC^0 in data complexity, thus improving the upper bound given in Theorem 10. We do so by proving FO-rewritability of the problem.

The following crucial property is at the basis of the rewriting technique that we describe immediately after. Intuitively, the conditions listed in the next proposition say that entailment under \mathbf{CQ}_k sensors of a ground atom α holds if and only if (a) α is entailed by the CQE instance without considering the policy; (b) α can be disclosed without violating confidentiality rules encoded through to the policy; (c) there is no \mathbf{CQ}_k optimal censor C , denial $\delta \in \mathcal{P}$, and subquery q'_δ of q_δ , with $\text{length}(q'_\delta) \leq \text{length}(q_\delta) - 1$, having an image on $\text{Th}(C)$ (conditions (c1) and (c2)) that can be “extended” into an image of q_δ by using α (condition (c3)), and thus causing that $\mathcal{T} \cup \text{Th}(C) \cup \{\alpha\}$ violates δ . Notice that q'_δ might not belong to $\text{Th}(C)$ or to any \mathbf{CQ}_k censor, because k might be smaller than $\text{length}(q'_\delta)$.

Proposition 6. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a DL-Lite_R CQE instance, let α be a ground atom, let $h = \max\text{len}(\mathcal{P}) - 1$, and let $k \geq 1$ be an integer. Then, $\mathcal{E} \models_{\mathbf{CQ}_k}^{cqe} \alpha$ if and only if the following conditions hold:

- (a) $\mathcal{T} \cup \mathcal{A} \models \alpha$;
- (b) $\{\alpha\} \cup \mathcal{T} \cup \mathcal{P}$ is consistent;
- (c) there exists no BCQ $q' \in \mathbf{CQ}_h(\{\alpha\} \cup \mathcal{T} \cup \mathcal{P})$ such that the following conditions hold:
 - (c1) there exists a partial instantiation q'' of q' with constants of \mathcal{A} not occurring in $\{\alpha\} \cup \mathcal{P}$ and a set $\Phi \subseteq \mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$ such that Φ is a decomposition of q'' ;
 - (c2) $\{q'\} \cup \mathcal{T} \cup \mathcal{P}$ is consistent;
 - (c3) $\{\alpha, q'\} \cup \mathcal{T} \cup \mathcal{P}$ is inconsistent.

Proof. First, it is easy to verify that $\mathcal{E} \models_{\mathbf{CQ}_k}^{cqe} \alpha$ if and only if condition (a) holds, condition (b) holds, and the following condition (c') holds: there exists no $\Phi' \subseteq \mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$ such that $\Phi' \cup \mathcal{T} \cup \mathcal{P}$ is consistent and $\{\alpha\} \cup \Phi' \cup \mathcal{T} \cup \mathcal{P}$ is inconsistent (the key point is that such a set Φ' would be certainly contained in the theory of some optimal censor in \mathbf{CQ}_k for \mathcal{E} that is inconsistent with α).

Then, since q'' is obtained from q' replacing some of its variables with constants not occurring in $\{\alpha\} \cup \mathcal{P}$, and since q'' is equivalent to the set Φ , it immediately follows that condition (c2) holds iff $\Phi \cup \mathcal{T} \cup \mathcal{P}$ is consistent, and condition (c3) holds iff $\{\alpha\} \cup \Phi \cup \mathcal{T} \cup \mathcal{P}$ is inconsistent.

This immediately implies that, if condition (a) does not hold, or condition (b) does not hold, or condition (c) does not hold, then $\mathcal{E} \not\models_{\mathbf{CQ}_k}^{cqe} \alpha$.

Now suppose $\mathcal{E} \not\models_{\mathbf{CQ}_k}^{cqe} \alpha$. There are three possible cases: (i) condition (a) does not hold, which implies the thesis; (ii) condition (b) does not hold, which implies the thesis; (iii) condition (c') does not hold. In the latter case, there exists a subset Φ' of $\mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$ such that $\Phi' \cup \mathcal{T} \cup \mathcal{P}$ is consistent and $\{\alpha\} \cup \Phi' \cup \mathcal{T} \cup \mathcal{P}$ is inconsistent. W.l.o.g. we can assume that Φ' is closed under subqueries. Now, since $\mathcal{T} \cup \Phi'$ is consistent, by Lemma 8 inconsistency w.r.t. \mathcal{P} corresponds to the entailment in $\mathcal{T} \cup \Phi'$ of a query q_δ for some denial $\delta \in \mathcal{P}$; moreover, from Lemma 10 and from the fact that $\Phi' \cup \mathcal{T} \cup \mathcal{P}$ is consistent and $\{\alpha\} \cup \Phi' \cup \mathcal{T} \cup \mathcal{P}$ is inconsistent, the existence of the set Φ' implies the existence of a subset Φ of Φ' such that $\Phi \cup \mathcal{T} \models q_\delta$ and $\text{length}(\text{Conj}(\Phi)) \leq \max\text{len}(\mathcal{P}) - 1$. Now let q' be the BCQ obtained from $\text{Conj}(\Phi)$ replacing every constant not occurring in $\{\alpha\} \cup \mathcal{P}$ with a new existential variable. Then, $q' \in \mathbf{CQ}_h(\{\alpha\} \cup \mathcal{T} \cup \mathcal{P})$. And since $\Phi \cup \mathcal{T} \models q_\delta$, it follows that condition (c3) holds for q' . Moreover, since $\Phi \subseteq \Phi'$ and $\Phi' \cup \mathcal{T} \cup \mathcal{P}$ is consistent, it follows that $\Phi \cup \mathcal{T} \cup \mathcal{P}$ is consistent too, which immediately implies that condition (c2) holds for q' . Finally, since $\Phi' \subseteq \mathbf{CQ}_k^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$ and $\Phi \subseteq \Phi'$, condition (c1) holds for Φ , thus proving the thesis. \square

Example 7. Let us consider a DL-Lite_R CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ such that $\mathcal{T} = \emptyset$, $\mathcal{A} = \{A(a), B(a), C(b)\}$, and $\mathcal{P} = \{\forall x(A(x) \wedge B(x) \wedge C(b) \rightarrow \perp)\}$. Then, let us fix $k = 1$ and check entailment of $\alpha = C(b)$ under \mathbf{CQ}_1 sensors. The set $\mathbf{CQ}_1^{\text{Ent}}(\mathcal{T} \cup \mathcal{A})$ consists of the ground atoms $A(a), B(a), C(b)$ and all BCQs of length 1 implied by such atoms. It is easy to see that there are three optimal \mathbf{CQ}_1 sensors, i.e.,

$$\begin{aligned} C_1 &= \{A(a), B(a), \exists x A(x), \exists x B(x), \exists x C(x)\} \\ C_2 &= \{A(a), C(b), \exists x A(x), \exists x B(x), \exists x C(x)\} \\ C_3 &= \{B(a), C(b), \exists x A(x), \exists x B(x), \exists x C(x)\} \end{aligned}$$

In this case, α is not entailed by \mathcal{E} under \mathbf{CQ}_1 censors. With respect to Proposition 6, we can see that conditions (a) and (b) are satisfied, but conditions (c) does not hold. To verify this latter point, consider $q' = \exists x(A(x) \wedge B(x))$. An instantiation of q' not using constants in $\alpha \cup \mathcal{P}$ is $q'' = A(a) \wedge B(a)$, and a decomposition of q'' contained in $\mathbf{CQ}_1^{Ent}(\mathcal{T} \cup \mathcal{A})$ is $\Phi = \{A(a), B(a)\}$. It is easy to see that conditions c1 – c3 are fulfilled. \square

We are now ready to provide a technique to obtain a FO rewriting for instance checking under \mathbf{CQ}_k censors over $DL\text{-Lite}_{\mathcal{R}}$ CQE instances.

In the following, given a BCQ q' and a positive integer k , we denote by $DVS_k(q')$ (k -decomposing variable sets of q') the set of every minimal subset X of variables of q' such that, if q'' is any BCQ obtained from q' instantiating the variables in X , then q'' admits a decomposition Φ such that $\Phi \in \mathbf{CQ}_k$. For instance, if $q' = \exists x, y, z(R(x, y) \wedge D(y) \wedge S(y, z) \wedge E(z))$, then $DVS_3(q') = \{\{y\}, \{z\}\}$, $DVS_2(q') = \{\{y\}\}$, $DVS_1(q') = \{\{y, z\}\}$.

Definition 8. Given a $DL\text{-Lite}_{\mathcal{R}}$ TBox \mathcal{T} , a policy \mathcal{P} , and a ground atom α , we define the sentence $CQ_k\text{CensEntailed}(\alpha, \mathcal{T}, \mathcal{P})$ as follows:

$$\begin{aligned} & \text{PerfectRef}(\alpha, \mathcal{T}) \wedge \text{Consistent}(\{\alpha\}, \mathcal{T}, \mathcal{P}) \wedge \\ & \bigwedge_{q' \in \mathbf{CQ}_h(\{\alpha\} \cup \mathcal{P} \cup \mathcal{T})} \left(\left(\bigwedge_{X \in DVS_k(q')} \neg \text{PerfectRef}(\text{Conj}^{\neq}(q', X, \alpha, \mathcal{P}), \mathcal{T}) \right) \vee \right. \\ & \qquad \qquad \qquad \neg \text{Consistent}(\{q'\}, \mathcal{T}, \mathcal{P}) \vee \\ & \qquad \qquad \qquad \left. \text{Consistent}(\{\alpha, q'\}, \mathcal{T}, \mathcal{P}) \right) \end{aligned}$$

where:

- $\text{PerfectRef}(q, \mathcal{T})$ is the perfect reformulation of q with respect to \mathcal{T} (see also Section 2);
- $\text{Consistent}(\Phi, \mathcal{T}, \mathcal{P})$ (where Φ is a set of BCQs) is the sentence *true* if $\Phi \cup \mathcal{T} \cup \mathcal{P}$ is consistent, and is the sentence *false* otherwise;⁶
- $h = \text{maxlen}(\mathcal{P}) - 1$;
- $\text{Conj}^{\neq}(q', X, \alpha, \mathcal{P})$ returns a BCQ with inequalities that is obtained by adding to the BCQ q' the conjunction of all the inequality atoms of the form $x \neq c$ for every variable $x \in X$ and for every constant occurring in $\{\alpha\} \cup \mathcal{P}$. For instance, if $q' = \exists x, y, z(R(x, y) \wedge D(y) \wedge S(y, z))$, $X = \{y\}$, and the constants occurring in $\{\alpha\} \cup \mathcal{P}$ are c and d , then

$$\text{Conj}^{\neq}(q', X, \alpha, \mathcal{P}) = \exists x, y, z(R(x, y) \wedge D(y) \wedge S(y, z) \wedge y \neq c \wedge y \neq d);$$

- $\text{PerfectRef}(\text{Conj}^{\neq}(q', X, \alpha, \mathcal{P}), \mathcal{T})$ considers the BCQ with inequalities $\text{Conj}^{\neq}(q', X, \alpha, \mathcal{P})$ as a standard BCQ without inequalities, managing inequality atoms as standard role atoms.

