Non-Uniform Data Complexity of Query Answering in Description Logics

Carsten Lutz[1] and Frank Wolter[2]

[1] Department of Computer Science, University of Bremen, Germany
[2] Department of Computer Science, University of Liverpool, UK
clu@uni-bremen.de,Wolter@liverpool.ac.uk

1 Introduction

In recent years, the use of ontologies to access instance data has become increasingly popular. The general idea is that an ontology provides a vocabulary or conceptual model for the application domain, which can then be used as an interface for querying instance data and to derive additional facts. In this emerging area, called ontology-based data access (OBDA), it is a central research goal to identify ontology languages for which query answering scales to large amounts of instance data. Since the size of the data is typically very large compared to the size of the ontology and the size of the query, the central measure for such scalability is provided by *data complexity*—the complexity of query answering where only the data is considered to be an input, but both the query and the ontology are fixed.

In description logic (DL), ontologies take the form of a TBox, instance data is stored in an ABox, and the most important class of queries are conjunctive queries (CQs). A fundamental observation regarding this setup is that, for expressive DLs such as $\mathcal{ALC}$ and $\mathcal{SHIQ}$, the complexity of query answering is coNP-complete [12] and thus intractable (when speaking of complexity, we *always* mean data complexity). The most popular strategy to avoid this problem is to replace $\mathcal{ALC}$ and $\mathcal{SHIQ}$ with less expressive DLs that *are Horn* in the sense that they can be embedded into the Horn fragment of first-order (FO) logic and have minimal models that can be exploited for PTIME query answering. Horn DLs in this sense include, for example, logics from the $\mathcal{EL}$ and DL-Lite families as well as Horn-$\mathcal{SHIQ}$, a large fragment of $\mathcal{SHIQ}$ for which CQ-answering is still in PTIME [12]. While CQ-answering in Horn-$\mathcal{SHIQ}$ and the $\mathcal{EL}$ family of DLs is also hard for PTIME, the problem has even lower complexity in DL-Lite. In fact, the design goal of DL-Lite was to achieve *FO-rewritability*, i.e., that any CQ q and TBox $\mathcal{T}$ can be rewritten into an FO query q' such that the answers to q w.r.t. $\mathcal{T}$ coincide with the answers that a standard database system produces for q' [6]. Achieving this goal requires CQ-answering to be in AC^0.

It thus seems that the data complexity of query answering in a DL context is well-understood. However, all results discussed above are on the *level of logics*, i.e., each result concerns a class of TBoxes that is defined syntactically through expressibility in a certain logic, but no attempt is made to identify more structure *inside* these classes. The aim of this paper is to advocate a fresh look on the subject, by taking a novel approach. Specifically, we advocate a *non-uniform* study of the complexity of query answering by considering data complexity on the *level of individual TBoxes*. For a TBox $\mathcal{T}$, we say that *CQ-answering w.r.t. $\mathcal{T}$ is in* PTIME if for every CQ q, there is a PTIME algorithm that, given an ABox $\mathcal{A}$, computes the answers to q in $\mathcal{A}$ w.r.t. $\mathcal{T}$. In a similar way, we can define coNP-hardness and FO-rewritability on the TBox level. The non-uniform perspective allows us to investigate more fine-grained questions regarding the data complexity of query answering such as: given an expressive DL $\mathcal{L}$ such as $\mathcal{ALC}$ or $\mathcal{SHIQ}$, how can one characterize those $\mathcal{L}$-TBoxes $\mathcal{T}$ for which CQ-answering is in PTIME? How can we do the same for FO-rewritability? Is there a dichotomy for the complexity of query answering w.r.t. TBoxes formulated in $\mathcal{L}$, such as: for any $\mathcal{L}$-TBox $\mathcal{T}$, CQ-answering w.r.t. $\mathcal{T}$ is either in PTIME or coNP-hard?

In this paper, we consider TBoxes formulated in the expressive DL $\mathcal{ALCFI}$, answer some of the above questions, and take some steps towards others. Our main results are:

1. there is a dichotomy between PTIME and coNP-complete for CQ-answering w.r.t. $\mathcal{ALC}$-TBoxes if, and only if, Feder and Vardi's dichotomy conjecture that "constraint satisfaction problems (CSPs) with finite template are in PTIME or NP-complete" [10] is true; the same holds for $\mathcal{ALCI}$-TBoxes;
2. there is no dichotomy between PTIME and coNP-complete for CQ-answering w.r.t. $\mathcal{ALCF}$-TBoxes, unless PTIME $=$ NP; moreover, PTIME-complexity of CQ answering and many related problems are undecidable for $\mathcal{ALCF}$.
3. there is a dichotomy between PTIME and coNP-complete for CQ-answering w.r.t. $\mathcal{ALCFI}$-TBoxes of depth one, i.e., TBoxes where concepts have role depth ≤ 1;
4. FO-rewritability is decidable for Horn-$\mathcal{ALCFI}$-TBoxes of depth two and all Horn-$\mathcal{ALCF}$-TBoxes;

It should be noted that there has been steady progress regarding the dichotomy conjecture of Feder and Vardi over the last fifteen years and though the problem is still open, a solution does not seem completely out of reach [4, 5]. Our proof of Point 1 is based on a novel connection between CSPs and query answering w.r.t. $\mathcal{ALCI}$-TBoxes that can be exploited to transfer numerous results from the CSP world to query answering w.r.t. $\mathcal{ALCI}$-TBoxes and related problems. For example, together with [16, 5] we obtain the following results on 'FO-rewritability of ABox consistency':

5. Given an $\mathcal{ALCI}$-TBox $\mathcal{T}$, it can be decided in NEXPTIME whether there is an FO-sentence $\varphi_{\mathcal{T}}$ such that for all ABoxes $\mathcal{A}$, $\mathcal{A}$ is consistent w.r.t. $\mathcal{T}$ iff $\mathcal{A}$ viewed as an FO-structure satisfies $\varphi_{\mathcal{T}}$. Moreover, such a sentence $\varphi_{\mathcal{T}}$ exists iff ABox consistency w.r.t. $\mathcal{T}$ can be decided in non-uniform AC^0. Finally, if no such sentence $\varphi_{\mathcal{T}}$ exists, then ABox consistency w.r.t. $\mathcal{T}$ is LOGSPACE-hard (under FO-reductions).

