Non-Uniform Data Complexity of Query Answering in Description Logics

Carsten Lutz
Fachbereich Informatik
Universität Bremen, Germany

Frank Wolter
Department of Computer Science
University of Liverpool, UK

Abstract

In ontology-based data access (OBDA), ontologies are used as an interface for querying instance data. Since in typical applications the size of the data is much larger than the size of the ontology and query, data complexity is the most important complexity measure. In this paper, we propose a new method for investigating data complexity in OBDA: instead of classifying whole logics according to their complexity, we aim at classifying *each individual ontology* within a given master language. Our results include a P/coNP-dichotomy theorem for ontologies of depth one in the description logic $\mathcal{ALCFI}$, the equivalence of a P/coNP-dichotomy theorem for $\mathcal{ALC}$/$\mathcal{ALCI}$-ontologies of unrestricted depth to the famous dichotomy conjecture for CSPs by Feder and Vardi, and a non-P/coNP-dichotomy theorem for $\mathcal{ALCF}$-ontologies.

1 Introduction

In recent years, the use of ontologies to access instance data has become increasingly popular (Poggi et al. 2008; Dolby et al. 2008). The general idea is that an ontology provides an enriched vocabulary or conceptual model for the application domain, thus serving as an interface for querying instance data and allowing to derive additional facts. In this emerging area, called ontology-based data access, 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 and thus intractable (when speaking of complexity, we *always* mean data complexity; references are given at the end of this section). 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. Horn DLs in this sense include 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.

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 in a syntactic way in terms of 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 initiate a *non-uniform* study of the complexity of query answering by considering data complexity on the *level of individual TBoxes*. We say that CQ-answering w.r.t. a TBox $\mathcal{T}$ is in PTime if for every CQ q, there is a PTime algorithm that computes, given an ABox $\mathcal{A}$, the answers to q in $\mathcal{A}$ w.r.t. $\mathcal{T}$; CQ-answering w.r.t. $\mathcal{T}$ is coNP-hard if there exists a Boolean CQ q such that it is coNP-hard to answer q in ABoxes $\mathcal{A}$ w.r.t. $\mathcal{T}$. Other complexities can be defined similarly. The ultimate goal of our approach is as follows:

For a fixed master DL $\mathcal{L}$, classify all TBoxes $\mathcal{T}$ in $\mathcal{L}$ according to the complexity of CQ-answering w.r.t. $\mathcal{T}$.

In this paper, we consider as master DLs the basic expressive DL $\mathcal{ALC}$, its extensions $\mathcal{ALCI}$ with inverse roles and $\mathcal{ALCF}$ with functional roles, and their union $\mathcal{ALCFI}$. It turns out that, even for $\mathcal{ALC}$, fully achieving the above goal is far beyond the scope of a single research paper. In fact, we show that a full classification of the complexity of $\mathcal{ALC}$-TBoxes is essentially equivalent to a full classification of the complexity of non-uniform constraint satisfaction problems with finite templates (CSPs). The latter is a major research programme ongoing for many years that combines complexity theory, graph theory, logic, and algebra; see below for references and additional details.

In the current paper, we mainly concentrate on understanding the boundary between PTime and coNP-hardness of CQ-answering w.r.t. DL TBoxes, mostly neglecting other relevant classes such as AC^0, LogSpace, and NLogSpace. Our main results are as follows.

1. There is a PTIME/coNP-dichotomy for CQ-answering w.r.t. $\mathcal{ALCFI}$-TBoxes of depth one, i.e., TBoxes in which existential/universal restrictions are not nested.

The proof introduces model-theoretic characterizations of polytime CQ-answering which are discussed below. Note that this is a relevant case since most TBoxes from practical applications have depth one. In particular, all TBoxes formulated in DL-Lite and its extensions proposed in (Calvanese et al. 2006; Artale et al. 2009) have depth one, and the same is true for more than 85 percent of all TBoxes in the TONES ontology repository (http://owl.cs.manchester.ac.uk/repository/).

2. There is a PTIME/coNP-dichotomy for CQ-answering w.r.t. $\mathcal{ALC}$-TBoxes if and only if Feder and Vardi's dichotomy conjecture for CSPs is true; the same holds for $\mathcal{ALCI}$-TBoxes.

The proof of this result establishes the close link between CQ-answering in $\mathcal{ALC}$ and CSP that was mentioned above. While dichotomy questions are mainly of theoretical interest, linking these two worlds is potentially very relevant also for applied DL research.

3. There is no PTIME/CONP-dichotomy for CQ-answering w.r.t. $\mathcal{ALCF}$-TBoxes (unless PTIME = NP).

This is proved by showing that, for every problem in coNP, there is an $\mathcal{ALCF}$-TBox for which CQ-answering has the same complexity (up to polytime reductions); it then remains to apply Ladner's Theorem, which guarantees the existence of NP-intermediate problems. Consequently, we cannot expect an exhaustive classification of the complexity of CQ-answering w.r.t. $\mathcal{ALCF}$-TBoxes.

To prove these results, we introduce two new notions that are of independent interest and general utility. The first one is *materializability* of a TBox $\mathcal{T}$, which means that answering a CQ over an ABox $\mathcal{A}$ w.r.t. $\mathcal{T}$ can be reduced to query evaluation in a single model of $\mathcal{A}$ and $\mathcal{T}$. Note that such models play a crucial role in the context of Horn DLs, where they are often called least models or canonical models. In contrast to the Horn DL case, however, we only require the *existence* of such a model without making any assumptions about its form or construction.

4. If an $\mathcal{ALCFI}$-TBox $\mathcal{T}$ is not materializable, then CQ-answering w.r.t. $\mathcal{T}$ is CONP-hard.

Perhaps in contrary to the intuitions that arise from the experience with Horn-DLs, materializability of a TBox $\mathcal{T}$ is *not* a sufficient condition for CQ-answering w.r.t. $\mathcal{T}$ to be in PTIME (unless PTIME = NP). This leads us to study the notion of *unraveling tolerance* of a TBox $\mathcal{T}$, meaning that answers to tree-shaped CQs over an ABox $\mathcal{A}$ w.r.t. $\mathcal{T}$ are preserved under unraveling the ABox $\mathcal{A}$. In CSP, unraveling tolerance corresponds to the existence of tree obstructions, a notion that characterizes the well-known arc consistency condition (Krokhin 2010; Dechter 2003). It can be shown that every TBox formulated in Horn-$\mathcal{ALCFI}$ (the intersection of $\mathcal{ALCFI}$ and Horn-$\mathcal{SHIQ}$) is unraveling tolerant and that there are unraveling tolerant TBoxes which are not equivalent to any Horn-$\mathcal{ALCFI}$-TBox. Thus, the following result yields a rather general (and uniform!) PTIME upper bound for CQ-answering.

5. If an $\mathcal{ALCFI}$-TBox $\mathcal{T}$ is unraveling tolerant, then CQ-answering w.r.t. $\mathcal{T}$ is in PTIME.

Although the above result is rather general, unraveling tolerance of a TBox $\mathcal{T}$ is *not* a necessary condition for CQ-answering w.r.t. $\mathcal{T}$ to be in PTIME (unless PTIME = NP). However, for $\mathcal{ALCFI}$-TBoxes $\mathcal{T}$ *of depth one*, being materializable and being unraveling tolerant turns out to be equivalent. We thus obtain that CQ-answering w.r.t. $\mathcal{T}$ is in PTIME iff $\mathcal{T}$ is materializable iff $\mathcal{T}$ is unraveling tolerant while, otherwise, CQ-answering w.r.t. $\mathcal{T}$ is CONP-hard. This establishes the first main result above.

Our framework also allows to formally capture some intuitions and beliefs commonly held in the context of CQ-answering in DLs. For example, we show that for every $\mathcal{ALCFI}$-TBox $\mathcal{T}$, CQ-answering is in PTIME iff answering positive existential queries is in PTIME iff answering $\mathcal{ELI}$-instance queries (tree-shaped CQs) is in PTIME. This implies that all results mentioned above apply not only to CQ answering, but also to answering queries in any of these other languages. In fact, the use of multiple query languages and in particular of $\mathcal{ELI}$-instance queries does not only yield additional results, but is also at the heart of our proof strategies, which would not work for CQs alone.

Another interesting observation in this spirit is that an $\mathcal{ALCFI}$-TBox is materializable iff it is convex, a condition that is also called the disjunction property and plays a central role in attaining PTIME complexity for standard reasoning in Horn DLs such as $\mathcal{EL}$, DL-Lite, and Horn-$\mathcal{SHIQ}$; see for example (Baader, Brandt, and Lutz 2005; Krisnadhi and Lutz 2007) for more details.

Most proofs are deferred to the long version, available at http://www.csc.liv.ac.uk/~frank/publ/publ.html.