The following proposition states that the sentence $CQ_k\text{CensEntailed}(q, \mathcal{T}, \mathcal{P})$ can be used to decide $CQ_k\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$.

Proposition 7. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a $DL\text{-Lite}_{\mathcal{R}}$ CQE instance and let α be a ground atom. Then, $\mathcal{E} \models_{\mathbf{CQ}_k}^{cqe} \alpha$ if and only if $I_{\mathcal{A}} \models CQ_k\text{CensEntailed}(\alpha, \mathcal{T}, \mathcal{P})$.

Proof. The proof is an immediate consequence of Proposition 6 and of the following facts:

- for a consistent $DL\text{-Lite}_{\mathcal{R}}$ ontology $\mathcal{T} \cup \mathcal{A}$, we have that $I_{\mathcal{A}} \models \text{PerfectRef}(\alpha, \mathcal{T})$ if and only if $\mathcal{T} \cup \mathcal{A} \models \alpha$ (see Condition (a) in Proposition 1);
- $\text{Consistent}(\{\alpha\}, \mathcal{T}, \mathcal{P})$ captures Condition (b) of Proposition 1;
- $\text{PerfectRef}(\text{Conj}^{\neq}(q', X, \alpha, \mathcal{P}), \mathcal{T})$ evaluates to true in $I_{\mathcal{A}}$ if and only if there exist a partial instantiation q'' of q' with constants not occurring in $\{\alpha\} \cup \mathcal{P}$ and a set $\Phi \subseteq \mathbf{CQ}_k^{Ent}(\mathcal{T} \cup \mathcal{A})$ such that Φ is a decomposition of q'' (see Condition (c1) of Proposition 1);
- $\neg \text{Consistent}(\{q'\}, \mathcal{T}, \mathcal{P})$ captures Condition (c2) of Proposition 1 (note that sentence $CQ_k\text{CensEntailed}(\alpha, \mathcal{T}, \mathcal{P})$, modulo the conjunct $\neg \text{PerfectRef}(\text{Conj}^{\neq}(q', X, \alpha, \mathcal{P}), \mathcal{T})$, is in CNF);
- $\text{Consistent}(\{\alpha, q'\}, \mathcal{T}, \mathcal{P})$ captures Condition (c3) of Proposition 1. \square

Example 8. Let us consider again Example 7. Let $k = 1$, $h = 2$, $q' = \exists x(A(x) \wedge B(x))$; then, the set DVS_1 is $\{\{x\}\}$. Now, it is immediate to verify that:

$$\begin{aligned} & \text{PerfectRef}(\alpha, \mathcal{T}) = C(b); \\ & \text{Consistent}(\{\alpha\}, \mathcal{T}, \mathcal{P}) = \text{true}; \\ & \neg \text{PerfectRef}(\text{Conj}^{\neq}(q', X, \alpha, \mathcal{P}), \mathcal{T}) = \exists x(A(x) \wedge B(x) \wedge x \neq b); \end{aligned}$$

⁶ Notice that the consistency of $\Phi \cup \mathcal{T} \cup \mathcal{P}$ is equivalent to checking the consistency of the $DL\text{-Lite}_{\mathcal{R}, den}$ ontology $\text{Freeze}(\Phi) \cup \mathcal{T} \cup \mathcal{P}$ (this check is decidable as shown in [14]).

$$\begin{aligned}\neg\text{Consistent}(\{q'\}, \mathcal{T}, \mathcal{P}) &= \text{false}; \\ \text{Consistent}(\{\alpha, q'\}, \mathcal{T}, \mathcal{P}) &= \text{false}.\end{aligned}$$

Thus, the sentence $CQ_k\text{CensEntailed}(\alpha, \mathcal{T}, \mathcal{P})$ (for $k = 1$) contains a subsentence (in conjunction with the rest of the sentence) of the form

$$C(b) \wedge \text{true} \wedge (\neg(\exists x A(x) \wedge B(x) \wedge x \neq b) \vee \text{false} \vee \text{false})$$

which evaluates to false over $\mathcal{I}_{\mathcal{A}}$ (because $\mathcal{I}_{\mathcal{A}}$ satisfies $\exists x(A(x) \wedge B(x) \wedge x \neq b)$). Hence, we conclude that $\mathcal{I}_{\mathcal{A}}$ does not satisfy $CQ_k\text{CensEntailed}(\alpha, \mathcal{T}, \mathcal{P})$, and therefore $C(b)$ is not entailed under \mathbf{CQ}_k censors for $k = 1$. \square

Since evaluating an FO sentence over an ABox is in AC^0 in data complexity, Proposition 7 immediately implies the main result of this subsection.

Theorem 13. *For every integer $k \geq 1$, the following holds: $CQ_k\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$ is FO-rewritable, and therefore in AC^0 in data complexity.*

The following corollary summarizes the results given in this section. Namely, it recalls the data complexity of $CQ_k\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$, where $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\mathbf{CQ}, \mathbf{CQ}_{\exists}, \mathbf{GA}\}$, for every integer $k \geq 1$.

Corollary 3. *Let k be a integer such that $k \geq 1$. We have that the problem $CQ_k\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$ is in AC^0 in data complexity. Furthermore, the following problems are coNP-complete in data complexity:*

- $CQ_k\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathcal{Q})$, for every $\mathcal{Q} \in \{\mathbf{CQ}_{\exists}, \mathbf{CQ}\}$;
- $CQ_k\text{-Cens-Entailment}(\mathcal{EL}_{\perp}, \mathcal{Q})$, for every $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}, \mathbf{CQ}\}$.

7. CQE under full censor language

In this section, we study the data complexity of the problems $CQ\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$, where $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_{\perp}\}$ and $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}, \mathbf{CQ}\}$. We start, in Section 7.1, with the case of $DL\text{-Lite}_{\mathcal{R}}$ CQE instances, and prove that $CQ\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$ is tractable and actually in AC^0 in data complexity (Theorem 14). As done for showing membership in AC^0 of $CQ_k\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{GA})$ in Section 6.3, we first provide a crucial property, given in Proposition 8, that establishes some necessary and sufficient conditions for a BCQ to be entailed by a $DL\text{-Lite}_{\mathcal{R}}$ CQE instance under \mathbf{CQ} censors, and then, based on such a property, we define a query rewriting technique that is able to reduce $CQ\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$ to the evaluation of a first-order sentence on the ABox of the CQE instance based. This membership in AC^0 obviously holds even for $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_{\exists}\}$. Then, in Section 7.2 we approach the case of \mathcal{EL}_{\perp} CQE instances, and show that $CQ\text{-Cens-Entailment}(\mathcal{EL}_{\perp}, \mathcal{Q})$ is PTIME-complete when $\mathcal{Q} = \mathbf{CQ}_{\exists}$ (Theorem 15), and in coNP when $\mathcal{Q} = \mathbf{CQ}$ (Theorem 16). Of course, this latter membership holds even when $\mathcal{Q} = \mathbf{GA}$.

The following general property of entailment under \mathbf{CQ} censors will be useful in both the next subsections. The proof follows immediately from the definition of $CQ\text{-Cens-Entailment}$.

Lemma 11. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be either a $DL\text{-Lite}_{\mathcal{R}}$ CQE instance or an \mathcal{EL}_{\perp} CQE instance, and let q be a BCQ. Then, $\mathcal{E} \vDash_{\mathbf{CQ}}^{cqe} q$ if and only if at least one of the following conditions holds: (i) $\mathcal{T} \cup \mathcal{A} \not\models q$; (ii) $\{q\} \cup \mathcal{T} \cup \mathcal{P}$ is inconsistent; (iii) there exists a BCQ $q' \in \mathbf{CQ}^{Ent}(\mathcal{T} \cup \mathcal{A})$ such that $\{q'\} \cup \mathcal{T} \cup \mathcal{P}$ is consistent and $\{q, q'\} \cup \mathcal{T} \cup \mathcal{P}$ is inconsistent.*

Intuitively, the above proposition says that a query q is not entailed under \mathbf{CQ} censors if and only if (i) q is not entailed by the CQE instance without considering the policy, or (ii) q discloses some information considered confidential according to the policy, or (iii) there is a q' as described above which implies the existence of an optimal \mathbf{CQ} censor C such that $q' \in \text{Th}(C)$ (because $q' \in \mathbf{CQ}^{Ent}(\mathcal{T} \cup \mathcal{A})$) and $\mathcal{T} \cup \text{Th}(C) \not\models q$ (because $\{q, q'\}$ is inconsistent with $\mathcal{T} \cup \mathcal{P}$).