To prove our results, we introduce some new notions that are relevant for studying the questions raised and prove some additional results of general interest. A central such notion is *materializability* of a TBox $\mathcal{T}$, which formalizes the existence of minimal models as known from Horn-DLs. We show that, in the case of TBoxes of depth one, materializability characterizes PTIME CQ-answering, which allows us to establish Point 2 above. For TBoxes of unrestricted depth, non-materializability still provides a sufficient condition for coNP-hardness of CQ-answering. We also develop the notion of *unraveling tolerance* of a TBox $\mathcal{T}$, which provides a sufficient condition for query answering to be in PTIME. The resulting upper bound strictly generalizes the known result that CQ-answering in Horn-$\mathcal{ALCFI}$ is in PTIME. Our framework also allows to formally establish some common intuitions and beliefs held in the context of CQ-answering in description logics. For example, we show that for any $\mathcal{ALCFI}$-TBox $\mathcal{T}$, CQ-answering is in PTIME iff answering positive existential queries is in PTIME iff answering $\mathcal{ELI}$-instance queries is in PTIME and likewise for FO-rewritability. Another observation in this spirit is that an $\mathcal{ALCFI}$-TBox is materializable (has minimal models) iff it is convex (a notion related to the entailment of disjunctions).

Most proofs in this paper are deferred to the (appendix of the) long version, which is available at http://www.csc.liv.ac.uk/~frank/publ/publ.html.

2 Preliminaries

We use standard notation for the syntax and semantics of $\mathcal{ALCFI}$ and other well-known DLs. Our TBoxes are finite sets of concept inclusions $C \sqsubseteq D$, where C and D are potentially compound concepts, and functionality assertions $\mathsf{func}(r)$, where r is a potentially inverse role. ABoxes are finite sets of assertions $A(a)$ and $r(a, b)$ with A a concept *name* and r a role name. We use $\mathsf{Ind}(\mathcal{A})$ to denote the set of individual names used in the ABox $\mathcal{A}$ and sometimes write $r^-(a, b) \in \mathcal{A}$ instead of $r(b, a) \in \mathcal{A}$. For the interpretation of individual names, we make the unique name assumption.

A *first-order query (FOQ)* $q(\boldsymbol{x})$ is a first-order formula with free variables $\boldsymbol{x}$ constructed from atoms $A(t)$, $r(t, t')$, and $t = t'$ (where t, t' range over individual names and variables) using negation, conjunction, disjunction, and existential quantification. The variables in $\boldsymbol{x}$ are the *answer variables* of q. A FOQ without answer variables is *Boolean*. We say that a tuple $\boldsymbol{a} \subseteq \mathsf{Ind}(\mathcal{A})$ is an *answer to* $q(\boldsymbol{x})$ *in an interpretation* $\mathcal{I}$ if $\mathcal{I} \models q[\boldsymbol{a}]$, where $q[\boldsymbol{a}]$ results from replacing the answer variables $\boldsymbol{x}$ in $q(\boldsymbol{x})$ with $\boldsymbol{a}$. A tuple $\boldsymbol{a} \subseteq \mathsf{Ind}(\mathcal{A})$ is a *certain answer to* $q(\boldsymbol{x})$ *in* $\mathcal{A}$ *given* $\mathcal{T}$, in symbols $\mathcal{T}, \mathcal{A} \models q(\boldsymbol{a})$, if $\mathcal{I} \models q[\boldsymbol{a}]$ for all models $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}$. Set $\mathsf{cert}_{\mathcal{T}}(q, \mathcal{A}) = \{\boldsymbol{a} \mid \mathcal{T}, \mathcal{A} \models q(\boldsymbol{a})\}$. A *positive existential query (PEQ)* $q(\boldsymbol{x})$ is a FOQ without negation and equality and a *conjunctive query (CQ)* is a positive existential query without disjunction. If C is an $\mathcal{ELI}$-concept and $a \in \mathsf{N_I}$, then $C(a)$ is an $\mathcal{ELI}$*-query (ELIQ)*. $\mathcal{EL}$*-queries (ELQs)* are defined analogously. Note that $\mathcal{ELI}$-queries and $\mathcal{EL}$-queries are always Boolean. In what follows, we sometimes slightly abuse notation and use FOQ to denote the set of all first-order queries, and likewise for CQ, PEQ, ELIQ, and ELQ.

Definition 1. *Let $\mathcal{T}$ be an $\mathcal{ALCFI}$-TBox. Let $\mathcal{Q} \in \{CQ, PEQ, ELIQ, ELQ\}$. Then*

- $\mathcal{Q}$-answering w.r.t. $\mathcal{T}$ is in PTIME *if for every $q(\boldsymbol{x}) \in \mathcal{Q}$, there is a polytime algorithm that computes, given an ABox $\mathcal{A}$, the answer* $\mathsf{cert}_{\mathcal{T}}(q, \mathcal{A})$*;*
- $\mathcal{Q}$-answering w.r.t. $\mathcal{T}$ is coNP-hard *if there is a Boolean $q \in \mathcal{Q}$ such that, given an ABox $\mathcal{A}$, it is* coNP*-hard to decide whether $\mathcal{T}, \mathcal{A} \models q$;*
- $\mathcal{T}$ *is* FO-rewritable for $\mathcal{Q}$ *iff for every $q(\boldsymbol{x}) \in \mathcal{Q}$ one can effectively construct an FO-formula $q'(\boldsymbol{x})$ such that for every ABox $\mathcal{A}$,* $\mathsf{cert}_{\mathcal{T}}(q, \mathcal{A}) = \{\boldsymbol{a} \mid \mathcal{I}_{\mathcal{A}} \models q'(\boldsymbol{a})\}$*, where $\mathcal{I}_{\mathcal{A}}$ denotes $\mathcal{A}$ viewed as an interpretation.*

The above notions of complexity are rather robust under changing the query language: as we show next, neither the PTIME bounds nor FO-rewritability depend on whether we consider PEQs, CQs, or ELIQs.