Related Work

An early reference on data complexity in DLs is (Schaerf 1993), showing CONP-hardness of instance queries in the moderately expressive DL $\mathcal{ALE}$. A CONP upper bound for instance queries in the much more expressive $\mathcal{SHIQ}$ was obtained in (Hustadt, Motik, and Sattler 2007) and generalized to CQs in (Glimm et al. 2008). Horn-$\mathcal{SHIQ}$ was first defined in (Hustadt, Motik, and Sattler 2007), where also a PTIME upper bound for instance queries is established; the generalization to CQs can be found in (Eiter et al. 2008). See also (Krisnadhi and Lutz 2007; Calvanese et al. 2006) and references therein for data complexity in DLs and (Barany, Gottlob, and Otto 2010; Baget et al. 2011) for related work beyond standard DLs.

To the best of our knowledge, the current paper presents the first study of data complexity in OBDA at the level of individual TBoxes and the first formal link between OBDA and CSP. There is, however, a vague technical similarity to the link between view-based query processing for regular path queries (RPQs) and CSP found in (Calvanese et al. 2000; 2003b; 2003a). In this case, the recognition problem for perfect rewritings for RPQs can be polynomially reduced to non-uniform CSP and vice versa.

The work on CSP dichotomies started with Schaefer's PTIME/NP-dichotomy theorem, stating that every binary CSP is in PTIME or NP-hard (Schaefer 1978). Here, a binary CSP is defined by a relational structure $\mathcal{B}$ whose domain consists of two elements and the problem is to decide for a given relational structure $\mathcal{C}$ over the same relation symbols, whether there is a homomorphism from $\mathcal{C}$ to $\mathcal{B}$. To appreciate Schaefer's result, recall that Ladners theorem guarantees, in general, the existence of problems that are NP-intermediate and thus neither in PTIME nor NP-hard, unless PTIME = NP (Ladner 1975). Schaefer's theorem was followed by a dichotomy result for CSPs with graph templates (Hell and Nesetril 1990) and the seminal Feder-Vardi PTIME/NP-dichotomy conjecture for all CSPs (Feder and Vardi 1993), confirmed for ternary CSPs in (Bulatov 2002). Interesting results have also been obtained for other complexity classes such as AC^0 (Allender et al. 2005; Larose, Loten, and Tardif 2007). The state of the art is summarized, for example, in (Bulatov, Jeavons, and Krokhin 2005; Kun and Szegedy 2009; Bulatov 2011).

2 Preliminaries

We start with introducing the DL $\mathcal{ALC}$ and its extensions $\mathcal{ALCI}$ and $\mathcal{ALCFI}$. As usual, we use $\mathsf{N_C}$, $\mathsf{N_R}$, and $\mathsf{N_I}$ to denote countably infinite sets of *concept names*, *role names*, and *individual names*, respectively. *$\mathcal{ALC}$-concepts* are constructed according to the rule

$$C, D \ := \ \top \mid \bot \mid A \mid C \sqcap D \mid C \sqcup D \mid \neg C \mid \exists r.C \mid \forall r.C$$

where A ranges over $\mathsf{N_C}$ and r ranges over $\mathsf{N_R}$. *$\mathcal{ALCI}$-concepts* admit, in addition, *inverse roles* from the set $\mathsf{N_R^-} = \{r^- \mid r \in \mathsf{N_R}\}$. To avoid heavy notation, we set $r^- = s$ if $r = s^-$ for a role name s. An *$\mathcal{ALC}$-TBox* is a finite set of *concept inclusions (CIs)* $C \sqsubseteq D$, where C, D are $\mathcal{ALC}$-concepts, and likewise for $\mathcal{ALCI}$-TBoxes. An *$\mathcal{ALCFI}$-TBox* is an $\mathcal{ALCI}$-TBox that additionally admits functionality assertions $\mathsf{func}(r)$, where $r \in \mathsf{N_R} \cup \mathsf{N_R^-}$.

An *ABox* $\mathcal{A}$ is a finite set of assertions of the form $A(a)$ and $r(a, b)$ with $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and $a, b \in \mathsf{N_I}$. In some cases, we drop the finiteness condition on ABoxes and then explicitly speak about *infinite ABoxes*. 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 $r(b, a) \in \mathcal{A}$.

The semantics of DLs is given by *interpretations* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$, where $\Delta^\mathcal{I}$ is a non-empty set and $\cdot^\mathcal{I}$ maps each concept name $A \in \mathsf{N_C}$ to a subset $A^\mathcal{I}$ of $\Delta^\mathcal{I}$, each role name $r \in \mathsf{N_R}$ to a binary relation $r^\mathcal{I}$ on $\Delta^\mathcal{I}$, and each individual name a to an element $a^\mathcal{I} \in \Delta^\mathcal{I}$. We make the *unique name assumption*, i.e., $a^\mathcal{I} \neq b^\mathcal{I}$ whenever $a \neq b$. For $\mathcal{ALCI}$ the results of this paper do not depend on this assumption, but for $\mathcal{ALCFI}$ dependency on the unique name assumption is left open. The extension $C^\mathcal{I} \subseteq \Delta^\mathcal{I}$ of a concept C under the interpretation $\mathcal{I}$ is defined as usual, see (Baader et al. 2003). For the purposes of this paper, it is often convenient to work with interpretations that interpret only *some* individual names, but not all. In this case, we use $\mathsf{Ind}(\mathcal{I})$ to denote the set of individual names interpreted by $\mathcal{I}$.

We say that $\mathcal{I}$ *satisfies* a CI $C \sqsubseteq D$ if $C^\mathcal{I} \subseteq D^\mathcal{I}$, an assertion $A(a)$ if $a \in \mathsf{Ind}(\mathcal{I})$ and $a^\mathcal{I} \in C^\mathcal{I}$, an assertion $r(a, b)$ if $a, b \in \mathsf{Ind}(\mathcal{I})$ and $(a^\mathcal{I}, b^\mathcal{I}) \in r^\mathcal{I}$, and a functionality assertion $\mathsf{func}(r)$ if $r^\mathcal{I}$ is a function. Finally, $\mathcal{I}$ is a *model* of a TBox $\mathcal{T}$ (ABox $\mathcal{A}$) if it satisfies all inclusions in $\mathcal{T}$ (all assertions in $\mathcal{A}$). The class of all models of $\mathcal{T}$ and $\mathcal{A}$ is denoted by $\mathsf{Mod}(\mathcal{T}, \mathcal{A})$. We call an ABox $\mathcal{A}$ *consistent w.r.t. a TBox* $\mathcal{T}$ if $\mathsf{Mod}(\mathcal{T}, \mathcal{A}) \neq \emptyset$.

Throughout this paper, we consider several query languages which can all be seen as fragments of *positive existential queries (PEQs)*. A PEQ $q(\vec{x})$ is a first-order formula with free variables $\vec{x}$ constructed from atoms $A(t)$, $r(t, t')$, and $t = t'$, (where $A \in \mathsf{N_C}$, $r \in \mathsf{N_R}$, and t, t' range over individual names and variables) using conjunction, disjunction, and existential quantification. The variables in $\vec{x}$ are the *answer variables* of q. A PEQ without answer variables is *Boolean*. We say that a tuple $\vec{a} \subseteq \mathsf{Ind}(\mathcal{A})$ of the same arity as $\vec{x}$ is an *answer to $q(\vec{x})$ in an interpretation* $\mathcal{I}$ if $\mathcal{I} \models q[\vec{a}]$, where $q[\vec{a}]$ results from replacing the answer variables $\vec{x}$ in $q(\vec{x})$ with $\vec{a}$. Moreover, $\vec{a}$ is a *certain answer to $q(\vec{x})$ in $\mathcal{A}$ w.r.t.* $\mathcal{T}$, in symbols $(\mathcal{T}, \mathcal{A}) \models q(\vec{a})$, if $\mathcal{I} \models q[\vec{a}]$ for all $\mathcal{I} \in \mathsf{Mod}(\mathcal{T}, \mathcal{A})$. The set of all certain answers is denoted with $\mathsf{cert}_\mathcal{T}(q, \mathcal{A}) = \{\vec{a} \mid (\mathcal{T}, \mathcal{A}) \models q(\vec{a})\}$.

For Boolean queries q, we write $(\mathcal{T}, \mathcal{A}) \models q$ instead of $\mathsf{cert}_\mathcal{T}(q, \mathcal{A}) = \{()\}$ with $()$ the empty tuple; we then speak of deciding $(\mathcal{T}, \mathcal{A}) \models q$ rather than of computing $\mathsf{cert}_\mathcal{T}(q, \mathcal{A})$.

Example 1. (1) Let $\mathcal{T}_r = \{\exists r.A \sqsubseteq A\}$ and $q_0(x) = A(x)$. For any ABox $\mathcal{A}$, $\mathsf{cert}_{\mathcal{T}_r}(q_0, \mathcal{A})$ is the set of all $a \in \mathsf{Ind}(\mathcal{A})$ such that there is an r-path in $\mathcal{A}$ from a to some b with $A(b) \in \mathcal{A}$; i.e., there are $r(a_0, a_1), \ldots, r(a_{n-1}, a_n) \in \mathcal{A}$, $n \geq 0$, with $a_0 = a$, $a_n = b$, and $A(b) \in \mathcal{A}$.