7.1. $DL\text{-Lite}_{\mathcal{R}}$ CQE instances

We now focus on the case of $DL\text{-Lite}_{\mathcal{R}}$ CQE instances, analyzing the data complexity of $CQ\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$.

We first prove the following property, which is analogous to what Proposition 6 establishes in the case $\mathbf{CQ}_k\text{-Cens-Entailment}$. Such a property is crucial for the FO-rewritability of $CQ\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathbf{CQ})$, since each of its conditions can be tested efficiently through an FO encoding.

Proposition 8. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a $DL\text{-Lite}_{\mathcal{R}}$ CQE instance, let q be a BCQ, and let $h = \text{maxlen}(\mathcal{P}) - 1$. Then, $\mathcal{E} \vDash_{\mathbf{CQ}}^{cqe} q$ if and only if the following conditions hold:*

- (a) $\mathcal{T} \cup \mathcal{A} \models q$;

- (b) $\{q\} \cup \mathcal{T} \cup \mathcal{P}$ is consistent;
(c) there exists no BCQ $q' \in \mathbf{CQ}_h^{Ent}(\mathcal{T} \cup \mathcal{A})$ such that:
(c1) $\{q'\} \cup \mathcal{T} \cup \mathcal{P}$ is consistent;
(c2) $\{q, q'\} \cup \mathcal{T} \cup \mathcal{P}$ is inconsistent.

Proof. First, by Lemma 11, $\mathcal{E} \models_{\mathbf{CQ}}^{cqe} q$ if and only if conditions (a) and (b) hold, and there exists no $q' \in \mathbf{CQ}^{Ent}(\mathcal{T} \cup \mathcal{A})$ such that conditions (c1) and (c2) hold. Now, since $\{q, q'\} \cup \mathcal{T}$ is consistent, by Lemma 8 inconsistency w.r.t. \mathcal{P} corresponds to the entailment in $\{q, q'\} \cup \mathcal{T}$ of a query q_δ for some denial $\delta \in \mathcal{P}$. By Lemma 10, we can assume that $length(q') \leq maxlen(\mathcal{P}) - 1$. But, since $\{q'\} \cup \mathcal{T} \cup \mathcal{P}$ is consistent, at least one atom of q_δ must be satisfied by q , therefore we can assume that $length(q') \leq maxlen(\mathcal{P}) - 1$. Consequently, $q' \in \mathbf{CQ}_h^{Ent}(\mathcal{T} \cup \mathcal{A})$, which proves the thesis. \square

Notice that, with respect to Proposition 6, Proposition 8 is simpler: in particular, condition (c1) of Proposition 6 is not needed anymore, since the BCQ q' of condition (c) certainly belongs to the censor language.

Based on Proposition 8, we are now able to define a first-order sentence that encodes the entailment of BCQs under **CQ** censors in *DL-Lite_R* CQE instances.

Definition 9. Let \mathcal{T} be a *DL-Lite_R* TBox, \mathcal{P} be a policy, q a BCQ, and let $h = maxlen(\mathcal{P}) - 1$, we define the sentence $CQCensEntailed(q, \mathcal{T}, \mathcal{P})$ as follows:

$$\text{PerfectRef}(q, \mathcal{T}) \wedge \text{Consistent}(\{q\}, \mathcal{T}, \mathcal{P}) \wedge \bigwedge_{q' \in \mathbf{CQ}_h(\{q\} \cup \mathcal{P} \cup \mathcal{T})} (\neg \text{PerfectRef}(q', \mathcal{T}) \vee \neg \text{Consistent}(\{q'\}, \mathcal{T}, \mathcal{P})) \vee \text{Consistent}(\{q, q'\}, \mathcal{T}, \mathcal{P})$$

where $\text{PerfectRef}(\cdot, \cdot)$ and $\text{Consistent}(\cdot, \cdot, \cdot)$ are as in Definition 8.

The following proposition states that $CQCensEntailed(q, \mathcal{T}, \mathcal{P})$ can be used to decide $CQ\text{-Cens-Entailment}(DL\text{-Lite}_R, \mathbf{CQ})$.

Proposition 9. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be a *DL-Lite_R* CQE instance and let q be a BCQ. Then, $\mathcal{E} \models_{\mathbf{CQ}}^{cqe} q$ if and only if $I_{\mathcal{A}} \models CQCensEntailed(q, \mathcal{T}, \mathcal{P})$.

Proof. The proof is an immediate consequence of Proposition 8, of the fact that, for a consistent *DL-Lite_R* ontology $\mathcal{T} \cup \mathcal{A}$, we have that $I_{\mathcal{A}} \models \text{PerfectRef}(q, \mathcal{T})$ if and only if $\mathcal{T} \cup \mathcal{A} \models q$ (cf. Proposition 1), and of the fact that, in condition (c) of Proposition 8, it is sufficient to consider BCQs q' of maximum length $maxlen(\mathcal{P}) - 1$ mentioning only constants appearing in q or in \mathcal{P} . \square

The above property immediately implies the FO-rewritability of $CQ\text{-Cens-Entailment}(DL\text{-Lite}_R, \mathbf{CQ})$.

Theorem 14. $CQ\text{-Cens-Entailment}(DL\text{-Lite}_R, \mathbf{CQ})$ is FO-rewritable, and therefore in AC^0 in data complexity.

Example 9. Consider the CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, where \mathcal{T} is the *DL-Lite_R* TBox of Example 2, and \mathcal{A} and \mathcal{P} are as in Example 3. For clarity and convenience, we reproduce them below:

$$\begin{aligned} \mathcal{T} &= \{ \exists \text{buy} \sqsubseteq \text{Customer}, \\ &\quad \exists \text{buy}^- \sqsubseteq \text{Medicine}, \\ &\quad \text{Medicine} \sqsubseteq \exists \text{treats}, \\ &\quad \exists \text{treats}^- \sqsubseteq \text{Condition} \} \\ \mathcal{A} &= \{ \text{buy}(c_1, m_A), \text{buy}(c_1, m_B), \text{buy}(c_2, m_A) \} \\ \mathcal{P} &= \{ \forall x(\text{buy}(x, m_A) \wedge \text{buy}(x, m_B) \rightarrow \perp) \} \end{aligned}$$

Additionally, consider the following CQ asking whether the customer c_1 bought the medicine m_B :

$$q = \text{Customer}(c_1) \wedge \text{buy}(c_1, m_B) \wedge \text{Medicine}(m_B).$$

We aim to verify whether $\mathcal{E} \models_{\mathbf{CQ}}^{cqe} q$. By exploiting Proposition 9, we can do this by verifying whether $I_{\mathcal{A}} \models CQCensEntailed(q, \mathcal{T}, \mathcal{P})$. Let's analyze each component of $CQCensEntailed(q, \mathcal{T}, \mathcal{P})$ separately. $\text{PerfectRef}(q, \mathcal{T})$ contains several queries among which the query $\text{buy}(c_1, m_B)$ is the only one satisfied by $I_{\mathcal{A}}$. Moreover, since $\{\text{buy}(c_1, m_B)\} \cup \mathcal{T} \cup \mathcal{P}$ is consistent, then $\text{Consistent}(\{q\}, \mathcal{T}, \mathcal{P})$ is true. Continuing with the construction of the formula, we need to compute the set $\mathbf{CQ}_{maxlen(\mathcal{P})-1}(\{q\} \cup \mathcal{P} \cup \mathcal{T})$, that in our case is constituted by BCQs of maximum length 1 built using the predicates in $\mathcal{T} \cup \mathcal{P}$, and the constants in q or in \mathcal{P} . Queries in this set include the query $q' = \text{buy}(c_1, m_A)$. One can verify that the presence of such a query leads $CQCensEntailed(q, \mathcal{T}, \mathcal{P})$ to evaluate to false in $I_{\mathcal{A}}$, specifically because we have that $\text{Consistent}(\{q, q'\}, \mathcal{T}, \mathcal{P})$ is false. Hence, we can conclude that $\mathcal{E} \not\models_{\mathbf{CQ}}^{cqe} q$. In fact, as in Example 3, we have only two optimal censors for \mathcal{E} in **CQ**, and we can verify that the theory of one of them does not contain $\text{buy}(c_1, m_B)$. \square

7.2. \mathcal{EL}_\perp CQE instances

We now consider the case of \mathcal{EL}_\perp CQE instances. For this case, we are able to prove two complexity results:

- (1) for the class of \mathbf{CQ}_\exists queries, CQ-Cens-Entailment is PTIME-complete in data complexity;
- (2) for the class of all BCQs, CQ-Cens-Entailment is in coNP in data complexity.

We first turn our attention to (1). To prove such a result, we introduce a property (Lemma 12) that establishes that the BCQ q' of Lemma 11 can be searched among the subqueries of the BCQs occurring in the denials of \mathcal{P} . Such a property immediately implies the membership in PTIME (Theorem 15).

Hereinafter, we say that a BCQ q can be partitioned into two BCQs q' and q'' if q' and q'' have no variables in common and $q = q' \wedge q''$.