Theorem 1. *For all $\mathcal{ALCFI}$-TBoxes $\mathcal{T}$, the following equivalences hold:*

1. *CQ-answering w.r.t. $\mathcal{T}$ is in* PTIME *iff PEQ-answering w.r.t. $\mathcal{T}$ is in* PTIME *iff ELIQ-answering w.r.t. $\mathcal{T}$ is in* PTIME*;*
2. *$\mathcal{T}$ is FO-rewritable for CQ iff it is FO-rewritable for PEQ iff it is FO-rewritable for ELIQ.*

If $\mathcal{T}$ is an $\mathcal{ALCF}$-TBox, then we can replace ELIQ in Points 1 and 2 with ELQ.

The proof is based on Theorems 2 and 3 below. Theorem 1 allows us to (sometimes) speak of the 'complexity of query answering' without reference to a query language.

3 Materializability

An important tool we use for analyzing the complexity of query answering is the notion of materializability of a TBox $\mathcal{T}$, which means that computing the certain answers to any query q and ABox $\mathcal{A}$ w.r.t. $\mathcal{T}$ reduces to evaluating q in a single model of $\mathcal{A}$ and $\mathcal{T}$.

Definition 2. *Let $\mathcal{T}$ be an $\mathcal{ALCFI}$-TBox and $\mathcal{Q} \in \{CQ, PEQ, ELIQ, ELQ\}$. $\mathcal{T}$ is $\mathcal{Q}$-*materializable *if for every ABox $\mathcal{A}$ that is consistent w.r.t. $\mathcal{T}$, there exists a model $\mathcal{I}$ of $\mathcal{T}$ and $\mathcal{A}$ such that $\mathcal{I} \models q[\boldsymbol{a}]$ iff $\mathcal{T}, \mathcal{A} \models q(\boldsymbol{a})$ for all $q(\boldsymbol{x}) \in \mathcal{Q}$ and $\boldsymbol{a} \subseteq \mathsf{Ind}(\mathcal{A})$.*

We show that PEQ, CQ, and ELIQ-materializability coincide (and for $\mathcal{ALC}$-TBoxes, all these also coincide with ELQ-materializability). Materializability is also equivalent to the following disjunction property (sometimes also called *convexity*): a TBox $\mathcal{T}$ has the *ABox disjunction property* if for all ABoxes $\mathcal{A}$ and ELIQs $C_1(a_1), \ldots, C_n(a_n)$, from $\mathcal{T}, \mathcal{A} \models C_1(a_1) \vee \ldots \vee C_n(a_n)$ it follows that $\mathcal{T}, \mathcal{A} \models C_i(a_i)$, for some $i \leq n$.

Theorem 2. *Let $\mathcal{T}$ be an $\mathcal{ALCFI}$-TBox. The following equivalences hold: $\mathcal{T}$ is PEQ-materializable iff $\mathcal{T}$ is CQ-materializable iff $\mathcal{T}$ is ELIQ-materializable iff $\mathcal{T}$ has the ABox disjunction property.*

If $\mathcal{T}$ is an $\mathcal{ALC}$-TBox, the above are equivalent to ELQ-materializability.

Because of Theorem 2, we sometimes use the term materializability without reference to a query language. We call an interpretation $\mathcal{I}$ that satisfies the condition formulated in Definition 2 for PEQs a *minimal model* of $\mathcal{T}$ and $\mathcal{A}$. Note that in many cases, only an infinite minimal models exists. For example, for $\mathcal{T} = \{A \sqsubseteq \exists r.A\}$ and $\mathcal{A} = \{A(a)\}$ every minimal model $\mathcal{I}$ of $\mathcal{T}$ and $\mathcal{A}$ comprises an infinite r-chain starting at $a^{\mathcal{I}}$. Every TBox that is equivalent to an FO Horn sentence (in the general sense of [7]) is materializable: to construct a minimal model for such a TBox $\mathcal{T}$ and some ABox $\mathcal{A}$, one can take the direct product of all at most countable models of $\mathcal{T}$ and $\mathcal{A}$ (for additional information on direct products in DLs, see [17]). Conversely, however, there are simple materializable TBoxes that are not equivalent to FO Horn sentences.

Example 1. Let $\mathcal{T} = \{\exists r.(A \sqcap \neg B \sqcap \neg E) \sqsubseteq \exists r.(\neg A \sqcap \neg B \sqcap \neg E)\}$. One can easily show that $\mathcal{T}$ is not preserved under direct products; thus, it is not equivalent to a Horn sentence. However, one can construct a minimal model $\mathcal{I}$ for $\mathcal{T}$ and any ABox $\mathcal{A}$ by taking the interpretation $\mathcal{I}_{\mathcal{A}}$ obtained by viewing $\mathcal{A}$ as an interpretation and then adding, for any $a \in \mathsf{Ind}(\mathcal{A})$ with $a \in (\exists r.(A \sqcap \neg B \sqcap \neg E))^{\mathcal{I}_\mathcal{A}}$, a fresh d_a such that $(a, d_a) \in r^\mathcal{I}$ and d_a is not in the extension of any concept name. PEQ-answering w.r.t. $\mathcal{T}$ is FO-rewritable since for any PEQ q, $\mathsf{cert}_\mathcal{T}(q, \mathcal{A})$ consists of precisely the answers to q in $\mathcal{I}_\mathcal{A}$ (i.e., no query rewriting is necessary). Thus, PEQ-answering w.r.t. $\mathcal{T}$ is also in PTIME.

We show that materializability is a necessary condition for query answering being in PTIME.

Theorem 3. *If an $\mathcal{ALCFI}$-TBox $\mathcal{T}$ ($\mathcal{ALCF}$-TBox $\mathcal{T}$) is not materializable, then ELIQ-answering (ELQ-answering) is* coNP*-hard w.r.t. $\mathcal{T}$.*

The proof uses the violation of the ABox disjunction property stated in Theorem 2 and generalizes the reduction of 2+2-SAT used in [19] to prove that instance checking in a variant of $\mathcal{EL}$ is coNP-hard.