(2) Consider an undirected graph represented as an ABox $\mathcal{A}$ with assertions $r(a, b), r(b, a) \in \mathcal{A}$ iff there is an edge between a and b. Let $A_1, \ldots, A_k, M$ be fresh concept names. Then $\mathcal{A}$ is k-colorable iff $(\mathcal{T}_k, \mathcal{A}) \not\models \exists x.M(x)$, where

$$\begin{array}{rcl} \mathcal{T}_k & = & \{A_i \sqcap A_j \sqsubseteq M \mid 1 \leq i < j \leq k\} \cup \\ & & \{A_i \sqcap \exists r.A_i \sqsubseteq M \mid 1 \leq i \leq k\} \cup \\ & & \{\top \sqsubseteq \bigsqcup_{1 \leq i \leq k} A_i\}. \end{array}$$

⊣

As additional query languages, we consider *conjunctive queries (CQs)*, which are PEQs without disjunction, as well as the following two weaker languages that are frequently used in an OBDA context.

Recall that *$\mathcal{EL}$-concepts* are constructed from $\mathsf{N_C}$ and $\mathsf{N_R}$ using conjunction, existential restriction, and the $\top$-concept (Baader, Brandt, and Lutz 2005). *$\mathcal{ELI}$-concepts* additionally admit inverse roles. If C is an $\mathcal{ELI}$-concept and $a \in \mathsf{N_I}$, then $C(a)$ is called an *$\mathcal{ELI}$-query (ELIQ)*; if C is an $\mathcal{EL}$-concept, then $C(a)$ is called an *$\mathcal{EL}$-query (ELQ)*. Note that every ELIQ (and, therefore, every ELQ) can be regarded as an acyclic Boolean CQ. For example, the ELIQ $\exists r.(A \sqcap \exists s^-.B)(a)$ is equivalent to the Boolean CQ

$$\exists x \exists y.(r(a, x) \wedge A(x) \wedge s(y, x) \wedge B(y)).$$

In what follows, we will not distinguish between an ELIQ and its translation into a Boolean CQ and freely apply notions introduced for PEQs also to ELIQs and ELQs.

For an ABox $\mathcal{A}$, we denote by $\mathcal{I}_\mathcal{A}$ the interpretation with $\Delta^{\mathcal{I}_\mathcal{A}} = \mathsf{Ind}(\mathcal{A})$, $a^{\mathcal{I}_\mathcal{A}} = a$ for all $a \in \mathsf{Ind}(\mathcal{A})$, and

$$\begin{aligned} A^\mathcal{I} &= \{a \mid A(a) \in \mathcal{A}\} \\ r^\mathcal{I} &= \{(a,b) \mid r(a,b) \in \mathcal{A}\} \end{aligned}$$

for any $A \in \mathsf{N_C}$ and $r \in \mathsf{N_R}$. Note that $\mathsf{Ind}(\mathcal{I}) = \mathsf{Ind}(\mathcal{A})$.

In what follows, we sometimes slightly abuse notation and use PEQ to denote the set of all first-order queries, and likewise for CQ, ELIQ, and ELQ. We now introduce the main notions investigated in this paper.

Definition 2 (Complexity). Let $\mathcal{T}$ be an $\mathcal{ALCFI}$-TBox and let $\mathcal{Q} \in \{\text{CQ}, \text{PEQ}, \text{ELIQ}, \text{ELQ}\}$. Then

- *$\mathcal{Q}$-answering w.r.t. $\mathcal{T}$ is in* PTIME if for every $q(\vec{x}) \in \mathcal{Q}$, there is a polytime algorithm that computes, given an ABox $\mathcal{A}$, the set $\mathsf{cert}_\mathcal{T}(q, \mathcal{A})$;
- *$\mathcal{Q}$-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 $(\mathcal{T}, \mathcal{A}) \models q$.

Note that $\mathcal{Q}$-answering w.r.t. $\mathcal{T}$ is in PTIME iff for every *Boolean* query $q \in \mathcal{Q}$, there is a polytime algorithm deciding, given an ABox $\mathcal{A}$, whether $(\mathcal{T}, \mathcal{A}) \models q$. We give some examples that illustrate the above notions.

Example 3. (1) CQ-answering w.r.t. $\mathcal{T}_r$ from Example 1 is in PTIME since for any ABox $\mathcal{A}$, $\mathsf{cert}_{\mathcal{T}_r}(q, \mathcal{A})$ can be computed as follows. Let $\mathcal{A}'$ by the ABox obtained from $\mathcal{A}$ by adding $A(a)$ to $\mathcal{A}$ if there is an r-path from a to some b with $A(b) \in \mathcal{A}$. Then $\mathsf{cert}_{\mathcal{T}_r}(q, \mathcal{A}) = \{\vec{a} \mid \mathcal{I}_{\mathcal{A}'} \models q(\vec{a})\}$ can be computed in PTIME (actually in AC^0) by evaluating the PEQ q in the structure $\mathcal{I}_{\mathcal{A}'}$.

(2) Consider the TBoxes $\mathcal{T}_k$ from Example 1 that express k-colorability . For $k \geq 3$, CQ-answering w.r.t. $\mathcal{T}_k$ is CONP-hard since k-colorability is NP-hard. However, in contrast to the tractability of 2-colorability, CQ-answering w.r.t. $\mathcal{T}_2$ is CONP-hard as well. This follows from Theorem 11 below and, intuitively, is the case because $\mathcal{T}_2$ 'entails a disjunction': for $\mathcal{A} = \{B(a)\}$, we have $(\mathcal{T}_2, \mathcal{A}) \models A_1(a) \vee A_2(a)$, but neither $(\mathcal{T}_2, \mathcal{A}) \models A_1(a)$ nor $(\mathcal{T}_2, \mathcal{A}) \models A_2(a)$. ⊣

Interestingly, PTIME upper bounds (and the observations in Example 3) do not depend on whether we consider PEQs, CQs, or ELIQs.

Theorem 4. *For all $\mathcal{ALCFI}$-TBoxes $\mathcal{T}$,*

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. *ELIQ-answering w.r.t. $\mathcal{T}$ is in* PTIME *iff ELQ-answering w.r.t. $\mathcal{T}$ is in* PTIME*, provided that $\mathcal{T}$ is an $\mathcal{ALCF}$-ABox.*

The proof is based on Theorems 9 and 11 below. Theorem 4 gives a uniform explanation for the fact that, in the traditional logic-centered approach to data complexity in OBDA, the complexity of answering PEQs, CQs, and ELIQs has turned out to be identical for many DLs. It allows us to (sometimes) speak of the 'complexity of query answering' without reference to a concrete query language.

3 Materializability

We introduce materializability of a TBox $\mathcal{T}$ as a central tool for analyzing the complexity of query answering. Our main result is that *non*-materializability of a TBox is a sufficient condition for query answering being CONP-hard.

Definition 5 (Materializable). Let $\mathcal{T}$ be an $\mathcal{ALCFI}$-TBox and $\mathcal{Q} \in \{\text{CQ}, \text{PEQ}, \text{ELIQ}, \text{ELQ}\}$. Then,

- a model $\mathcal{I}$ of $\mathcal{T}$ and an ABox $\mathcal{A}$ is a *$\mathcal{Q}$-materialization of $\mathcal{T}$ and $\mathcal{A}$* if for all queries $q(\vec{x}) \in \mathcal{Q}$ and potential answers $\vec{a} \subseteq \mathsf{Ind}(\mathcal{A})$, we have $\mathcal{I} \models q[\vec{a}]$ iff $(\mathcal{T}, \mathcal{A}) \models q(\vec{a})$;
- $\mathcal{T}$ is *$\mathcal{Q}$-materializable* if for every ABox $\mathcal{A}$ that is consistent w.r.t. $\mathcal{T}$, there is a $\mathcal{Q}$-materialization of $\mathcal{T}$ and $\mathcal{A}$.

It can be proved that, in Example 3 (1), the interpretation $\mathcal{I}_{\mathcal{A}'}$ is a PEQ-materialization of $\mathcal{T}_r$ and $\mathcal{A}$. Note that a $\mathcal{Q}$-materialization can be viewed as a more abstract version of a *canonical model* as often used in the context of 'Horn DLs' such as $\mathcal{EL}$ and DL-Lite (Lutz, Toman, and Wolter 2009; Kontchakov et al. 2010). In fact, the ELQ-materialization in the next example is exactly the 'compact canonical model' from (Lutz, Toman, and Wolter 2009).