Lemma 12. *Let \mathcal{T} be an \mathcal{EL}_\perp TBox, let \mathcal{P} be a policy, let q be a \mathbf{BCQ}_\exists , and let q' be a BCQ such that $\mathcal{T} \cup \{q\}$ is consistent and $\mathcal{T} \cup \{q'\}$ is consistent. Then, $\mathcal{T} \cup \mathcal{P} \cup \{q', q\}$ is inconsistent iff there exists a denial $\delta \in \mathcal{P}$ such that q_δ can be partitioned into two BCQs q'_δ, q''_δ such that $\mathcal{T} \cup \{q'\} \models q'_\delta$ and $\mathcal{T} \cup \{q\} \models q''_\delta$.*

Proof. To prove this lemma we make use of the well-known notion of chase [30], and in particular we use the function $\text{chase}(\mathcal{A}, \mathcal{T})$ that computes the chase of an ABox \mathcal{A} with respect to an \mathcal{EL}_\perp TBox \mathcal{T} . First, by Lemma 4, $\mathcal{T} \cup \mathcal{P} \cup \{q', q\}$ is inconsistent if and only if $\mathcal{T} \cup \{q', q\} \models q_\delta$ for some $\delta \in \mathcal{P}$. Moreover, by Lemma 7, $\mathcal{T} \cup \{q', q\} \models q_\delta$ iff $\mathcal{T} \cup \text{Freeze}(\{q', q\}) \models q_\delta$. Now, $\text{Freeze}(\{q', q\}) = \text{Freeze}(q') \cup \text{Freeze}(q)$, and since $q \in \mathbf{CQ}_\exists$, it follows that $\text{Freeze}(q)$ does not share any constant symbol with $\text{Freeze}(q')$. Consequently, $\text{chase}(\text{Freeze}(\{q', q\}), \mathcal{T}) = \text{chase}(\text{Freeze}(q'), \mathcal{T}) \cup \text{chase}(\text{Freeze}(q), \mathcal{T})$, which implies that $\mathcal{T} \cup \{q', q\} \models q_\delta$ iff $\text{chase}(\text{Freeze}(q'), \mathcal{T}) \cup \text{chase}(\text{Freeze}(q), \mathcal{T}) \models q_\delta$, and since $\text{chase}(\text{Freeze}(q'), \mathcal{T})$ and $\text{chase}(\text{Freeze}(q), \mathcal{T})$ do not share any constant, the thesis follows. \square

We are now ready to prove that $\text{CQ-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ}_\exists)$ is tractable.

Theorem 15. *$\text{CQ-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ}_\exists)$ is PTIME-complete in data complexity.*

Proof. First, we prove PTIME-hardness by reducing the problem of instance checking in \mathcal{EL} under the standard semantics, which is a PTIME-hard problem [21] to $\text{CQ-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ}_\exists)$. Given any PTIME-hard instance checking problem (under the standard semantics) $\mathcal{T} \cup \mathcal{A} \models C(a)$, where \mathcal{T} is an \mathcal{EL} TBox, let us call \mathcal{T}' the TBox obtained from \mathcal{T} adding the inclusion $C \sqcap C' \sqsubseteq \exists R'.T$, where C' (resp. R') is an atomic concept (resp. an atomic role) not occurring in $\mathcal{T} \cup \mathcal{A}$, and let us call \mathcal{A}' the ABox $\mathcal{A} \cup \{C'(a)\}$. It is immediate to see that $\mathcal{T} \cup \mathcal{A} \models C(a)$ iff $\mathcal{T}' \cup \mathcal{A}' \models \exists x, y(R'(x, y))$. Moreover, since in the above reduction the ontology $\mathcal{T} \cup \mathcal{A}$ is consistent, it immediately follows that $\mathcal{T}' \cup \mathcal{A}' \models \exists x, y(R'(x, y))$ iff the CQE instance $\langle \mathcal{T}', \mathcal{A}', \emptyset \rangle \models_{\mathbf{CQ}}^{cqe} \exists x, y(R'(x, y))$, which proves the PTIME-hardness of $\text{CQ-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ}_\exists)$.

As for PTIME-membership, let q be a \mathbf{CQ}_\exists , Lemma 11 and Lemma 12 immediately imply that $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle \not\models_{\mathbf{CQ}}^{cqe} q$ iff: either (i) $\mathcal{T} \cup \mathcal{A} \not\models q$; or (ii) $\mathcal{T} \cup \mathcal{P} \cup \{q\}$ is inconsistent; or (iii) there exists a denial $\delta \in \mathcal{P}$ such that q_δ can be partitioned into two subqueries q' and q'' such that (a) $\mathcal{T} \cup \mathcal{A} \models q'$, (b) $\mathcal{T} \cup \mathcal{P} \cup \{q'\}$ is consistent, (c) $\mathcal{T} \cup \mathcal{P} \cup \{q', q\}$ is inconsistent, and (d) $\mathcal{T} \cup \{q\} \models q''$. It is immediate to verify that the above conditions (i), (ii) and (iii) can be verified in PTIME, since in \mathcal{EL}_\perp each of the above entailment checks under the standard semantics can be decided in PTIME in data complexity, and the number of possible decompositions of every denial in two subqueries is finite and independent of the ABox. \square

We now address (2), i.e., we consider the case of entailment of BCQs, and show that $\text{CQ-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ})$ is in coNP in data complexity.

Given a CQE instance $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, a BCQ q and a BCQ q' , in the following it will be convenient to use a specific definition that summarizes the following three conditions: (i) $\mathcal{T} \cup \mathcal{A} \models q'$; (ii) $\mathcal{T} \cup \mathcal{P} \cup \{q'\}$ is consistent; (iii) $\mathcal{T} \cup \mathcal{P} \cup \{q', q\}$ is inconsistent. If this is the case, we say that q' is *ensor-conflicting with q in \mathcal{E}* . Now, Lemma 11 can be immediately rephrased in terms of censor-conflicting queries as follows:

Lemma 13. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be an \mathcal{EL}_\perp CQE instance and let q be a BCQ. We have that $\mathcal{E} \models_{\mathbf{CQ}}^{cqe} q$ if and only if $\mathcal{T} \cup \mathcal{A} \models q$ and there exists no BCQ q' that is censor-conflicting with q in \mathcal{E} .*

Below we outline the structure of our proof for (2).

- We first recall the notion of partial rewriting and the existence of a partial rewriting for every BCQ over an \mathcal{EL}_\perp ontology (Lemma 14).
- Then, we show (Lemma 16) that, if there exists a BCQ q' that is censor-conflicting with a BCQ q in \mathcal{E} , then there exists a BCQ q'' that is censor-conflicting with q in \mathcal{E} and which is a subquery of a partial rewriting (with respect to \mathcal{T}) of the BCQ q_δ of some denial δ in \mathcal{P} .

- Finally, using a property (Lemma 15) of the non-tree-shaped part (called *graph-core*) of the partial rewriting identified by Lemma 16, we are able to prove that (Lemma 17), if there exists a BCQ q' that is censor-conflicting with q in \mathcal{E} , then there exists a BCQ q'' that is censor-conflicting with q in \mathcal{E} and such that the size of q'' is polynomially bounded (in data complexity). This immediately implies that CQ-Cens-Entailment is in coNP in data complexity (Theorem 16).

We recall the notion of *normalized \mathcal{EL}_\perp TBox* [31]. An \mathcal{EL}_\perp TBox \mathcal{T} is normalized if every concept inclusion in \mathcal{T} has one of the following forms:

$$\begin{aligned} A &\sqsubseteq C \\ A &\sqsubseteq \exists R.B \\ A \sqcap B &\sqsubseteq C \\ \exists R.A &\sqsubseteq C \end{aligned}$$

where A, B are either atomic concepts or \top , C is either an atomic concept or \perp , and R is an atomic role.

It is easy to verify that, given an \mathcal{EL}_\perp TBox \mathcal{T} , it is possible to compute in linear time (see [31]) a normalized \mathcal{EL}_\perp TBox \mathcal{T}' that uses a linear number of new, auxiliary atomic concepts and is such that, for every ABox \mathcal{A} and policy \mathcal{P} , $\langle \mathcal{T}', \mathcal{A}, \mathcal{P} \rangle \models_{\text{CQ}}^{cqe} q$ if and only if $\langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle \models_{\text{CQ}}^{cqe} q$ for every BCQ q on the signature of $\mathcal{T} \cup \mathcal{A} \cup \mathcal{P}$.

Consequently, from now on, without loss of generality we assume that the \mathcal{EL}_\perp CQE instances that we consider are such that their TBox \mathcal{T} is normalized.

A *homomorphism* from a BCQ q to a BCQ q' is a function h that maps the terms of q into the terms of q' in such a way that $h(q)$ is a subquery of q' , where $h(q)$ denotes the BCQ obtained from q replacing every term t with $h(t)$. It is immediate to verify that, given a set of BCQs Φ and a BCQ q , $\Phi \models q$ iff there exists a homomorphism from q to $\text{Conj}(\Phi)$, where $\text{Conj}(\Phi)$ is the BCQ corresponding to the conjunction of all the BCQs in Φ (see Section 6.1).

Given an \mathcal{EL}_\perp TBox \mathcal{T} and a BCQ q , a *partial rewriting of q w.r.t. \mathcal{T}* is a BCQ q' such that, for every set of BCQs Φ , if $\Phi \models q'$ then $\mathcal{T} \cup \Phi \models q$. The following property follows immediately from well-known properties of query answering and query rewriting in \mathcal{EL}_\perp [32,33].

Lemma 14. *Let \mathcal{T} be an \mathcal{EL}_\perp TBox, let Φ be a set of BCQs such that $\mathcal{T} \cup \Phi$ is consistent, and let q be a BCQ. If $\mathcal{T} \cup \Phi \models q$, then there exists a partial rewriting q' of q w.r.t. \mathcal{T} such that $\Phi \models q'$.*

Now, we associate every BCQ q with a directed graph $G(q)$ defined as follows. The nodes of $G(q)$ are the terms occurring in q . Every node t of $G(q)$ is labeled with a conjunction of atomic concepts (corresponding to the concept atoms for t in q). Moreover, there is an edge with label R from t_1 to t_2 in $G(q)$ if the atom $R(t_1, t_2)$ occurs in q . We say that q is *tree-shaped with root-variable v* if $G(q)$ is a tree whose root is the variable v . Note that multi-edges between two nodes prevent a query from being tree-shaped. For instance, the BCQ $q = \exists x, y, z, w (C(x) \wedge R(x, y) \wedge S(x, z) \wedge D(z) \wedge E(z) \wedge R(y, w))$ is tree-shaped with root-variable x .