Materializability is not a sufficient condition for query answering to be in PTIME. In fact, we show that for any non-uniform constraint satisfaction problem, there is a materializable $\mathcal{ALC}$-TBox for which Boolean CQ-answering has the same complexity, up to complementation of the complexity class. For two finite relational FO-structures $\mathcal{R}$ and $\mathcal{R}'$ over relation symbols Σ, we write $\mathsf{Hom}(\mathcal{R}', \mathcal{R})$ if there is a homomorphism from $\mathcal{R}'$ to $\mathcal{R}$. The non-uniform constraint satisfaction problem for $\mathcal{R}$, denoted by $\mathsf{CSP}(\mathcal{R})$, is the problem to decide, for every finite $\mathcal{R}'$ over Σ, whether $\mathsf{Hom}(\mathcal{R}', \mathcal{R})$. Numerous algorithmic problems, among them many NP-complete ones such as k-SAT and k-colourability of graphs, can be given in the form $\mathsf{CSP}(\mathcal{R})$. It is known that every problem of the form $\mathsf{CSP}(\mathcal{R})$ is polynomially equivalent to some $\mathsf{CSP}(\mathcal{R}')$ with $\mathcal{R}'$ a digraph [10]. Thus, in what follows we can restrict ourselves to considering CSPs of the form $\mathsf{CSP}(\mathcal{I})$, where $\mathcal{I}$ is a DL interpretation. A *signature* Σ is a set of concept and role names. The signature $\mathsf{sig}(\mathcal{T})$ of a TBox $\mathcal{T}$ is the set of concept and role names that occur in $\mathcal{T}$. A Σ-TBox is a TBox that uses symbols from Σ only. Similar notation is used for ABoxes, concepts, and interpretations. For an ABox $\mathcal{A}$, we denote by $\mathcal{A}^\Sigma$ the subset of $\mathcal{A}$ containing symbols from Σ only. We will often not distinguish between ABoxes and finite interpretations.

Theorem 4. *For every non-uniform constraint satisfaction problem* $\mathsf{CSP}(\mathcal{I})$*, one can compute in polytime a materializable $\mathcal{ALC}$-TBox $\mathcal{T}$ such that for all ABoxes $\mathcal{A}$,*

1. $\mathsf{Hom}(\mathcal{A}^\Sigma, \mathcal{I})$*, with $\Sigma = \mathsf{sig}(\mathcal{I})$, iff $\mathcal{A}$ is consistent w.r.t. $\mathcal{T}$;*
2. *for any Boolean CQ q, answering q w.r.t. $\mathcal{T}$ is polynomially reducible to the complement of* $\mathsf{CSP}(\mathcal{I})$*.*

The proof Theorem 4 relies on the existence of $\mathcal{ALC}$-concepts H whose value $H^\mathcal{I}$ in interpretations $\mathcal{I}$ cannot be detected directly using CQs, but which can be used in a TBox to influence the values $A^\mathcal{I}$ of concept names A and, therefore, have an indirect effect on the answers to CQs. From the viewpoint of CQ query answering, they thus behave similarly to second-order variables. More precisely, let, for a finite set V of indices, Z_v, r_v, s_v be concept and role names, respectively. Let

$$\mathcal{T}_V = \{\top \sqsubseteq \exists r_v.\top, \top \sqsubseteq \exists s_v.Z_v \mid v \in V\}, \quad H_v = \forall r_v.\exists s_v.\neg Z_v.$$

Lemma 1. *For any ABox $\mathcal{A}$ and sets $I_v \subseteq \mathsf{Ind}(\mathcal{A})$, $v \in V$, one can construct a minimal model $\mathcal{I}$ of $(\mathcal{T}_V, \mathcal{A})$ such that $H_v^\mathcal{I} = I_v$ for all $v \in V$. $\mathcal{T}_V$ is FO-rewritable for PEQ.*

To prove Theorem 4, one extends the TBox $\mathcal{T}_V$. Assume $\mathsf{CSP}(\mathcal{I})$ is given. Let $V = \Delta^{\mathcal{I}}$ and assume, for simplicity, that $\mathsf{sig}(\mathcal{I}) = \{r\}$. Define

$$\mathcal{T} = \mathcal{T}_V \cup \{H_v \sqcap \exists r.H_w \sqsubseteq \bot \mid v, w \in V, (v, w) \notin r^{\mathcal{I}}\} \cup \\ \{H_v \sqcap H_w \sqsubseteq \bot \mid v, w \in V, v \neq w\} \cup \{\bigsqcap_{v \in V} \neg H_v \sqsubseteq \bot\}$$

Based on Lemma 1, it is possible to verify Points 1 and 2 of Theorem 4. For Point 2, it can be seen that for all Boolean CQs q and ABoxes $\mathcal{A}$, $(\mathcal{T}, \mathcal{A}) \models q$ iff $(\mathcal{T}_V, \mathcal{A}) \models q$ or not $\mathsf{Hom}(\mathcal{A}^{\Sigma}, \mathcal{I})$; since $\mathcal{T}_V$ is FO-rewritable, the former can be checked in PTIME.

4 (Towards) Dichotomies

We start with a reduction of Boolean CQ-answering w.r.t. $\mathcal{ALCI}$-TBoxes to CSPs that yields, together with Theorem 4, a proof of Point 1 in the introduction: the dichotomy problem for CSPs is equivalent to the dichotomy problem for CQ answering w.r.t. $\mathcal{ALC}$- (and $\mathcal{ALCI}$-) TBoxes.