Example 6. Let $\mathcal{T} = \{A \sqsubseteq \exists r.A\}$ and $\mathcal{A}$ be an ABox with at least one assertion of the form $A(a)$. To obtain an ELQ-materialization $\mathcal{M}_\mathcal{A}$ of $\mathcal{T}$ and $\mathcal{A}$, start with the interpretation $\mathcal{I}_\mathcal{A}$, add a fresh domain element d_r, and set

$$\begin{aligned} A^{\mathcal{M}_\mathcal{A}} &= A^{\mathcal{I}_\mathcal{A}} \cup \{d_r\} \\ r^{\mathcal{M}_\mathcal{A}} &= r^{\mathcal{I}_\mathcal{A}} \cup \{(a, d_r) \mid A(a) \in \mathcal{A}\} \cup \{(d_r, d_r)\}. \quad \dashv \end{aligned}$$

Trivially, a PEQ-materialization is a CQ-materialization is an ELIQ-materialization is an ELQ-materialization. We show below as part of Lemma 8 that the converse holds for the CQ-PEQ case. However, the following example demonstrates that ELQ-materializations are different from ELIQ-materialization. A similar argument separates ELIQ-materializations from CQ-materializations.

Example 7. Let $\mathcal{T}$ be as in Example 6,

$$\begin{aligned} \mathcal{A} &= \{B_1(a), B_2(b), A(a), A(b)\} \text{ and} \\ q &= (B_1 \sqcap \exists r.\exists r^-.B_2)(a), \end{aligned}$$

Then the ELQ-materialization $\mathcal{M}_\mathcal{A}$ from Example 6 is not a $\mathcal{Q}$-materialization for any $\mathcal{Q} \in \{\text{ELIQ}, \text{CQ}, \text{PEQ}\}$. For example, we have $\mathcal{M}_\mathcal{A} \models q$, but $(\mathcal{T}, \mathcal{A}) \not\models q$. An ELIQ/CQ/PEQ-materialization of $\mathcal{T}$ and $\mathcal{A}$ is obtained by unfolding $\mathcal{M}_\mathcal{A}$: instead of using only one additional individual d_r as a witness for $\exists r.A$, we attach to both a ad b an infinite r-path of elements that satisfy A. Note that every CQ/PEQ-materialization of $\mathcal{T}_r$ and $\mathcal{A}$ must be infinite. ⊣

Before linking materializations to the complexity of query answering, we characterize them semantically in terms of simulations and homomorphisms. This is interesting in its own right and establishes a close connection between materialization and initial models as studied in model theory, algebraic specification, and logic programming (Malcev 1971; Meseguer and Goguen 1985; Makowsky 1987). It also allows us to show that, despite the discrepancies between materiali*zations* for different query languages pointed out above, materializa*bility* coincides for PEQs, CQs, and ELIQs (and ELQs when the TBox is formulated in $\mathcal{ALCF}$).

Analyzing Materializability

A *simulation from an interpretation* $\mathcal{I}_1$ *to an interpretation* $\mathcal{I}_2$ is a relation $S \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$ such that

1. for all $A \in \mathsf{N_C}$: if $d_1 \in A^{\mathcal{I}_1}$ and $(d_1, d_2) \in S$, then $d_2 \in A^{\mathcal{I}_2}$;
2. for all $r \in \mathsf{N_R}$: if $(d_1, d_2) \in S$ and $(d_1, d_1') \in r^{\mathcal{I}_1}$, then there exists $d_2' \in \Delta^{\mathcal{I}_2}$ such that $(d_1', d_2') \in S$ and $(d_2, d_2') \in r^{\mathcal{I}_2}$;
3. for all $a \in \mathsf{Ind}(\mathcal{I}_1)$: $a \in \mathsf{Ind}(\mathcal{I}_2)$ and $(a^{\mathcal{I}_1}, a^{\mathcal{I}_2}) \in S$.

We call S an *i-simulation* if Condition 2 is satisfied also for *inverse* roles and a *homomorphism* if S is a function. An interpretation $\mathcal{I}$ is called *hom-initial in a class* $\mathbb{K}$ *of interpretations* if for every $\mathcal{J} \in \mathbb{K}$, there exists a homomorphism from $\mathcal{I}$ to $\mathcal{J}$. $\mathcal{I}$ is called *sim-initial (i-sim-initial) in a class* $\mathbb{K}$ *of interpretations* if for every $\mathcal{J} \in \mathbb{K}$, there exists a simulation (i-simulation) from $\mathcal{I}$ to $\mathcal{J}$.

An interpretation $\mathcal{I}$ is *generated* if every $d \in \Delta^{\mathcal{I}}$ is reachable from some $a^{\mathcal{I}}$, $a \in \mathsf{Ind}(\mathcal{I})$, in the undirected graph $(\Delta^{\mathcal{I}}, \{\{d, d'\} \mid (d, d') \in \bigcup_{r \in \mathsf{N_R}} r^{\mathcal{I}}\})$. The next result relates simulations and homomorphisms to materializations.

Lemma 8. *Let $\mathcal{T}$ be an $\mathcal{ALCFI}$-TBox, $\mathcal{A}$ an ABox, and $\mathcal{I} \in \mathsf{Mod}(\mathcal{T}, \mathcal{A})$. Then $\mathcal{I}$ is*

1. *an ELIQ-materialization of $\mathcal{T}$ and $\mathcal{A}$ iff it is i-sim-initial in* $\mathsf{Mod}(\mathcal{T}, \mathcal{A})$*;*
2. *a CQ-materialization of $\mathcal{T}$ and $\mathcal{A}$ iff it is a PEQ-materialization of $\mathcal{T}$ and $\mathcal{A}$ iff it is hom-initial in* $\mathsf{Mod}(\mathcal{T}, \mathcal{A})$*, provided that $\mathcal{I}$ is countable and generated;*
3. *an ELQ-materialization of $\mathcal{T}$ and $\mathcal{A}$ iff it is sim-initial in* $\mathsf{Mod}(\mathcal{T}, \mathcal{A})$*, provided that $\mathcal{T}$ is an $\mathcal{ALCF}$-TBox.*

Proof. (Sketch) The proofs of "$\Leftarrow$" are straightforward since matches of PEQs, CQs, and ELIQs are preserved under i-simulations and homomorphisms, and matches of ELQs are preserved under simulations. We thus concentrate on "$\Rightarrow$".

(1) Assume $\mathcal{I}$ is an ELIQ-materialization and let $\mathcal{J} \in \mathsf{Mod}(\mathcal{T}, \mathcal{A})$. If $\mathcal{J}$ has finite outdegree, an i-simulation from $\mathcal{I}$ to $\mathcal{J}$ can be constructed in the same way as in standard proofs showing that simulations characterize the expressive power of $\mathcal{EL}$-concepts (Lutz, Piro, and Wolter 2011). If $\mathcal{J}$ has infinite outdegree, then one can construct a selective unfolding $\mathcal{J}^* \in \mathsf{Mod}(\mathcal{T}, \mathcal{A})$ of $\mathcal{J}$ whose outdegree is finite and such that there is a homomorphism from $\mathcal{J}^*$ to $\mathcal{J}$. It remains to compose an i-simulation from $\mathcal{I}$ to $\mathcal{J}^*$ with the homomorphism from $\mathcal{J}^*$ to $\mathcal{J}$.

For (2), we show that any countable and generated CQ-materialization is hom-initial. If $\mathcal{I}$ is such a CQ-materialization and $\mathcal{J} \in \mathsf{Mod}(\mathcal{T}, \mathcal{A})$, then by the semantics of CQs we can find a homomorphism from any *finite subinterpretation* of $\mathcal{I}$ to $\mathcal{J}$. If $\mathcal{J}$ is of finite outdegree, we can assemble all those homomorphisms into a homomorphism from $\mathcal{I}$ to $\mathcal{J}$ in a direct way (using that $\mathcal{I}$ is countable and generated). For $\mathcal{J}$ of non-finite outdegree, we compose the homorphism from $\mathcal{I}$ to $\mathcal{J}^*$ with the homomorphism from $\mathcal{J}^*$ to $\mathcal{J}$, with $\mathcal{J}^*$ constructed as in (1). The claim for $\mathcal{ALCF}$-TBoxes is proved similarly to (1). ❑

In Point 2 of Lemma 8, we cannot drop the condition that $\mathcal{I}$ is generated without losing correctness, please see the long version for details. It is open whether the same is true for countability.

We now show that materializability coincides for the query languages studied in this paper.

Theorem 9. *Let $\mathcal{T}$ be an $\mathcal{ALCFI}$-TBox. Then*

1. *$\mathcal{T}$ is PEQ-materializable iff $\mathcal{T}$ is CQ-materializable iff $\mathcal{T}$ is ELIQ-materializable;*
2. *the above is the case iff* $\mathsf{Mod}(\mathcal{T}, \mathcal{A})$ *contains a hom-initial $\mathcal{I}$ for every ABox $\mathcal{A}$ iff* $\mathsf{Mod}(\mathcal{T}, \mathcal{A})$ *contains an i-sim-initial $\mathcal{I}$ for every ABox $\mathcal{A}$;*
3. *the above is the case iff $\mathcal{T}$ is ELQ-materializable iff* $\mathsf{Mod}(\mathcal{T}, \mathcal{A})$ *contains a sim-initial $\mathcal{I}$ for every ABox $\mathcal{A}$, provided that $\mathcal{T}$ is an $\mathcal{ALCF}$-TBox.*

This theorem is essentially a consequence of Lemma 8. The proof of "$\Leftarrow$" in Point 2 employs a selective unfolding technique (similar to the one used in the the proof of Lemma 8) to transform an i-simulation into a homomorphism. Due to this technique, the conditions of generatedness and countability from Point 2 of Lemma 8 can be avoided in Theorem 9.