An *existential subtree* of a BCQ q is a subquery q' of q that is tree-shaped, does not contain constants, and shares (at most) its root-variable with the atoms of q that do not occur in q' .

The *graph-core* of a BCQ q is the BCQ obtained from q by eliminating all the existential subtrees of q .

Lemma 15. *Let \mathcal{T} be an \mathcal{EL}_\perp TBox, let \mathcal{A} be an ABox such that $\mathcal{T} \cup \mathcal{A}$ is consistent, and let q be a BCQ. If $\mathcal{T} \cup \mathcal{A} \models q$, then there exists a partial rewriting q' of q w.r.t. \mathcal{T} such that $\mathcal{I}_{\mathcal{A}} \models q'$ and the number of distinct terms occurring in the graph-core of q' is not greater than the number of distinct terms occurring in q .*

Proof. Since by hypothesis $\mathcal{T} \cup \mathcal{A}$ is consistent, it follows that $\mathcal{T} \cup \mathcal{A} \models q$ iff $\mathcal{T}^- \cup \mathcal{A} \models q$, where \mathcal{T}^- is the \mathcal{EL} TBox obtained from \mathcal{T} by eliminating the concept inclusions mentioning \perp . Thus, the thesis follows immediately from the query rewriting technique for \mathcal{EL} shown in [27]. \square

We are now ready to prove a crucial property.

Lemma 16. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be an \mathcal{EL}_\perp CQE instance such that $\mathcal{A} = \mathcal{A}_{\mathcal{T}}$, i.e., the ABox \mathcal{A} coincides with its ground closure w.r.t. \mathcal{T} . Let q and q' be two BCQs. If q' is censor-conflicting with q in \mathcal{E} , then there exist: (i) a denial δ in \mathcal{P} , (ii) a BCQ q'_δ that is a partial rewriting of q'_δ w.r.t. \mathcal{T} , and (iii) a homomorphism h from q'_δ to $\text{Conj}(\{q', q\})$, such that $h'(q''_\delta)$ is censor-conflicting with q in \mathcal{E} and $\mathcal{I}_{\mathcal{A}} \models h'(q''_\delta)$, where h' is the restriction of h to the mapping of the variables of q'_δ to constants, and q''_δ is constituted by the set of atoms of q'_δ that are mapped by h onto atoms of q' .*

Proof. Since $\{q', q\}$ is inconsistent with $\mathcal{T} \cup \mathcal{P}$, it follows that there exists a denial $\delta \in \mathcal{P}$ such that $\mathcal{T} \cup \{q', q\} \models \delta$, hence by Lemma 14 there exists a BCQ q'_δ that is a partial rewriting of q'_δ w.r.t. \mathcal{T} such that $\{q', q\} \models q'_\delta$, which in turn implies that there exists a homomorphism h from q'_δ to $\text{Conj}(\{q', q\})$. Now let h' be the restriction of h to the mapping of variables of q'_δ into constants, and let q''_δ be the BCQ constituted by the set of atoms of q'_δ that are mapped by h into atoms of q' : since q' and q do not share variable symbols, all the variables that appear both in q''_δ and in atoms of q'_δ outside q''_δ are mapped to constants by h . Consequently, $h'(q''_\delta)$ does not share variables with the other atoms of q'_δ . It is now immediate to verify that $\{h'(q''_\delta), q\} \models \delta$, hence (a) $\mathcal{T} \cup \mathcal{P} \cup \{h'(q''_\delta), q\}$

is inconsistent; moreover, $\{q'\} \models h(q''_\delta)$, and since q' is censor-conflicting with q in \mathcal{E} it follows that (b) $\mathcal{T} \cup \mathcal{P} \cup \{h'(q''_\delta)\}$ is consistent, and (c) $\mathcal{T} \cup \mathcal{A} \models h'(q''_\delta)$. Consequently, $h'(q''_\delta)$ is censor-conflicting with q in \mathcal{E} .

Now, since $\mathcal{T} \cup \mathcal{A} \models h'(q''_\delta)$, by Lemma 14 there exists a partial rewriting q'' of $h'(q''_\delta)$ w.r.t. \mathcal{T} such that $\mathcal{I}_{\mathcal{A}} \models q''$. Since by hypothesis $\mathcal{A} = \mathcal{A}_{\mathcal{T}}$, it follows ([27]) that such a q'' can be obtained from $h'(q''_\delta)$ by roll-up steps that use inclusions of the TBox \mathcal{T} : since we assume that \mathcal{T} is a normalized TBox, such roll-up steps rewrite pairs of role and concept atoms $R(t, x)$, $C(x)$ (or single role atoms $R(t, x)$) such that x is a variable not occurring elsewhere in the query. Given the fact that q''_δ is a subquery of a rewriting q'_δ of q_δ , it follows that there exists a partial rewriting q'''_δ of q_δ w.r.t. \mathcal{T} in which the variables eliminated by the roll-up steps that transform $h'(q''_\delta)$ in q'' are not occurring at all (i.e., none of the backward chaining steps that introduce such variables in q'_δ is executed in the rewriting process that generates q'''_δ) and $q'_\delta \models q'''_\delta$. Therefore, assuming now that q''_δ is the BCQ constituted by the set of atoms of q'_δ that are mapped by h onto atoms of q' , the above properties (a), (b) and (c) still hold, and in addition, $\mathcal{I}_{\mathcal{A}} \models h'(q''_\delta)$, thus proving the thesis. \square

The next lemma establishes an upper bound on the size of a censor-conflicting BCQ.

Lemma 17. *Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$ be an \mathcal{EL}_\perp CQE instance such that $\mathcal{A} = \mathcal{A}_{\mathcal{T}}$, and let q be a BCQ. If there exists a BCQ q' that is censor-conflicting with q in \mathcal{E} , then there exists a BCQ q'' that is censor-conflicting with q in \mathcal{E} and whose length is no greater than $k(k + nk^2(n + 2k)^{2k})^2$, where k is the size of $\mathcal{T} \cup \mathcal{P} \cup \{q\}$ and n is the size of \mathcal{A} .*

Proof. Given a BCQ q and two variables x, y occurring in q , we say that x and y are $\mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A})$ -equivalent in q if, for every BCQ $q' \in \mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A})$, $\sigma(q) \models \sigma'(q')$ iff $\sigma(q) \models \sigma''(q')$, where $\sigma = \{x \rightarrow a_x, y \rightarrow a_y\}$, $\sigma' = \{z \rightarrow a_x\}$, $\sigma'' = \{z \rightarrow a_y\}$, z is a variable occurring in q' and a_x and a_y are constants not occurring in q .

Now let $h'(q''_\delta)$ be the BCQ of Lemma 16. Let h_0 be a homomorphism that maps (the variables of) $h'(q''_\delta)$ into (the constants of) \mathcal{A} . Let q'' be the BCQ obtained from $h'(q''_\delta)$ unifying two of its variables x and y such that $h_0(x) = h_0(y)$, x and y are the root-variables of two existential subtrees of q''_δ , and x and y are $\mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A})$ -equivalent. Also, let S_x be the subtree of x in $h'(q''_\delta)$ and let S_y be the subtree of y in $h'(q''_\delta)$. Now suppose that $\mathcal{T} \cup \mathcal{P} \cup \{q''\}$ is inconsistent. Then, there exists a denial δ_0 such that q_{δ_0} contains a variable z whose subtree S_z is isomorphic to $S_x \cup S_y$. However, since $\delta \in \mathcal{P}$, it immediately follows that the denial δ_1 obtained from δ_0 replacing the subtree S_z with S_{xz} , where S_{xz} is the subtree obtained from S_x replacing x with z , is a denial implied by $\mathcal{T} \cup \mathcal{P}$. But now, it is immediate to verify that $h'(q''_\delta)$ implies q_{δ_1} , consequently $\mathcal{T} \cup \mathcal{P} \cup \{h'(q''_\delta)\}$ is inconsistent, thus contradicting the hypothesis. Therefore, $\mathcal{T} \cup \mathcal{P} \cup \{q''\}$ is consistent.

Now let q' be the BCQ obtained from $h'(q''_\delta)$ by unifying *all* the variables x and y such that $h_0(x) = h_0(y)$ and x and y are $\mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A})$ -equivalent. By iterating the above argument used for q'' , it follows that $\mathcal{T} \cup \mathcal{P} \cup \{q'\}$ is consistent. Moreover, of course $\mathcal{T} \cup \mathcal{P} \cup \{q', q\}$ is inconsistent (since $\mathcal{T} \cup \mathcal{P} \cup \{h'(q''_\delta), q\}$ is inconsistent and $q' \models h'(q''_\delta)$). Finally, it is immediate to see that $\mathcal{I}_{\mathcal{A}} \models q'$ (because of the homomorphism h_0). Consequently, q' is censor-conflicting with q in \mathcal{E} .

Moreover, observe that there are at most $nk^2(n + 2k)^{2k}$ variables in the subtrees of $h'(q''_\delta)$ that either are not $\mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A})$ -equivalent or are mapped by h_0 into different constants (since $n + 2k$ is the number of terms that can occur in the BCQs of $\mathbf{CQ}_k(\mathcal{T} \cup \mathcal{A})$, k is the maximum number of atoms and $2k$ is the maximum number of terms in a BCQ), and, by Lemma 15, we can assume that q'_δ has a graph-core such that the number of terms occurring in it is not greater than the number of terms occurring in q_δ , hence such a number of terms is bounded by k . Consequently, we have that in $h'(q''_\delta)$ there are at most k (initial) terms occurring outside the subtrees of $h'(q''_\delta)$. Therefore, the number of distinct terms occurring in $h'(q''_\delta)$ is bounded by $k + nk^2(n + 2k)^{2k}$, which implies that the number of distinct atoms that occur in $h'(q''_\delta)$ is bounded by $k(k + nk^2(n + 2k)^{2k})^2$. \square

Using Lemma 17, we are able to provide an upper bound for the data complexity of $CQ\text{-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ})$.