Theorem 5. *Let $\mathcal{T}$ be an $\mathcal{ALCI}$-TBox and $C(a)$ an ELIQ. Then one can construct, in time exponential in $|\mathcal{T}| + |C|$,*

1. *a Σ-interpretation $\mathcal{I}$, $\Sigma = (\mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(C)) \uplus \{P\}$, with P a concept name, such that for all ABoxes $\mathcal{A}$,*
   - *(a) there is a polynomial reduction of answering $C(a)$ w.r.t. $\mathcal{T}$ to the complement of $\mathsf{CSP}(\mathcal{I})$;*
   - *(b) there is a polynomial reduction from the complement of $\mathsf{CSP}(\mathcal{I})$ to Boolean CQ-answering w.r.t. $\mathcal{T}$;*
2. *a Σ-interpretation $\mathcal{I}$, $\Sigma = \mathsf{sig}(\mathcal{T})$, such that for every ABox $\mathcal{A}$, $\mathcal{A}$ is consistent w.r.t. $\mathcal{T}$ iff $\mathsf{Hom}(\mathcal{A}^{\Sigma}, \mathcal{I})$.*

For Point 1, $\mathcal{I}$ is in fact the interpretation that is obtained by the standard type elimination procedure for $\mathcal{ALCI}$-TBoxes $\mathcal{T}$ and concepts C. More specifically, let S be the closure under single negation of all subconcepts of $\mathcal{T}$ and C. A *type* t is a maximal subset of S that is satisfiable w.r.t. $\mathcal{T}$. Then $\Delta^{\mathcal{I}}$ is the set of all types, $t \in A^{\mathcal{I}}$ iff $A \in t$, and $(t, t') \in r^{\mathcal{I}}$ iff $\forall r.D \in t$ implies $D \in t'$ and $\forall r^{-}.D \in t'$ implies $D \in t$. For the special concept name P, set $P^{\mathcal{I}} = \{t \mid C \notin t\}$. With the type elimination algorithm, $\mathcal{I}$ can be constructed in exponential time. The mentioned reductions are then as follows:

(a) $(\mathcal{T}, \mathcal{A}) \models C(a)$ iff not $\mathsf{Hom}(\mathcal{A}^{\Sigma}_{P(a)}, \mathcal{I})$, where $\mathcal{A}_{P(a)}$ results from $\mathcal{A}$ by adding $P(a)$ to $\mathcal{A}$ and removing all other assertions using P from $\mathcal{A}$;
(b) not $\mathsf{Hom}(\mathcal{A}^{\Sigma}, \mathcal{I})$ iff $(\mathcal{T}, \mathcal{A}) \models \exists v.(P(v) \wedge C(v))$.

Result 1 from the introduction can be derived as follows. Let $\mathsf{CSP}(\mathcal{I})$ be an NP-intermediate CSP, i.e., a CSP that is neither in PTIME nor NP-hard. Take the TBox $\mathcal{T}$ from Theorem 4. By Point 1 of that theorem and since consistency of ABoxes w.r.t. $\mathcal{T}$ can trivially be reduced to the complement of answering Boolean CQs w.r.t. $\mathcal{T}$, CQ-answering w.r.t. $\mathcal{T}$ is not in PTIME. By Point 2, CQ-answering w.r.t. $\mathcal{T}$ is not coNP-hard either. Conversely, let $\mathcal{T}$ be a TBox for which CQ-answering w.r.t. $\mathcal{T}$ is neither in

PTIME nor coNP-hard. Then by Theorem 1 and since every ELIQ is a CQ, the same holds for ELIQ-answering w.r.t. $\mathcal{T}$. Thus, there is a concrete ELIQ $C(a)$ such that answering $C(a)$ w.r.t. $\mathcal{T}$ is coNP-intermediate. Let $\mathcal{I}$ be the interpretation constructed in Point 1 of Theorem 5 for $\mathcal{T}$ and $C(a)$. By Point 1a, $\mathsf{CSP}(\mathcal{I})$ is not in PTIME; by Point 1b, it is not NP-hard either.

Result 5 from the introduction can be derived as follows. It is proved in [16, 5] that the problem to decide whether the class of structures $\{\mathcal{I}' \mid \mathsf{Hom}(\mathcal{I}', \mathcal{I})\}$ is FO-definable is NP-complete. We obtain a NEXPTIME upper bound since the template $\mathcal{I}$ associated with $\mathcal{T}$ can be constructed in exponential time. The claims for AC^0 and LOGSPACE follow in the same way from other results in [16, 5].

We now develop a condition on TBoxes, called unraveling tolerance, that is sufficient for PTIME CQ-answering and strictly generalizes Horn-$\mathcal{ALCFI}$, the $\mathcal{ALCFI}$-fragment of Horn-$\mathcal{SHIQ}$. For the case of TBoxes of depth one, we obtain a PTIME/coNP dichotomy result. The notion of unraveling tolerance is based on an unraveling operation on ABoxes, in the same spirit as the well-known unraveling of an interpretation into a tree interpretation. This is inspired by (i) the observation that, in the proof of Theorem 3, the non-tree-shape of ABoxes is essential; and (ii) by Theorem 5 together with the known fact the non-uniform CSPs are tractable when restricted to tree-shaped input structures. The *unraveling* $\mathcal{A}_u$ of an ABox $\mathcal{A}$ is the following ABox:

- the individual names $\mathsf{Ind}(\mathcal{A}_u)$ of $\mathcal{A}_u$ are sequences $b_0 r_0 b_1 \cdots r_{n-1} b_n$, $b_0, \ldots, b_n \in \mathsf{Ind}(\mathcal{A})$ and $r_0, \ldots, r_{n-1}$ (possibly inverse) roles such that for all $i < n$, we have $r_i(b_i, b_{i+1}) \in \mathcal{A}$ and $b_{i+1} \neq b_{i-1}$ (whenever $i > 0$);
- for each $C(b) \in \mathcal{A}$ and $\alpha = b_0 r_0 b_1 \cdots r_{n-1} b_n \in \mathsf{Ind}(\mathcal{A}_u)$ with $b_n = b$, we have $C(\alpha) \in \mathcal{A}_u$;
- for each $b_0 r_0 b_1 \cdots r_{n-1} b_n \in \mathsf{Ind}(\mathcal{A}_u)$, we have $r_{n-1}(b_{n-1}, b_n) \in \mathcal{A}_u$.