Because of Theorem 9, we sometimes speak of *materializability* without reference to a query language and of *materializations* instead of PEQ-materializations. Interestingly, materializability turns out to (also) be equivalent to the disjunction property, which is sometimes also called convexity and plays a central role in attaining PTIME complexity for standard reasoning in DLs (Baader, Brandt, and Lutz 2005). This observation will be useful for the proof of our main theorem below.

A TBox $\mathcal{T}$ has the *ABox disjunction property* if for all ABoxes $\mathcal{A}$ and ELIQs $C_1(a_1), \dots, C_n(a_n)$, it follows from $(\mathcal{T}, \mathcal{A}) \models C_1(a_1) \vee \dots \vee C_n(a_n)$ that $(\mathcal{T}, \mathcal{A}) \models C_i(a_i)$ for some $i \leq n$.

Theorem 10. *An $\mathcal{ALCFI}$-TBox $\mathcal{T}$ is materializable iff it has the disjunction property.*

Proof. For the nontrivial "$\Leftarrow$" direction, let $\mathcal{A}$ be an ABox that is consistent w.r.t. $\mathcal{T}$ and such that there is no ELIQ-materialization of $\mathcal{T}$ and $\mathcal{A}$. Then $\mathcal{T} \cup \mathcal{A} \cup \Gamma$ is not satisfiable, where

$$\Gamma = \{\neg C(a) \mid (\mathcal{T}, \mathcal{A}) \not\models C(a), a \in \mathsf{Ind}(\mathcal{A}), C(a) \text{ ELIQ}\}.$$

In fact, any satisfying interpretation would be an ELIQ-materialization. By compactness, there is a finite subset Γ' of Γ such that $\mathcal{T} \cup \mathcal{A} \cup \Gamma'$ is not satisfiable, i.e. $(\mathcal{T}, \mathcal{A}) \models \bigvee_{\neg C(a) \in \Gamma'} C(a)$. By definition of Γ', $(\mathcal{T}, \mathcal{A}) \not\models C(a)$, for all $\neg C(a) \in \Gamma'$. Thus, $\mathcal{T}$ lacks the ABox disjunction property. ❑

Materializability and CONP-hardness

Based on Theorems 9 and 10, we now establish the main result on materializability.

Theorem 11. *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 exploits failure of the ABox disjunction property to generalize the reduction of 2+2-SAT used in (Schaerf 1993).

The converse of Theorem 11 fails, i.e., there are TBoxes that are materializable, but for which ELIQ-answering is coNP-hard. In fact, materializations of such a TBox $\mathcal{T}$ and ABox $\mathcal{A}$ are guaranteed to exist, but cannot always be computed in PTIME (unless PTIME = coNP). Technically, this follows from Theorem 20 later on which states that for every non-uniform CSP, there is a materializable $\mathcal{ALC}$-TBox for which Boolean CQ-answering has the same complexity, up to complementation of the complexity class.

Theorem 11 also allows us to prove Theorem 4 (for this purpose, it is crucial for Theorem 11 to refer to ELIQs and ELQs rather than CQs or PEQs).

Proof of Theorem 4 (sketch). By Theorem 11, it is sufficient to consider materializable TBoxes when proving Theorem 4. To show, for example, that if CQ-answering w.r.t. $\mathcal{T}$ is in PTIME then PEQ-answering w.r.t. $\mathcal{T}$ is in PTIME, one can first transform a PEQ $q(\vec{x})$ into an equivalent union of CQs $\bigsqcup_{i\in I} q_i(\vec{x})$. CQ-materializability of $\mathcal{T}$ implies that, for any ABox $\mathcal{A}$, we have $\mathsf{cert}_{\mathcal{T}}(q,\mathcal{A}) = \bigcup_{i\in I} \mathsf{cert}_{\mathcal{T}}(q_i,\mathcal{A})$. It thus remains to note that each set $\mathsf{cert}_{\mathcal{T}}(q_i,\mathcal{A})$ can be computed in PTIME. The remaining reductions are more involved, but based on similar ideas. □

4 Unraveling Tolerance

We develop a condition on TBoxes, called unraveling tolerance, that is sufficient for PTIME query answering and strictly generalizes syntactic 'Horn conditions' such as the ones used to define the DL Horn-$\mathcal{SHIQ}$, which was designed as a maximal DL with PTIME query answering (Hustadt, Motik, and Sattler 2007; Eiter et al. 2008). 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. More precisely, the *unraveling* $\mathcal{A}^u$ of an ABox $\mathcal{A}$ is the following (possibly infinite) ABox:

- $\mathsf{Ind}(\mathcal{A}^u)$ is the set of sequences $b_0 r_0 b_1 \cdots r_{n-1} b_n$, $n \geq 0$, with $b_0,\ldots,b_n \in \mathsf{Ind}(\mathcal{A})$ and $r_0,\ldots,r_{n-1} \in \mathsf{N_R} \cup \mathsf{N_R^-}$ such that for all $i < n$, we have $r_i(b_i,b_{i+1}) \in \mathcal{A}$ and $(b_{i-1}, r_{i-1}^-) \neq (b_{i+1}, r_i)$ when $i > 0$;
- for each $C(b) \in \mathcal{A}$ and $\alpha = b_0 \cdots b_n \in \mathsf{Ind}(\mathcal{A}^u)$ with $b_n = b$, we have $C(\alpha) \in \mathcal{A}^u$;
- for each $\alpha = b_0 r_0 \cdots r_{n-1} b_n \in \mathsf{Ind}(\mathcal{A}^u)$ with $n > 0$, we have $r_{n-1}(b_0 \cdots b_{n-1}, \alpha) \in \mathcal{A}^u$.

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

Definition 12 (Unraveling Tolerance). 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. Note that it is pointless to define unraveling tolerance for queries that are not necessarily tree shaped, such as CQs.

Example 13. (1) The $\mathcal{ALC}$-TBox $\mathcal{T}_1 = \{A \sqsubseteq \forall r.B\}$ is unraveling tolerant. This can be proved by showing that (i) for any (finite or infinite) ABox $\mathcal{A}$, the interpretation $\mathcal{I}_{\mathcal{A}}^+$ that is obtained from $\mathcal{I}_{\mathcal{A}}$ by setting $B^{\mathcal{I}_{\mathcal{A}}^+} = B^{\mathcal{I}_{\mathcal{A}}} \cup (\exists r^-.A)^{\mathcal{I}_{\mathcal{A}}}$ is an ELIQ-materialization of $\mathcal{T}_1$ and $\mathcal{A}$; and (ii) $\mathcal{I}_{\mathcal{A}}^+ \models C(a)$ iff $\mathcal{I}_{\mathcal{A}^u}^+ \models C(a)$ for all ELIQs $C(a)$. The proof of (ii) is based on a simple induction on the structure of the $\mathcal{ELI}$-concept C. As witnessed by the ABox $\mathcal{A} = \{r(a,b), A(a)\}$ and ELIQ $B(b)$, the use of inverse roles in the definition of $\mathcal{A}^u$ is crucial here despite the fact that $\mathcal{T}_1$ does not use inverse roles.

(2) A simple example for an $\mathcal{ALC}$-TBox that is not unraveling tolerant is $\mathcal{T}_2 = \{A \sqcap \exists r.A \sqsubseteq B, \neg A \sqcap \exists r.\neg A \sqsubseteq B\}$. For $\mathcal{A} = \{r(a,a)\}$, it is easy to see that we have $(\mathcal{T}_2,\mathcal{A}) \models B(a)$ (use a case distinction on the truth value of A at a!), but $(\mathcal{T}_2,\mathcal{A}^u) \not\models B(a)$. ⊣

Before we show that unraveling tolerance indeed implies PTIME query answering, we first demonstrate the generality of this property by relating it to Horn-$\mathcal{ALCFI}$, the $\mathcal{ALCFI}$-fragment of Horn-$\mathcal{SHIQ}$. Different versions of Horn-$\mathcal{SHIQ}$ have been proposed in the literature, giving rise to different versions of Horn-$\mathcal{ALCFI}$ (Hustadt, Motik, and Sattler 2007; Krötzsch, Rudolph, and Hitzler 2007; Eiter et al. 2008; Kazakov 2009). As the original definition from (Hustadt, Motik, and Sattler 2007) based on polarity is rather technical, we prefer to work with the following, more direct syntax. A *Horn-$\mathcal{ALCFI}$-TBox* has the form $\mathcal{T} = \{\top \sqsubseteq C_{\mathcal{T}}\} \cup \mathcal{F}$, where $\mathcal{F}$ is a set of functionality assertions and $C_{\mathcal{T}}$ is built according to the topmost rule in

$$R, R' ::= \top \mid \bot \mid A \mid \neg A \mid R \sqcap R' \mid L \rightarrow R \mid \exists r.R \mid \forall r.R$$
$$L, L' ::= \top \mid \bot \mid A \mid L \sqcap L' \mid L \sqcup L' \mid \exists r.L$$

where r ranges over $\mathsf{N_R} \cup \mathsf{N_R^-}$ and $L \rightarrow R := \neg L \sqcup R$. By applying some simple transformations, it is not hard to show that every Horn-$\mathcal{ALCFI}$-TBox according to the original, polarity-based definition is equivalent to a Horn-$\mathcal{ALCFI}$-TBox of the form introduced here. Although not important in our context, we note that even a polytime transformation is possible.