Theorem 16. *$CQ\text{-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ})$ is in coNP in data complexity.*

Proof. Let $\mathcal{E} = \langle \mathcal{T}, \mathcal{A}, \mathcal{P} \rangle$, with \mathcal{T} an \mathcal{EL}_\perp TBox, and let q be a BCQ. By Lemma 13 and Lemma 17, we can decide $\mathcal{E} \not\models_{\mathbf{CQ}}^{cqe} q$ by first computing $\mathcal{A}_{\mathcal{T}}$, then guessing both a BCQ q' whose size is no greater than $k(n + k + nk^2(n + 2k)^{2k})^2$, and a homomorphism h from the variables of q' to the constants of \mathcal{A} , and then checking that: (i) every atom of $h(q')$ belongs to $\mathcal{A}_{\mathcal{T}}$ (i.e. $\mathcal{I}_{\mathcal{A}_{\mathcal{T}}} \models q'$); (ii) $\mathcal{T} \cup \mathcal{P} \cup \{q'\}$ is consistent, i.e. by Lemma 4, $\mathcal{T} \cup \{q'\} \not\models q_\delta$ for every $\delta \in \mathcal{P}$; (iii) $\mathcal{T} \cup \mathcal{P} \cup \{q, q'\}$ is inconsistent, i.e. by Lemma 4, $\mathcal{T} \cup \{q', q\} \models q_\delta$ for some $\delta \in \mathcal{P}$. Since both the computation of $\mathcal{A}_{\mathcal{T}}$ and the three above conditions can be checked in PTIME in data complexity, the thesis follows. \square

We finally recall the results established in this section regarding the data complexity of $CQ\text{-Cens-Entailment}(\mathcal{L}_{\mathcal{T}}, \mathcal{Q})$, where $\mathcal{L}_{\mathcal{T}} \in \{DL\text{-Lite}_{\mathcal{R}}, \mathcal{EL}_\perp\}$ and $\mathcal{Q} \in \{\mathbf{CQ}, \mathbf{CQ}_\exists, \mathbf{GA}\}$.

Corollary 4. *$CQ\text{-Cens-Entailment}(DL\text{-Lite}_{\mathcal{R}}, \mathcal{Q})$ is in AC^0 in data complexity, for every $\mathcal{Q} \in \{\mathbf{GA}, \mathbf{CQ}_\exists, \mathbf{CQ}\}$. Furthermore, we have that:*

- $CQ\text{-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ}_\exists)$ is PTIME-complete in data complexity;
- $CQ\text{-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{GA})$ and $CQ\text{-Cens-Entailment}(\mathcal{EL}_\perp, \mathbf{CQ})$ are both PTIME-hard and in coNP in data complexity.

8. Related work

In the last thirty years, CQE has been studied quite extensively in the context of databases and knowledge bases. It was originally introduced in [5] for propositional databases. In that paper, the authors provide a general framework for confidentiality-preserving query answering and discuss some criteria that can be used by a censor to alter user queries. Analogously to our framework, the aim is to ensure that an attacker cannot exactly deduce secret information from the responses of the system, whereas other works on confidentiality preservation (e.g. [34,35]) also study how to prevent the user from establishing which secret is more probable. In [5], data are interpreted under the closed world assumption and censors may refuse to provide an answer to a query but never lie to users. The relationship between the lying approach, i.e. when the system may return non-correct answers, and the refusal one, i.e. when the system can refuse to answer some queries, has been then investigated for propositional closed databases in [36–38,6]. The case of propositional open databases is instead studied in [16].

Clearly, all the above papers focus on a setting different from the one studied in this article. In particular, we remark that ontologies considered in our framework are first-order open theories and that our censors never mark a query as refused, but enforce confidentiality by returning a strict subset of the query answers entailed by the system. Thus, we adopt a form of lying, since may say that a query is not entailed, even if in fact it is.

Works on CQE over DL ontologies are closer to our research [15,8,7,39]. In [15], the authors generalize the CQE paradigm for incomplete databases presented in [16], and study CQE for ontologies in OWL 2 RL, one of the tractable profiles of OWL 2 [20], and policies represented by sets of ground atoms. The paper [15] proposes the use of optimal (view-based) censors that allow users to query the ontology without inferring the ground atoms in the policy. It also identifies a fragment of OWL 2 RL for which constructing (a secure view of the ontology through) one such censor is polynomial. The same authors then continue their investigation in [8], for ontologies specified in Datalog or in one of the OWL2 profiles and for policy expressed as a CQ. As in [15], the main focus of the paper is on verifying the existence of a censor and establishing the computational complexity of producing it. Two incomparable different censor notions are considered, one based on views (as in [15]), and another based on obstruction.

The paper [7] instead considers a framework in which a knowledge base \mathcal{K} , expressed in any DL \mathcal{L} with decidable reasoning and enjoying compactness (as, e.g. \mathcal{ALC}), is coupled with a set S of so-called secrecies, i.e. axioms in \mathcal{L} that, if entailed by \mathcal{K} , should not be disclosed to the user. The paper analyzes two security models. In the first model, a view \mathcal{K}_u is exposed to the user only if \mathcal{K}_u is contained in the set of the logical consequences of \mathcal{K} and has an empty intersection with S . The second model is more sophisticated and imposes, in addition, that a user should not understand whether \mathcal{K}_u is obtained by filtering \mathcal{K} or another knowledge base not inferring secrecies (this property has been later called *indistinguishability* in the literature). The latter model is shown to be more robust than the former to attacks of users that may have additional object-level background knowledge and/or are aware that the underlying ontology has complete knowledge about a certain set of axioms. Differently from [8], the paper [7] does not clearly distinguish between the intensional and extensional levels of the ontology. Moreover, it does not allow to specify CQs in the policy, since the considered secrecies are axioms in the ontology language \mathcal{L} , which, being a DL, cannot capture CQs. A restricted policy language, consisting only of subsumptions between concepts, is also investigated in [39], where the authors define and analyze properties of censors for Boolean \mathcal{ALC} ontologies.

In the present paper, we elaborate on the framework proposed in [15,8], since we want an expressive policy (i.e. given in terms of CQs) and confidentiality enforced by hiding only facts contained in the ABox, as needed in many practical settings. More precisely, we revise the approach of [15,8] and define CQE as the problem of computing the answers to a query that are in the intersection of the answers returned by all the optimal censors, i.e. we study a form of skeptical reasoning over all such censors. As noted in the introduction, this concept has already been discussed in [15], which offers some initial contributions in this direction, though addressing the issue is not among its primary objectives. Specifically, the paper identifies *linear RL*⁻ ontologies among those that allow for a single (view-definable) censor. For non-linear *RL*⁻ ontologies, it proposes a WIDTIO (When In Doubt Throw It Out) approach, where the censor is determined as the intersection of all optimal censors. We emphasize that this latter strategy is substantially different from our approach of skeptical reasoning on all optimal censors. A further difference of our paper with respect to [15,8], is that we also parametrize the notion of censor to a language. Then, we study computational complexity of CQE for four different specific languages: the language coinciding with the ABox of the CQE instance, **GA**, **CQ_k**, and **CQ**. This latter language is the only one implicitly considered in [8].

Aside the above mentioned differences, in the present article, we inherit the user model proposed in [8], and therefore, as in that paper, we do not require censors to enjoy the indistinguishability property. We instead elaborate on indistinguishability in [40–42] for confidentiality-preserving query answering over (possibly prioritized) DL ontologies, and in [43] in the context of ontology-based data access and integration. Information disclosure under censors enjoying indistinguishability in a data integration setting is also studied in [44,45]. For a thorough comparison between censors that fulfill or not indistinguishability in the context of DL ontologies, we direct the reader to [40] and to the more recent note [46].

Besides CQE, confidentiality issues in DLs have also been studied using different approaches. The paper [4] proposes the use of authorization views to specify the information accessible by users. The use of views to protect sensitive data can be considered somewhat complementary to the CQE approach, in which the policy specifies the non-accessible information. It is worth noting that the idea of authorization views was originally introduced in [47] within the context of databases and has since been further developed [35,48]. These papers explore the notion of perfect privacy, wherein non-disclosure of confidential data is guaranteed if the adversary's belief about the query answer remains unchanged even after seeing the answer to the views. The setting investigated is probabilistic and does not account for ontologies.

Provable data privacy on views has been considered in [49], for concept retrieval and subsumption queries over \mathcal{ALC} ontologies. Secrecy preserving reasoning in the presence of several agents has been instead studied in [50], for propositional Horn logics and the DL \mathcal{AL} . Privacy-preserving query answering as a reasoning problem has been addressed in [2], whereas instance checking for \mathcal{EL} ontologies has been studied in [3], in both cases in frameworks different from CQE. Data anonymization in the context of ontologies has been instead studied in [51–53]. In particular, in [52,53] the authors propose a comprehensive solution for both (a form of) repairing and privacy-preservation, moving from the observation that such problems share the aim of having to hide some unwanted (incorrect or confidential) consequences to users. Even though the spirit of this observation is similar to the motivation at the basis of the present work, the approach of [52,53] is inherently different from ours, because it is based on the idea of modifying the data in order to guarantee consistency or privacy through the use of existential individuals in place of hard constants.