For all $\beta = b_0 r_0 \cdots r_{n-1} b_n \in \mathsf{Ind}(\mathcal{A}_u)$, we write $\mathsf{tail}(\beta)$ to denote b_n. Note that the condition $b_{i+1} \neq b_{i-1}$ is needed to ensure that functional roles can still be interpreted in a functional way after unraveling, despite the UNA.

Definition 3. *A TBox $\mathcal{T}$ is* unraveling tolerant *if for all ABoxes $\mathcal{A}$ and ELIQs q, we have that $\mathcal{T}, \mathcal{A} \models q$ implies $\mathcal{T}, \mathcal{A}_u \models q$.*

It is not hard to prove that the converse direction '$\mathcal{T}, \mathcal{A}_u \models q$ implies $\mathcal{T}, \mathcal{A} \models q$' is true for *all* $\mathcal{ALCFI}$-TBoxes. We now show that the class of unraveling tolerant $\mathcal{ALCFI}$-TBoxes generalizes Horn-$\mathcal{ALCFI}$. This is based on the original and most general definition of Horn-$\mathcal{SHIQ}$ in [12] and thus also captures weaker variants as used e.g. in [13, 9]. The TBox in Example 1, which is unraveling tolerant but not a Horn-$\mathcal{ALCFI}$-TBox, demonstrates that the generalization is strict.

Lemma 2. *Every Horn-$\mathcal{ALCFI}$-TBox is unraveling tolerant.*

It is interesting to note that unraveling tolerance implies materializability. We shall see that the converse is, in general, not true.

Lemma 3. *Every unraveling-tolerant $\mathcal{ALCFI}$-TBox is materializable.*

We now show that unraveling tolerance yields a class of $\mathcal{ALCFI}$-TBoxes for which query answering is in PTIME. By Lemma 2 and since we actually exhibit a *uniform* algorithm for query answering w.r.t. unraveling tolerant TBoxes, this also reproves the known PTIME upper bound for CQ-answering in Horn-$\mathcal{ALCFI}$ [9]. This result is not a consquence of Theorem 4 and known results for CSPs since we capture full $\mathcal{ALCFI}$.

Theorem 6. *If an $\mathcal{ALCFI}$-TBox $\mathcal{T}$ is unraveling tolerant, then PEQ-answering w.r.t. $\mathcal{T}$ is in* PTIME.

To see that unraveling tolerance does not capture all $\mathcal{ALCFI}$-TBoxes for which query answering is in PTIME, we can invoke Theorem 4. For example, taking a CSP for 2-colorability, we obtain a TBox $\mathcal{T}$ for which CQ-answering is in PTIME and such that an ABox $\mathcal{A}$ with $\mathsf{sig}(\mathcal{A}) = \{r\}$ is consistent w.r.t. $\mathcal{T}$ iff $\mathcal{A}$ is 2-colorable. Thus, $\mathcal{A}, \mathcal{T} \models X(a)$, X a fresh concept name, iff $\mathcal{A}$ is not 2-colorable. It follows that $\mathcal{T}$ is not unraveling tolerant. We conjecture that it is possible to generalize Theorem 6 to larger classes of TBoxes by relaxing the operation of ABox unraveling such that it yields ABoxes of bounded treewidth instead of tree-shaped ABoxes. Such a generalization would still not capture 2-colorability.

We now turn to TBoxes of depth one. The central observation is that for this special case, we can prove a converse of Lemma 3.

Lemma 4. *Every materializable $\mathcal{ALCFI}$-TBox of depth one is unraveling tolerant.*

This brings us into the position where we can establish the announced dichotomy result for $\mathcal{ALCFI}$-TBoxes of depth one. If such a TBox $\mathcal{T}$ is materializable, then Lemma 4 and Theorem 6 yield that PEQ-answering w.r.t. $\mathcal{T}$ is in PTIME. Otherwise, ELIQ-answering w.r.t. $\mathcal{T}$ is coNP-complete by Theorem 3. We thus obtain the following.

Theorem 7 (Dichotomy). *For every $\mathcal{ALCFI}$-TBox $\mathcal{T}$ of depth one, one of the following is true:*

- *$\mathcal{Q}$-answering w.r.t. $\mathcal{T}$ is in* PTIME *for any $\mathcal{Q} \in \{PEQ, CQ, ELIQ\}$;*
- *$\mathcal{Q}$-answering w.r.t. $\mathcal{T}$ is* coNP-*complete for any $\mathcal{Q} \in \{PEQ, CQ, ELIQ\}$.*

5 Deciding FO-Rewritability

The results of this section are based on the observation that for materializable TBoxes of depth one, FO-rewritability for CQ follows from FO-rewritability for *atomic* concepts, i.e., concept names and $\bot$. We say that an atomic concept A is *FO-rewritable w.r.t. a TBox $\mathcal{T}$ and a signature Σ* if there exists an FO-formula φ_A such that for all Σ-ABoxes $\mathcal{A}$ and $a \in \mathsf{Ind}(\mathcal{A})$: $\mathcal{T}, \mathcal{A} \models A(a)$ iff $\mathcal{I}_\mathcal{A} \models \varphi_A[a]$. Clearly, if $\mathcal{T}$ is FO-rewritable for CQ, then every atomic concept is FO-rewritable w.r.t. $\mathcal{T}$ and any signature. For materializable TBoxes of depth one, the converse is also true.

Lemma 5. *A materializable $\mathcal{ALCFI}$-TBox of depth one is FO-rewritable for CQs iff all atomic concepts are FO-rewritable w.r.t. $\mathcal{T}$ and* $\mathsf{sig}(\mathcal{T})$.

Based on Lemma 5, we can use Theorem 5 and results from [16] to obtain the following result, in a similar (but slightly more involved) way as in the proof of Result 5 from the introduction.