Theorem 14.
Every Horn-$\mathcal{ALCFI}$-TBox is unraveling tolerant.

Proof. (hint) Based on a generalization of the argument in Example 13 (1), where the ad hoc materialization $\mathcal{I}_{\mathcal{A}}^+$ is replaced with a systematically constructed *canonical model* of $\mathcal{T}$ and $\mathcal{A}$. ❏

Theorem 14 shows that unraveling tolerance and Horn logic are closely related. Yet, the next example shows that there are unraveling tolerant $\mathcal{ALCFI}$-TBoxes that are not equivalent to any Horn sentence of FO. Since any Horn-$\mathcal{ALCFI}$-TBox is equivalent to such a sentence, it follows that unraveling tolerant $\mathcal{ALCFI}$-TBoxes strictly generalize Horn-$\mathcal{ALCFI}$-TBoxes. This increased generality will pay off in

Section 5 when we establish a dichotomy result for TBoxes of depth one.

Example 15. Take the $\mathcal{ALC}$-TBox

$$\mathcal{T} = \{\exists r.(A \sqcap \neg B_1 \sqcap \neg B_2) \sqsubseteq \exists r.(\neg A \sqcap \neg B_1 \sqcap \neg B_2)\}.$$

One can show as in Example 13 (1) that $\mathcal{T}$ is unraveling tolerant; here, the materialization is actually $\mathcal{I}_\mathcal{A}$ itself instead of some $\mathcal{I}^+_\mathcal{A}$, i.e., as far as ELIQ (and even PEQ) answering is concerned, $\mathcal{T}$ cannot be distinguished from the empty TBox.

It is well-known that FO Horn sentences are preserved under direct products of interpretations (Chang and Keisler 1990). To show that $\mathcal{T}$ is not equivalent to any such sentence, it thus suffices to show that $\mathcal{T}$ is not preserved under direct products. This is simple: let $\mathcal{I}_1$ and $\mathcal{I}_2$ consist of a single r-edge between elements d and e, and let $e \in (A \sqcap B_1 \sqcap \neg B_2)^{\mathcal{I}_1}$ and $e \in (A \sqcap \neg B_1 \sqcap B_2)^{\mathcal{I}_2}$; then the direct product $\mathcal{I}$ of $\mathcal{I}_1$ and $\mathcal{I}_2$ still has the r-edge between (d,d) and (e,e) and satisfies $(e,e) \in (A \sqcap \neg B_1 \sqcap \neg B_2)^\mathcal{I}$, thus is not a model of $\mathcal{T}$. ⊣

We now establish the PTIME upper bound for unraveling tolerant TBoxes.

Theorem 16. *If an $\mathcal{ALCFI}$-TBox $\mathcal{T}$ is unraveling tolerant, then PEQ-answering w.r.t. $\mathcal{T}$ is in* PTIME.

Proof.(sketch) Let $\mathcal{T}$ be unraveling tolerant. By Theorem 4, it suffices to show that ELIQ-answering w.r.t. $\mathcal{T}$ is in PTIME. Let $\mathcal{A}$ be an ABox and $q = C_0(a_0)$ an ELIQ. Let $\mathsf{cl}(\mathcal{T}, C_0)$ denote the closure under single negation of the set of subconcepts of $\mathcal{T}$ and C_0. $\mathsf{tp}(\mathcal{T}, C_0)$ denotes the set of all *types* (aka Hintikka sets or maximal consistent sets) over $\mathsf{cl}(\mathcal{T}, C_0)$. A *type assignment* is a map $\mathsf{Ind}(\mathcal{A}) \to 2^{\mathsf{tp}(\mathcal{T},q)}$.

The PTIME algorithm for checking whether $(\mathcal{T}, \mathcal{A}) \models q$ is based on the computation of a sequence of type assignments $\pi_0, \pi_1, \ldots$ as follows. For every $a \in \mathsf{Ind}(\mathcal{A})$, $\pi_0(a)$ is the set of types $t \in \mathsf{tp}(\mathcal{T}, q)$ such that $A(a) \in \mathcal{A}$ implies $A \in t$. Then, $\pi_{i+1}(a)$ is defined as the set of types $t_a \in \pi_i(a)$ such that for all $r(a,b) \in \mathcal{A}$, r a role name or the inverse thereof, there is a type $t_b \in \pi_i(b)$ such that $t_a \rightsquigarrow_r t_b$, where we write $t_a \rightsquigarrow_r t_b$ if the following conditions are satisfied: if $C \in t_b$ then $\exists r.C \in t_a$, for all $\exists r.C \in \mathsf{cl}(\mathcal{T}, C_0)$; if $C \in t_a$ then $\exists r^-.C \in t_b$, for all $\exists r^-.C \in \mathsf{cl}(\mathcal{T}, C_0)$; $\exists r.C \in t_a$ iff $C \in t_b$, for all $\exists r.C \in \mathsf{cl}(\mathcal{T}, C_0)$ with $\mathsf{func}(r) \in \mathcal{T}$; $\exists r^-.C \in t_b$ iff $C \in t_a$, for all $\exists r^-.C \in \mathsf{cl}(\mathcal{T}, C_0)$ with $\mathsf{func}(r^-) \in \mathcal{T}$.

Clearly, the sequence $\pi_0, \pi_1, \ldots$ stabilizes after at most $\mathcal{O}(|\mathcal{A}|)$ steps and can be computed in time polynomial in $|\mathcal{A}|$ (since the cardinality of $\mathsf{tp}(\mathcal{T}, q)$ is bounded by a constant). Let π be the final type assignment in the sequence. In the long version, we show that $(\mathcal{T}, \mathcal{A}) \models q$ iff $C_0 \in t$ for all $t \in \pi(a_0)$. ❏

By Theorems 4 and 14 and since we actually exhibit a *uniform* algorithm for ELIQ-answering w.r.t. unraveling tolerant TBoxes, Theorem 16 also reproves the known PTIME upper bound for CQ-answering in Horn-$\mathcal{ALCFI}$ (Eiter et al. 2008).

By Theorems 11 and 16, unraveling tolerance implies materializability unless PTIME = NP. Based on the disjunction property, this implication can also be proved without the side condition.

Lemma 17. *Every unraveling tolerant $\mathcal{ALCFI}$-TBox is materializable.*

The converse of Lemma 17 and, more generally, of Theorem 16 fails (unless PTIME = NP). In fact, while unraveling tolerance is a sufficient condition for PTIME query answering, it is not a necessary one. An example is given in Section 6, where it is shown that the TBox $\mathcal{T}_2$ from Example 1 that represents 2-colorability has PTIME query answering, but is not unraveling tolerant.

The PTIME algorithm in Theorem 16 resembles the standard *arc consistency* algorithm for CSPs (Dechter 2003). This link to CSPs can be formalized for $\mathcal{ALCI}$-TBoxes using the templates $\mathcal{I}_{\mathcal{T},q}$ constructed in the proof of Theorems 22 and 24 below: it is known that a CSP can be solved using arc consistency iff it has tree obstructions (Krokhin 2010). Also, one can show that an $\mathcal{ALCI}$-TBox $\mathcal{T}$ is unraveling tolerant iff all templates $\mathcal{I}_{\mathcal{T},q}$ from Theorem 24 have tree obstructions. Consequently, for any $\mathcal{ALCI}$-TBox $\mathcal{T}$, ELIQs can be answered using an arc consistency algorithm iff $\mathcal{T}$ is unraveling tolerant.

5 Dichotomy for Depth One

We establish a dichotomy between PTIME and CONP for TBoxes of depth one, i.e., sets of CIs $C \sqsubseteq D$ such that the maximum nesting depth of the constructors $\exists r.E$ and $\forall r.E$ in C and D is one.[1] All examples given in the present paper up to this point use TBoxes of depth one.

Our main observation is that, when the depth of TBoxes is restricted to one, we can prove a converse of Theorem 17.