As thoroughly discussed in this paper, the problem of repairing data that violate integrity constraints specified over the data schema has been extensively studied within the context of Consistent Query Answering (CQA). Initially investigated for relational databases (see, e.g. [9,10,54]), this approach was later applied to ontologies (as in [14,11,55]). In abstract terms, CQA is the problem of computing answers to queries by reasoning over all possible repairs. In the context of ontologies, because of the open world semantics they adopt, a natural definition that has been frequently considered is that of ABox-repair (see Definition 4). Since CQA under ABox-repairs is inherently intractable even for lightweight ontologies like $DL\text{-}Lite_{\mathcal{R}}$ [56,57], people often adopted a WIDTIO approach and resorted to consider as (unique) repair the intersection of all the ABox-repairs, called *IAR-repair*. Entailment under IAR-repair is a sound approximation of entailment under ABox-repairs (see Definition 7), and may enjoy some nice computational behavior for lightweight ontology languages. For instance, it is first-order rewritable (and thus in AC^0 in data complexity) for some logics of the $DL\text{-}Lite$ family [14].

In this paper, we study the relationship between CQE and CQA, and to this aim we make use of the notions of ABox and IAR-repairs, as well as of some results about query entailment under such repairs for $DL\text{-}Lite$ and \mathcal{EL} ontologies. All these notions and results have been formally introduced in the previous sections. Other forms of repairs proposed in the context of ontologies, as the CAR, ICAR, or ICR-repair [57,6], are not considered in this paper.

We conclude this section by pointing out that some connections between CQA and declarative privacy preservation are also discussed in [58]. The framework in that paper is similar to ours, with so-called secrecy views playing essentially the role of the policy. However, the setting considered there is relational and without intensional knowledge (TBox), and secrecy views are enforced through virtual modifications of database values with SQL NULLS, so that this approach is incomparable with ours. Nonetheless, in the present article we elaborate on the intuition of [58] and investigate in depth the relationship between our CQE framework and CQA in DLs.

9. Discussion and conclusions

The complexity results presented in the paper provide an almost complete picture of the data complexity of conjunctive query entailment in our CQE framework for $DL\text{-}Lite_{\mathcal{R}}$ and \mathcal{EL}_{\perp} , as summarized in Fig. 1.

Among the data complexity results given in this paper, those for the entailment problem under **CQ** sensors are certainly the most surprising and unpredictable.

In fact, as highlighted by Fig. 1:

- in the case of $DL\text{-}Lite_{\mathcal{R}}$ CQE instances, entailment of BCQs under **CQ** sensors is tractable and in AC^0 , while the same problem under ABox sensors, **GA** sensors, and **CQ_k** sensors is intractable (coNP-complete) in data complexity;
- similarly, in the case of \mathcal{EL}_{\perp} CQE instances, entailment of BCQs without constants under **CQ** sensors is tractable and in PTIME, while the same problem under ABox sensors, **GA** sensors, and **CQ_k** sensors is intractable (coNP-complete) in data complexity.

In the case of $DL\text{-}Lite_{\mathcal{R}}$, the reason of this behavior lies in the fact that, as shown by Proposition 8 and Definition 9, the entailment of a BCQ q under **CQ** sensors can be decided essentially by checking the existence of a single BCQ of fixed size (in data complexity) having some properties that can be checked in polynomial time (actually in AC^0). This property does not hold for all the other sensor languages considered in the paper.

In the case of \mathcal{EL}_{\perp} , the reason for the tractability lies in the property expressed by Lemma 11 that, together with Lemma 12, identifies a technique that allows for deciding the entailment of a BCQ under **CQ** sensors in polynomial time (in data complexity). Notice that Lemma 11 does not hold for any sensor language different from **CQ** considered in this paper. Notice also that Lemma 12 does not hold for BCQs with constants (it does not hold even for ground atoms).

We also remark that, while some of the results for ABox and **GA** sensors have been obtained by exploiting the correspondence between CQE and CQA stated in Section 4 and some known complexity results on CQA in DLs, all the results for entailment under both **CQ** sensors and under **CQ_k** sensors are not related to any known complexity result in the setting of CQA, and require proofs and techniques rather different from those used in the field of CQA.

As for future work, we believe that it would be very important to provide an exact bound for the entailment of BCQs under **CQ** sensors in \mathcal{EL}_{\perp} CQE instances. In particular, it would be very interesting to verify whether the above described computational advantage provided by **CQ** sensors over the other sensor languages extends to such a case.

Then, the complexity analysis of CQE could be extended to more expressive policy languages (as in [41,59]), as well as to other sensor languages and other DLs. For example, it would be interesting to consider CQE instances whose TBox combines features of both $DL\text{-}Lite_{\mathcal{R}}$ and \mathcal{EL}_{\perp} .

Also, based on the complexity analysis of CQE presented in this paper, it would be very important to look for practical techniques allowing for the implementation of CQE extensions of current DL reasoners and Ontology-based Data Access systems [60,61]. In particular, we believe that the first-order rewritings presented in the paper (Definition 8 and Definition 9) constitute an important starting point towards the definition of practical algorithms for CQE in DLs. First attempts in this direction can be found in [43].

Finally, we remark that the approach to CQE pursued in this paper effectively protects sensitive information in the presence of users able to make classical first-order inferences over the ontology and the query answers. On the one hand, this approach has a natural relationship with CQA, and formalizing this correspondence has been one of the main objectives of our investigation. On the other hand, the confidentiality-preservation model we have realized does not consider more sophisticated capabilities of the users (which, for instance, could be able to make non-classical inferences exploiting forms of closed-world reasoning, or could be equipped with additional domain knowledge), as highlighted in [46]. Refining the framework so that it implements a richer confidentiality-preservation model is another important research direction which we intend to address in the future.

CRedit authorship contribution statement

Gianluca Cima: Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation, Formal analysis, Conceptualization. **Domenico Lembo:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation, Formal analysis, Conceptualization. **Riccardo Rosati:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation, Formal analysis, Conceptualization. **Domenico Fabio Savo:** Writing – review & editing, Writing – original draft, Validation, Supervision, Methodology, Investigation, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgements

This work was partially supported by: projects FAIR (PE0000013) and SERICS (PE0000014) under the MUR National Recovery and Resilience Plan funded by the EU - NextGenerationEU; GLACIATION (101070141) project funded by the EU; ANTHEM (PNC0000003) (AdvaNced Technologies for Human-centrEd Medicine) project (CUP B53C22006700001) funded by the MUR National Plan for NRRP Complementary Investments; and the MUR PRIN POLAR (2022LA8XBH) project (POLicy specificAtion and enfoRcement for privacy-enhanced data management).

References

- [1] D. Lembo, R. Rosati, D.F. Savo, Revisiting controlled query evaluation in description logics, in: Proc. of the 28th Int. Joint Conf. on Artificial Intelligence (IJCAI), 2019, pp. 1786–1792.
- [2] B. Cuenca Grau, I. Horrocks, Privacy-preserving query answering in logic-based information systems, in: Proc. of the 18th Eur. Conf. on Artificial Intelligence (ECAD), 2008, pp. 40–44.
- [3] J. Tao, G. Slutzki, V.G. Honavar, Secrecy-preserving query answering for instance checking in \mathcal{EL} , in: Proc. of the 4th Int. Conf. on Web Reasoning and Rule Systems (RR), 2010, pp. 195–203.
- [4] D. Calvanese, G. De Giacomo, M. Lenzerini, R. Rosati, View-based query answering in description logics: semantics and complexity, J. Comput. Syst. Sci. 78 (1) (2012) 26–46.
- [5] G.L. Sicherman, W. de Jonge, R.P. van de Riet, Answering queries without revealing secrets, ACM Trans. Database Syst. 8 (1) (1983) 41–59.
- [6] J. Biskup, P.A. Bonatti, Controlled query evaluation for enforcing confidentiality in complete information systems, Int. J. Inf. Secur. 3 (1) (2004) 14–27.
- [7] P.A. Bonatti, L. Sauro, A confidentiality model for ontologies, in: Proc. of the 12th Int. Semantic Web Conf. (ISWC), 2013, pp. 17–32.
- [8] B. Cuenca Grau, E. Kharlamov, E.V. Kostylev, D. Zheleznyakov, Controlled query evaluation for datalog and OWL 2 profile ontologies, in: Proc. of the 24th Int. Joint Conf. on Artificial Intelligence (IJCAI), 2015, pp. 2883–2889.
- [9] M. Arenas, L.E. Bertossi, J. Chomicki, Consistent query answers in inconsistent databases, in: Proc. of the 18th ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS), 1999, pp. 68–79.
- [10] L.E. Bertossi, Database Repairing and Consistent Query Answering, Synthesis Lectures on Data Management, Morgan & Claypool Publishers, 2011.
- [11] M. Bienvenu, C. Bourgaux, Inconsistency-tolerant querying of description logic knowledge bases, in: Reasoning Web. Semantic Technologies for Intelligent Data Access – 12th Int. Summer School Tutorial Lectures (RW), 2016, pp. 156–202.
- [12] M. Calautti, S. Greco, C. Molinaro, I. Trubitsyna, Preference-based inconsistency-tolerant query answering under existential rules, Artif. Intell. 312 (2022).
- [13] T. Lukasiewicz, E. Malizia, M.V. Martinez, C. Molinaro, A. Pieris, G.I. Simari, Inconsistency-tolerant query answering for existential rules, Artif. Intell. 307 (2022).
- [14] D. Lembo, M. Lenzerini, R. Rosati, M. Ruzzi, D.F. Savo, Inconsistency-tolerant query answering in ontology-based data access, J. Web Semant. 33 (2015) 3–29.
- [15] B. Cuenca Grau, E. Kharlamov, E.V. Kostylev, D. Zheleznyakov, Controlled query evaluation over OWL 2 RL ontologies, in: Proc. of the 12th Int. Semantic Web Conf. (ISWC), 2013, pp. 49–65.
- [16] J. Biskup, T. Weibert, Keeping secrets in incomplete databases, Int. J. Inf. Secur. 7 (3) (2008) 199–217.
- [17] R. Rosati, On the complexity of dealing with inconsistency in description logic ontologies, in: Proc. of the 22nd Int. Joint Conf. on Artificial Intelligence (IJCAI), 2011, pp. 1057–1062.