Theorem 8. *FO-rewritability for CQs is decidable in* NEXPTIME*, for any of the following classes of TBoxes: materializable $\mathcal{ALCI}$-TBoxes of depth one, Horn-$\mathcal{ALC}$-TBoxes, and Horn-$\mathcal{ALCI}$-TBoxes of depth two.*

Theorem 5 does not apply to DLs with functional roles. To analyze FO-rewritability in the presence of functional roles, we associate with every materializable TBox $\mathcal{T}$ of depth one a monadic datalog program $\Pi_{\mathcal{T}}$ such that $\mathcal{T}$ and $\Pi_{\mathcal{T}}$ give the same answers to queries $A(a)$, A atomic. We then show that $\mathcal{T}$ is FO-rewritable if, and only if, $\Pi_{\mathcal{T}}$ is equivalent to a non-recursive datalog program. The latter property is known as *boundedness* of a datalog program and has been studied extensively for fixpoint logics [3, 18] and datalog programs [8]. Using existing decidability results for boundedness, we can then establish a counterpart of Theorem 8 for the case of $\mathcal{ALCFI}$.

For our purposes, a monadic datalog program Π consists of rules $A(x) \leftarrow X$, where A is a concept name and X is a finite set consisting of assertions of the form $B(x)$, $r(x_1, x_2)$, and inequalities $x_1 \neq x_2$, where B is a concept name, r a role, and x, x_1, x_2 range over variables. Inequalities are required to model functional roles. We also use a special unary predicate $\bot$ and rules $\bot(x) \leftarrow X$ stating that X is inconsistent. For an ABox $\mathcal{A}$, we denote by $\Pi^i(\mathcal{A})$ the set of all assertions $A(a)$ that can be derived using i applications of rules from Π to $\mathcal{A}$. We set $\Pi^\infty(\mathcal{A}) = \bigcup_{i\geq 0} \Pi^i(\mathcal{A})$.

Definition 4 (Boundedness). *Let Π be a datalog program and Σ a signature. An atomic concept A* is bounded in Π for Σ-ABoxes *if there exists a $k > 0$ such that for all Σ-ABoxes $\mathcal{A}$ and all $a \in \mathsf{sig}(\mathcal{A})$: $A(a) \in \Pi^\infty(\mathcal{A})$ iff $A(a) \in \Pi^k(\mathcal{A})$.*

Let $\mathcal{T}$ be a materializable TBox of depth one. A Σ-neighbourhood ABox (Σ-NH) consists of a Σ-ABox $\mathcal{A}$ with a distinguished individual name f such that $\mathcal{A}$ consists of assertions of the form $r(f, a)$ with r a role and $a \neq f$ and $A(b)$ such that

- for each $b \neq f$ with $b \in \mathsf{Ind}(\mathcal{A})$ there is exactly one r such that $r(f, b) \in \mathcal{A}$;
- if $r(f, b_1)$ and $r(f, b_2) \in \mathcal{A}$ and $b_1 \neq b_2$, then there exists $A(b_1) \in \mathcal{A}$ with $A(b_2) \notin \mathcal{A}$ or vice versa.

The ABox $\mathcal{A}$ in which each individual b is replaced by a variable x_b is denoted by $\mathcal{A}^x$. Now define a monadic datalog program associated with $\mathcal{T}$, where $\Sigma = \mathsf{sig}(\mathcal{T})$:

$$\begin{aligned}\Pi_{\mathcal{T}} = \{&A(x_a) \leftarrow \mathcal{A}^x \mid \mathcal{A} \text{ is a } \Sigma\text{-NH}, a \in \mathsf{Ind}(\mathcal{A}), A \in \Sigma, (\mathcal{T}, \mathcal{A}) \models A(a)\} \cup \\ \{&\bot(x) \leftarrow \mathcal{A}^x \mid \mathcal{A} \text{ is a } \Sigma\text{-NH that is not consistent w.r.t. } \mathcal{T}\} \cup \\ \{&\bot(x) \leftarrow r(y, y_1), r(y, y_2), y_1 \neq y_2 \mid \mathsf{func}(r) \in \mathcal{T}\} \cup \\ \{&A(x) \leftarrow \bot(x) \mid A \in \Sigma\}.\end{aligned}$$

The following lemma states that $\Pi_{\mathcal{T}}$ behaves as intended.

Lemma 6. *For every materializable $\mathcal{ALCFI}$-TBox $\mathcal{T}$ of depth one, every $A \in \mathsf{sig}(\mathcal{T})$, every ABox $\mathcal{A}$, and every $a \in \mathsf{Ind}(\mathcal{A})$, $(\mathcal{T}, \mathcal{A}) \models A(a)$ iff $A(a) \in \Pi^\infty_{\mathcal{T}}(\mathcal{A})$. Moreover, $\bot(a) \in \Pi^\infty_{\mathcal{T}}(\mathcal{A})$ iff $\mathcal{A}$ is not consistent w.r.t. $\mathcal{T}$.*

Using unfolding tolerance of materializable TBoxes of depth one, one can show the following equivalence for FO-rewritability and boundedness.

Lemma 7. *For every materializable $\mathcal{ALCFI}$-TBox $\mathcal{T}$ of depth one and signature Σ: an atomic concept A is bounded in $\Pi_{\mathcal{T}}$ for Σ-ABoxes iff A is FO-rewritable w.r.t. $\mathcal{T}$ and Σ.*

Unfortunately, decidability results for boundedness of monadic datalog programs are not directly applicable to $\Pi_{\mathcal{T}}$ since they assume programs without inequalities [8, 11]. However, using unfolding tolerance, one can employ instead recent decidability results on boundedness of least fixed points over trees [18] to obtain the following theorem.

Theorem 9. *FO-rewritability for CQs is decidable, for any of the following classes of TBoxes: materializable $\mathcal{ALCFI}$-TBoxes of depth one, Horn-$\mathcal{ALCF}$-TBoxes, and Horn-$\mathcal{ALCFI}$-TBoxes of depth two.*

6 Non-Dichotomy and Undecidability in $\mathcal{ALCF}$

The aim of this section is to show that the addition of functional roles significantly complicates the problems studied in the previous sections. More precisely, we show that (i) for CQ-answering w.r.t. $\mathcal{ALCF}$-TBoxes, there is no dichotomy between PTIME and coNP unless PTIME = NP; and (ii) CQ-answering in PTIME is undecidable for $\mathcal{ALCF}$-TBoxes, and likewise for coNP-hardness, materializability and FO-rewritability. Point (i) is a consequence of the following result.