Theorem 18. *Every materializable $\mathcal{ALCFI}$-TBox of depth one is unraveling tolerant.*

Proof. (sketch) Let $\mathcal{T}$ be a materializable TBox of depth one, $\mathcal{A}$ an ABox, and q an ELIQ with $(\mathcal{T}, \mathcal{A}^u) \not\models q$. We have to show that $(\mathcal{T}, \mathcal{A}) \not\models q$. It follows from $(\mathcal{T}, \mathcal{A}^u) \not\models q$ that $\mathcal{A}^u$ is consistent w.r.t. $\mathcal{T}$ and thus there is a materialization $\mathcal{I}^u$ for $\mathcal{T}$ and $\mathcal{A}^u$ (even though $\mathcal{A}^u$ can be infinite, see long version). We have $\mathcal{I}^u \not\models q$ and our aim is to convert $\mathcal{I}^u$ into a model $\mathcal{I}$ of $\mathcal{T}$ and $\mathcal{A}$ such that $\mathcal{I} \not\models q$. This is done in two steps.

As a preliminary to the first step, we note that $\mathcal{I}^u$ can be assumed w.l.o.g. to have forest-shape, i.e., $\mathcal{I}^u$ can be constructed by selecting a tree-shaped interpretation $\mathcal{I}_\alpha$ with root α for each $\alpha \in \mathsf{Ind}(\mathcal{A}^u)$, then taking the disjoint union of all these interpretations, and finally adding role edges (α, β) to $r^{\mathcal{I}^u}$ whenever $r(\alpha, \beta) \in \mathcal{A}^u$. In fact, to achieve the desired shape we can simply unravel $\mathcal{I}^u$ starting from the elements $\mathsf{Ind}(\mathcal{A}^u) \subseteq \Delta^{\mathcal{I}^u}$ and then use Point 1 of Lemma 8 and the fact that there is an i-simulation from the unraveling of $\mathcal{I}^u$ to $\mathcal{I}^u$ to show that the obtained model is still a materialization of $\mathcal{T}$ and $\mathcal{A}$.

[1]Our results even apply to TBoxes that have depth one *after* replacing all $\mathcal{ELI}$-subconcepts with concept names, since $\mathcal{ELI}$-concept definitions do not affect the complexity of ELIQ-answering. This captures >90% of the TONES repository TBoxes.

Now, step one of the construction is to uniformize $\mathcal{I}^u$ such that for all $\alpha, \beta \in \mathsf{Ind}(\mathcal{A}^u)$ with $\mathsf{tail}(\alpha) = \mathsf{tail}(\beta)$, the tree component $\mathcal{I}_\alpha$ of $\mathcal{I}^u$ is isomorphic to the tree component $\mathcal{I}_\beta$ of $\mathcal{I}^u$. To achieve this while preserving the property that $\mathcal{I}^u \not\models q$, we rely on the self-similarity of the ABox $\mathcal{A}^u$: for all $\alpha, \beta \in \mathsf{Ind}(\mathcal{A}^u)$ with $\mathsf{tail}(\alpha) = \mathsf{tail}(\beta)$, we can find an automorphism on $\mathcal{A}^u$ that maps α to β.

Step two is to construct the desired model $\mathcal{I}$ of $\mathcal{T}$ and the original ABox $\mathcal{A}$, starting from the uniformized version of $\mathcal{I}^u$: take the disjoint union of all the tree components $\mathcal{I}_a$ of $\mathcal{I}_u$, with $a \in \mathsf{Ind}(\mathcal{A})$ (note that $\mathsf{Ind}(\mathcal{A}) \subseteq \mathsf{Ind}(\mathcal{A}^u)$), and add (a, b) to $r^\mathcal{I}$ whenever $r(a, b) \in \mathcal{A}$. Due to the uniformity of $\mathcal{I}^u$, we can find an i-simulation from $\mathcal{I}$ to $\mathcal{I}_u$. Since matches of ELIQs are preserved under i-simulations, $\mathcal{I}_u \not\models q$ thus implies $\mathcal{I} \not\models q$. ❑

The desired dichotomy follows: If an $\mathcal{ALCFI}$-TBox $\mathcal{T}$ of depth one is materializable, then PEQ-answering w.r.t. $\mathcal{T}$ is in PTIME by Theorems 18 and 16. Otherwise, ELIQ-answering w.r.t. $\mathcal{T}$ is CONP-complete by Theorem 11.

Theorem 19 (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}.*

We close this section by a brief discussion of why analyzing the complexity of query answering is easier for TBoxes of depth one than for TBoxes of unrestricted depth, when there is no such difference for other reasoning problems such as subsumption. Of course, every TBox can be converted to a TBox of depth one by introducing additional concept names A_C that replace compound concepts C which occur as an argument in $\exists r.C$ or $\forall r.C$. The problem is that these concept names can then be used in a CQ, which results in an 'import' of C into the query language. This is obviously problematic, for example when C has the form $\forall r.D$, which is otherwise not expressible as a CQ. In the next section, we will use this effect to reduce CSPs to query answering with TBoxes of depth > 1.

6 Query Answering in $\mathcal{ALC}/\mathcal{ALCI}$ = CSP

We show that query answering w.r.t. $\mathcal{ALC}$- and $\mathcal{ALCI}$-TBoxes has the same computational power as non-uniform CSPs in the following sense: (i) for every CSP, there is an $\mathcal{ALC}$-TBox such that query answering w.r.t. $\mathcal{T}$ is of the same complexity, up to complementation; conversely, (ii) for every $\mathcal{ALCI}$-TBox $\mathcal{T}$ and ELIQ q, there is a CSP that has the same complexity as answering q w.r.t. $\mathcal{T}$, up to complementation. This has many interesting consequences, a main one being that the Feder-Vardi conjecture holds if and only if there is a PTIME/CONP-dichotomy for query answering w.r.t. $\mathcal{ALC}$-TBoxes (equivalently $\mathcal{ALCI}$-TBoxes). All this is true already for *materializable* TBoxes. By Theorem 4 and since we carefully choose the appropriate query language in each technical result below, it is true for any of the languages ELIQ, CQ, and PEQ (and ELQ for $\mathcal{ALC}$-TBoxes).

We begin by introducing non-uniform CSPs. Since every non-uniform CSP is polynomially equivalent to a non-uniform CSP with one binary predicate (Feder and Vardi 1993), we consider CSPs over unary and binary predicates (concept names and role names), only. A *signature* Σ is a finite set of concept and role names. An interpretation $\mathcal{I}$ is a Σ*-interpretation* if $\mathsf{Ind}(\mathcal{I}) = \emptyset$ and $X^\mathcal{I} = \emptyset$ for all $X \in (\mathsf{N_C} \cup \mathsf{N_R}) \setminus \Sigma$. For two finite Σ-interpretations $\mathcal{I}$ and $\mathcal{I}'$, we write $\mathsf{Hom}(\mathcal{I}', \mathcal{I})$ if there is a homomorphism from $\mathcal{I}'$ to $\mathcal{I}$. Any Σ-interpretation $\mathcal{I}$ gives rise to the following non-uniform constraint satisfaction problem in signature Σ, denoted by $\mathsf{CSP}(\mathcal{I})$: given a finite Σ-interpretation $\mathcal{I}'$, decide whether $\mathsf{Hom}(\mathcal{I}', \mathcal{I})$. Numerous algorithmic problems can be given in the form $\mathsf{CSP}(\mathcal{I})$. For example, k-colorability is $\mathsf{CSP}(\mathsf{C}_k)$, where C_k is an $\{r\}$-interpretation defined by setting $\Delta^{\mathsf{C}_k} = \{1, \ldots, k\}$ and $r^{\mathsf{C}_k} = \{(i, j) \mid i \neq j\}$.[2]

We first show how to convert a CSP into a (materializable) $\mathcal{ALC}$-TBox. For a Σ-interpretation $\mathcal{I}$, $\mathcal{A}_\mathcal{I}$ denotes $\mathcal{I}$ viewed as an ABox: $\mathcal{A}_\mathcal{I} = \{A(a_d) \mid A \in \Sigma \cap \mathsf{N_C} \wedge d \in A^\mathcal{I}\} \cup \{r(a_d, a_e) \mid r \in \Sigma \cap \mathsf{N_R} \wedge (d, e) \in r^\mathcal{I}\}$.