- [18] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Tractable reasoning and efficient query answering in description logics: the DL-Lite family, *J. Autom. Reason.* 39 (3) (2007) 385–429.
- [19] F. Baader, R. Küsters, R. Molitor, Computing least common subsumers in description logics with existential restrictions, in: *Proc. of the 16th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 1999, pp. 96–101.
- [20] B. Motik, A. Fokoue, I. Horrocks, Z. Wu, C. Lutz, B. Cuenca Grau, O.W.L. Web, Ontology language profiles, W3C recommendation, World Wide Web Consortium, available at <http://www.w3.org/TR/owl-profiles/>, Oct. 2009.
- [21] D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, R. Rosati, Data complexity of query answering in description logics, *Artif. Intell.* 195 (2013) 335–360.
- [22] F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P.F. Patel-Schneider (Eds.), *The Description Logic Handbook: Theory, Implementation and Applications*, 2nd edition, Cambridge University Press, 2007.
- [23] M.Y. Vardi, The complexity of relational query languages, in: *Proc. of the 14th ACM SIGACT Symp. on Theory of Computing (STOC)*, 1982, pp. 137–146.
- [24] C.H. Papadimitriou, *Computational Complexity*, Addison Wesley Publ. Co., 1994.
- [25] T. Lukasiewicz, E. Malizia, M.V. Martínez, C. Molinaro, A. Pieris, G.I. Simari, Inconsistency-tolerant query answering for existential rules, *Artif. Intell.* 307 (2022) 103685.
- [26] M. Ortiz, S. Rudolph, M. Simkus, Query answering in the horn fragments of the description logics SHOIQ and SROIQ, in: *Proc. of the 22nd Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2011, pp. 1039–1044.
- [27] R. Rosati, On conjunctive query answering in \mathcal{EL} , in: *Proc. of the 20th Int. Workshop on Description Logic (DL)*, in: CEUR, vol. 250, 2007, pp. 451–458.
- [28] P.G. Hinman, *Fundamentals of Mathematical Logic*, A K Peters/CRC Press, 2005.
- [29] G. Gottlob, S. Kikot, R. Kontchakov, V.V. Podolskii, T. Schwentick, M. Zakharyashev, The price of query rewriting in ontology-based data access, *Artif. Intell.* 213 (2014) 42–59.
- [30] R. Fagin, P.G. Kolaitis, R.J. Miller, L. Popa, Data exchange: semantics and query answering, *Theor. Comput. Sci.* 336 (1) (2005) 89–124.
- [31] F. Baader, S. Brandt, C. Lutz, Pushing the \mathcal{EL} envelope, in: *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2005, pp. 364–369.
- [32] H. Pérez-Urbina, B. Motik, I. Horrocks, Tractable query answering and rewriting under description logic constraints, *J. Appl. Log.* 8 (2) (2010) 186–209.
- [33] M. König, M. Leclère, M. Mugnier, M. Thomazo, Sound, complete and minimal UCQ-rewriting for existential rules, *Semant. Web* 6 (5) (2015) 451–475.
- [34] F. Bancilhon, N. Spyrtatos, Protection of information in relational data bases, in: *Proc. of the 3rd Int. Conf. on Very Large Data Bases (VLDB)*, 1977, pp. 494–500.
- [35] G. Miklau, D. Suciu, A formal analysis of information disclosure in data exchange, *J. Comput. Syst. Sci.* 73 (3) (2007) 507–534.
- [36] J. Biskup, For unknown secrecies refusal is better than lying, *Data Knowl. Eng.* 33 (1) (2000) 1–23.
- [37] J. Biskup, P.A. Bonatti, Lying versus refusal for known potential secrets, *Data Knowl. Eng.* 38 (2) (2001) 199–222.
- [38] J. Biskup, P.A. Bonatti, Controlled query evaluation for known policies by combining lying and refusal, *Ann. Math. Artif. Intell.* 40 (1–2) (2004) 37–62.
- [39] T. Studer, J. Werner, Censors for boolean description logic, *Trans. Data Priv.* 7 (3) (2014) 223–252.
- [40] G. Cima, D. Lembo, R. Rosati, D.F. Savo, Controlled query evaluation in description logics through instance indistinguishability, in: *Proc. of the 29th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2020, pp. 1791–1797.
- [41] G. Cima, D. Lembo, L. Marconi, R. Rosati, D.F. Savo, Controlled query evaluation over prioritized ontologies with expressive data protection policies, in: *Proc. of the 20th Int. Semantic Web Conf. (ISWC)*, in: *Lecture Notes in Computer Science*, vol. 12922, Springer, 2021, pp. 374–391.
- [42] P. Bonatti, G. Cima, D. Lembo, L. Marconi, R. Rosati, L. Sauro, D.F. Savo, Controlled query evaluation in OWL 2 QL: a “longest honeymoon” approach, in: *Proc. of the 21th Int. Semantic Web Conf. (ISWC)*, in: *Lecture Notes in Computer Science*, vol. 12922, Springer, 2021, pp. 428–444.
- [43] G. Cima, D. Lembo, L. Marconi, R. Rosati, D.F. Savo, Controlled query evaluation in ontology-based data access, in: *Proc. of the 19th Int. Semantic Web Conf. (ISWC)*, in: *Lecture Notes in Computer Science*, vol. 12506, Springer, 2020, pp. 128–146.
- [44] M. Benedikt, B. Cuenca Grau, E.V. Kostylev, Logical foundations of information disclosure in ontology-based data integration, *Artif. Intell.* 262 (2018) 52–95.
- [45] M. Benedikt, P. Bourhis, L. Jachiet, M. Thomazo, Reasoning about disclosure in data integration in the presence of source constraints, in: *Proc. of the 28th Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2019, pp. 1551–1557.
- [46] P.A. Bonatti, A false sense of security, *Artif. Intell.* 310 (2022).
- [47] Z. Zhang, A. Mendelzon, Authorization views and conditional query containment, in: *Proc. of the 10th Int. Conf. on Database Theory (ICDT)*, in: *Lecture Notes in Computer Science*, vol. 3363, Springer, 2005, pp. 259–273.
- [48] A. Machanavajjhala, J. Gehrke, On the efficiency of checking perfect privacy, in: S. Vansummeren (Ed.), *Proc. of the 25th ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS)*, ACM Press, 2006, pp. 163–172.
- [49] P. Stouppa, T. Studer, Data privacy for \mathcal{ALC} knowledge bases, in: *Proc. of the Int. Symp. on Logical Foundations of Computer Science (LFCS)*, 2009, pp. 409–421.
- [50] J. Tao, G. Slutzki, V.G. Honavar, A conceptual framework for secrecy-preserving reasoning in knowledge bases, *ACM Trans. Comput. Log.* 16 (1) (2014) 3:1–3:32.
- [51] B.C. Grau, E.V. Kostylev, Logical foundations of linked data anonymisation, *J. Artif. Intell. Res.* 64 (2019) 253–314.
- [52] F. Baader, F. Kriegel, A. Nuradiansyah, R. Peñaloza, Computing compliant anonymisations of quantified ABoxes w.r.t. \mathcal{EL} policies, in: *Proc. of the 19th Int. Semantic Web Conf. (ISWC)*, in: *Lecture Notes in Computer Science*, vol. 12506, Springer, 2020, pp. 3–20.
- [53] F. Baader, P. Koopmann, F. Kriegel, A. Nuradiansyah, Computing Optimal Repairs of Quantified ABoxes w.r.t. Static \mathcal{EL} TBoxes, *Proc. of the 28th Int. Conf. on Automated Deduction (CADE)*, vol. 12699, Springer, 2021, pp. 309–326.
- [54] A. Cali, D. Lembo, R. Rosati, On the decidability and complexity of query answering over inconsistent and incomplete databases, in: *Proc. of the 22nd ACM SIGACT SIGMOD SIGART Symp. on Principles of Database Systems (PODS)*, 2003, pp. 260–271.
- [55] T. Lukasiewicz, E. Malizia, M.V. Martínez, C. Molinaro, A. Pieris, G.I. Simari, Inconsistency-tolerant query answering for existential rules, *Artif. Intell.* 307 (2022) 103685.
- [56] D. Lembo, M. Ruzzi, Consistent query answering over description logic ontologies, in: *Proc. of the 1st Int. Conf. on Web Reasoning and Rule Systems (RR)*, 2007.
- [57] D. Lembo, M. Lenzerini, R. Rosati, M. Ruzzi, D.F. Savo, Inconsistency-tolerant semantics for description logics, in: *Proc. of the 4th Int. Conf. on Web Reasoning and Rule Systems (RR)*, 2010, pp. 103–117.
- [58] L.E. Bertossi, L. Li, Achieving data privacy through secrecy views and null-based virtual updates, *IEEE Trans. Knowl. Data Eng.* 25 (5) (2013) 987–1000.
- [59] G. Cima, D. Lembo, L. Marconi, R. Rosati, D.F. Savo, D. Sinibaldi, Controlled query evaluation over ontologies through policies with numerical restrictions, in: *Proc. of the 4th IEEE Int. Conf. on Artificial Intelligence and Knowledge Engineering, (AIKE)*, IEEE, 2021, pp. 33–36.
- [60] D. Calvanese, B. Cogrel, S. Komla-Ebri, R. Kontchakov, D. Lanti, M. Rezk, M. Rodriguez-Muro, G. Xiao, Ontop: answering SPARQL queries over relational databases, *Semant. Web* 8 (3) (2017) 471–487.
- [61] G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, R. Rosati, M. Ruzzi, D.F. Savo, MASTRO: A Reasoner for Effective Ontology-Based Data Access, *Proc. of the 1st Int. Workshop on OWL Reasoner Evaluation (ORE)*, vol. 858, CEUR, 2012.