Theorem 10. *For every language L in* coNP*, there is an $\mathcal{ALCF}$-TBox $\mathcal{T}$ and query* $\mathsf{rej}(a)$*,* rej *a concept name, such that the following holds:*

1. *there exists a polynomial reduction of deciding $v \in L$ to answering* $\mathsf{rej}(a)$ *w.r.t. $\mathcal{T}$;*
2. *for every ELIQ q, answering q w.r.t. $\mathcal{T}$ is polynomially reducible to deciding $v \in L$.*

Ladners theorem [15] states that unless PTIME = NP, coNP intermediate problems exist. Suppose to the contrary of Point (i) that for every $\mathcal{ALCF}$-TBox $\mathcal{T}$, CQ answering w.r.t. $\mathcal{T}$ is in PTIME or coNP-hard. Take a coNP-intermediate language L and let $\mathcal{T}$ be the TBox from Theorem 10. By Point 1 of the theorem, CQ-answering w.r.t. $\mathcal{T}$ is not in PTIME. Thus it must be coNP-hard. By Theorem 1 and since a dichotomy for CQ-answering w.r.t. $\mathcal{T}$ also implies a dichotomy for ELIQ-answering w.r.t. $\mathcal{T}$, ELIQ-answering w.r.t. $\mathcal{T}$ is also coNP-hard. By Point 2 of Theorem 10, this is impossible.

The proof of Theorem 10 combines the 'hidden' concepts H_v from the proof of Theorem 4 with ideas from a proof in [1] which establishes undecidability of a certain *query emptiness* problem in $\mathcal{ALCF}$. Using a similar strategy, we establish the undecidability results announced as Point (ii) above, summarized by the following theorem.

Theorem 11. *For $\mathcal{ALCF}$-TBoxes $\mathcal{T}$, the following problems are undecidable (Points 1 and 2 are subject to the side condition that* PTIME $\neq$ NP*):*

1. *CQ-answering w.r.t. $\mathcal{T}$ is in* PTIME*;*
2. *CQ answering w.r.t. $\mathcal{T}$ is* coNP*-hard;*
3. *$\mathcal{T}$ is materializable.*

In the appendix, we also prove that FO-rewritability for CQ is undecidable in $\mathcal{ALCF}$, for a slightly modified definition of FO-rewritability that only considers *consistent* ABoxes.

7 Conclusions

We have have introduced non-uniform data complexity of query answering w.r.t. description logic TBoxes and proved that it enables a more fine-grained analysis than the standard approach. Many questions remain. In particular, the newly established CSP-connection should be exploited further. We believe that the techniques introduced in this paper can be extended to richer DLs such as $\mathcal{SHIQ}$.

Acknowledgments. C. Lutz was supported by the DFG SFB/TR 8 "Spatial Cognition".

References

1. F. Baader, M. Bienvenu, C. Lutz, and F. Wolter. Query and predicate emptiness in description logics. In *Proc. of KR2010*. AAAI Press, 2010.
2. F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proc. of IJCAI05*, pages 364–369. Professional Book Center, 2005.
3. J. Barwise and Y. N. Moschovakis. Global inductive definability. *J. Symb. Log.*, 43(3):521–534, 1978.
4. A. A. Bulatov. A dichotomy theorem for constraints on a three-element set. In *Proc. of FOCS02*, pages 649–658, 2002.
5. A. A. Bulatov, A. A. Krokhin, and B. Larose. Dualities for constraint satisfaction problems. In *Complexity of Constraints*, pages 93–124, 2008.
6. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. of Autom. Reasoning*, 39(3):385–429, 2007.
7. C. C. Chang and H. J. Keisler. *Model Theory*, volume 73 of *Studies in Logic and the Foundations of Mathematics*. Elsevier, 1990.
8. S. S. Cosmadakis, H. Gaifman, P. C. Kanellakis, and M. Y. Vardi. Decidable optimization problems for database logic programs (preliminary report). In *Proc. of STOC88*, pages 477–490, 1988.
9. T. Eiter, G. Gottlob, M. Ortiz, and M. Simkus. Query answering in the description logic Horn-$\mathcal{SHIQ}$. In *Proc. of JELIA08*, volume 5293 of LNCS, pages 166–179. Springer, 2008.
10. T. Feder and M. Y. Vardi. Monotone monadic snp and constraint satisfaction. In *Proc. of STOC93*, pages 612–622, 1993.
11. H. Gaifman, H. G. Mairson, Y. Sagiv, and M. Y. Vardi. Undecidable optimization problems for database logic programs. In *Proc. of LICS87*, pages 106–115, 1987.
12. U. Hustadt, B. Motik, and U. Sattler. Reasoning in description logics by a reduction to disjunctive datalog. *J. Autom. Reasoning*, 39(3):351–384, 2007.
13. Y. Kazakov. Consequence-driven reasoning for horn SHIQ ontologies. In *Proc. of IJCAI09*, pages 2040–2045, 2009.
14. M. Krötzsch. Efficient inferencing for OWL EL. In *Proc. of JELIA10*, volume 6341 of LNCS, pages 234–246. Springer, 2010.
15. R. E. Ladner. On the structure of polynomial time reducibility. *JACM*, 22(1):155171, 1975.
16. B. Larose, C. Loten, and C. Tardif. A characterisation of first-order constraint satisfaction problems. *Logical Methods in Computer Science*, 3(4), 2007.
17. C. Lutz, R. Piro, and F. Wolter. Description logic tboxes: Model-theoretic characterizations and rewritability. In *Proc. of IJCAI11*, 2011.
18. M. Otto, A. Blumensath, and M. Weyer. Decidability results for the boundedness problem. Technical report, TU Darmstadt, 2010.
19. A. Schaerf. On the complexity of the instance checking problem in concept languages with existential quantification. *J. of Intell. Inf. Sys.*, 2:265–278, 1993.