Theorem 20. *For every non-uniform constraint satisfaction problem $\mathsf{CSP}(\mathcal{I})$ in signature Σ, one can compute (in polytime) a materializable $\mathcal{ALC}$-TBox $\mathcal{T}_\mathcal{I}$ such that*

1. *$\mathsf{Hom}(\mathcal{J}, \mathcal{I})$ iff $\mathcal{A}_\mathcal{J}$ is consistent w.r.t. $\mathcal{T}_\mathcal{I}$, for all Σ-interpretations $\mathcal{J}$;*
2. *for any Boolean PEQ q, answering q w.r.t. $\mathcal{T}_\mathcal{I}$ is polynomially reducible (in fact, FO-reducible) to the complement of $\mathsf{CSP}(\mathcal{I})$.*

Note that $\mathsf{CSP}(\mathcal{I})$ and $\mathcal{T}_\mathcal{I}$ 'have the same complexity' in the following sense: by Point 1 of Theorem 20, $\mathsf{CSP}(\mathcal{I})$ reduces to consistency of ABoxes w.r.t. $\mathcal{T}_\mathcal{I}$; since an ABox $\mathcal{A}$ is consistent w.r.t. $\mathcal{T}_\mathcal{I}$ iff $(\mathcal{T}_\mathcal{I}, \mathcal{A}) \not\models A(a)$ with A a fresh concept name and $a \in \mathsf{Ind}(\mathcal{A})$, this also yields a reduction from the complement of $\mathsf{CSP}(\mathcal{I})$ to ELQ-answering w.r.t. $\mathcal{T}_\mathcal{I}$; conversely, Point 2 ensures that (Boolean) PEQ-answering w.r.t. $\mathcal{T}_\mathcal{I}$ reduces to the complement of $\mathsf{CSP}(\mathcal{I})$. All reductions are extremely simple, in polytime and in fact even FO-reductions.

Our approach to proving Theorem 20 is to generalize the reduction of k-colorability to query answering w.r.t. $\mathcal{ALC}$-TBoxes discussed in Examples 1 and 3, where the main challenge is to overcome the observation from Example 3 that PTIME CSPs such as 2-colorability may be translated into CONP-hard TBoxes. Note that this is due to the disjunction in the TBox $\mathcal{T}_k$ of Example 1, which causes non-materializability. Our solution is to replace the concept names $A_1, \ldots, A_k$ in $\mathcal{T}_k$ with compound concepts that are 'invisible to the query', behaving essentially like second-order variables. Unlike the original depth one TBox $\mathcal{T}_k$, the resulting TBox is of depth three.

In detail, fix a constraint satisfaction problem $\mathsf{CSP}(\mathcal{I})$, reserve a concept name Z_d and role names r_d, s_d for any $d \in \Delta^\mathcal{I}$, and set

$$\begin{aligned} \mathcal{T} &= \{\top \sqsubseteq \exists r_d.\top, \top \sqsubseteq \exists s_d.Z_d \mid d \in \Delta^\mathcal{I}\} \\ H_d &= \forall r_d.\exists s_d.\neg Z_d, \quad d \in \Delta^\mathcal{I} \end{aligned}$$

[2]Although the input to $\mathsf{CSP}(\mathsf{C}_k)$ formally is a digraph, it is treated like an undirected graph.

The following shows that we can use the concepts H_d as unary predicates to represent the 'values' of $\mathsf{CSP}(\mathcal{I})$ (these values are the domain elements of $\mathcal{I}$).

Lemma 21. *For every ABox $\mathcal{A}$ and family of sets $I_d \subseteq \mathsf{Ind}(\mathcal{A})$, $d \in \Delta^\mathcal{I}$, there is a materialization $\mathcal{J}$ of $\mathcal{T}$ and $\mathcal{A}$ such that $H_d^\mathcal{J} = I_d$ for all $d \in \Delta^\mathcal{I}$.*

Now, the TBox $\mathcal{T}_\mathcal{I}$ for $\mathsf{CSP}(\mathcal{I})$ in signature Σ from Theorem 20 is $\mathcal{T}$ extended with the following CIs:

$$\begin{array}{rl} \top \sqsubseteq \bigsqcup_{d\in\Delta^\mathcal{I}} H_d & \\ H_d \sqcap H_e \sqsubseteq \bot & \text{for all } d,e \in \Delta^\mathcal{I}, d \neq e \\ H_d \sqcap \exists r.H_e \sqsubseteq \bot & \text{for all } d,e \in \Delta^\mathcal{I}, r \in \Sigma, (d,e) \notin r^\mathcal{I} \\ H_d \sqcap A \sqsubseteq \bot & \text{for all } d \in \Delta^\mathcal{I}, A \in \Sigma, d \notin A^\mathcal{I}. \end{array}$$

Based on Lemma 21, it can be verified that $\mathcal{T}_\mathcal{I}$ satisfies Conditions 1 and 2 of Theorem 20. For Point 2, we show that for all Boolean PEQs q and ABoxes $\mathcal{A}$, $(\mathcal{T}_\mathcal{I},\mathcal{A}) \models q$ iff $(\mathcal{T},\mathcal{A}) \models q$ or not $\mathsf{Hom}(\mathcal{I}_\mathcal{A}^\Sigma, \mathcal{I})$ where $\mathcal{I}_\mathcal{A}^\Sigma$ is the restriction of $\mathcal{I}_\mathcal{A}$ to signature Σ and with $\mathsf{Ind}(\mathcal{I}_\mathcal{A}^\Sigma) = \emptyset$. Moreover, it is not hard to see that $\mathcal{T}$ is unraveling tolerant, thus $(\mathcal{T},\mathcal{A}) \models q$ is in PTIME.

We now come to the conversion of an $\mathcal{ALCI}$-TBox and query q into a CSP. We start with considering Boolean CQs of the form $\exists x.C(x)$ with C an $\mathcal{ELI}$-concept, which is not strong enough to obtain the desired dichotomy result, but serves as a warmup that is conceptually cleaner than the version for ELIQs that we present afterwards. We use $\mathsf{sig}(\mathcal{T})$ to denote the signature of the TBox $\mathcal{T}$, and likewise for a CQ q.

Theorem 22. *Let $\mathcal{T}$ be an $\mathcal{ALCI}$-TBox, $q = \exists x.C(x)$ with C an $\mathcal{ELI}$-concept, and $\Sigma = \mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(q)$. Then one can construct (in time exponential in $|\mathcal{T}| + |C|$) a Σ-interpretation $\mathcal{I}_{\mathcal{T},q}$ such that for all ABoxes $\mathcal{A}$:*

$$(\mathsf{HomDual}) \quad (\mathcal{T},\mathcal{A}) \models q \text{ iff not } \mathsf{Hom}(\mathcal{I}_\mathcal{A}^\Sigma, \mathcal{I}_{\mathcal{T},q})$$

Proof.(sketch) The interpretation $\mathcal{I}_{\mathcal{T},q}$ can be obtained using a standard type-based construction. We use the sets $\mathsf{cl}(\mathcal{T},C)$, $\mathsf{tp}(\mathcal{T},C)$, and the relation $\rightsquigarrow_r$ between types as defined in the proof of Theorem 16. A *$\mathcal{T}$-type t that omits q* is an element of $\mathsf{tp}(\mathcal{T},C)$ that is satisfiable in a model $\mathcal{J}$ of $\mathcal{T}$ with $C^\mathcal{J} = \emptyset$. Then $\Delta^{\mathcal{I}_{\mathcal{T},q}}$ is the set of all $\mathcal{T}$-types that omit q, $t \in A^{\mathcal{I}_{\mathcal{T},q}}$ iff $A \in t$, for all $A \in \Sigma$, and $(t,t') \in r^{\mathcal{I}_{\mathcal{T},q}}$ iff $t \rightsquigarrow_r t'$, for all $r \in \Sigma$. It is shown in the long version that condition (HomDual) is satisfied. A Pratt-style type elimination algorithm can be used to construct $\mathcal{I}_{\mathcal{T},q}$ in exponential time (Pratt 1979). ❑

Example 23. Let $\mathcal{T} = \{A \sqsubseteq \forall r.B\}$ and define $q = \exists x.B(x)$. Then $\mathcal{I}_{\mathcal{T},q}$ is defined, up to isomorphism, by $\Delta^{\mathcal{I}_{\mathcal{T},q}} = \{a,b,c\}$, $A^{\mathcal{I}_{\mathcal{T},q}} = \{b\}$, $B^{\mathcal{I}_{\mathcal{T},q}} = \emptyset$, and $r^{\mathcal{I}_{\mathcal{T},q}} = \{(a,a),(a,b),(a,c)\}$. ⊣

For ELIQs, the conversion of a TBox and query into a CSP is similar to the construction above, but employs a concept name P that represents the individual name used in the ELIQ.

Theorem 24. *Let $\mathcal{T}$ be an $\mathcal{ALCI}$-TBox, $C(a)$ an ELIQ and $\Sigma = \mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(C) \cup \{P\}$, where P is a fresh concept name. Then one can construct (in time exponential in $|\mathcal{T}| + |C|$) a Σ-interpretation $\mathcal{I}_{\mathcal{T},q}$ such that for all ABoxes $\mathcal{A}$:*

1. *$(\mathcal{T},\mathcal{A}) \models C(a)$ iff not $\mathsf{Hom}(\mathcal{I}_{\mathcal{A}'}^\Sigma, \mathcal{I}_{\mathcal{T},q})$, where $\mathcal{A}'$ is obtained from $\mathcal{A}$ by adding $P(a)$ and removing all other assertions that use P;*
2. *$(\mathcal{T},\mathcal{A}) \models \exists x.(P(x) \wedge C(x))$ iff not $\mathsf{Hom}(\mathcal{I}_\mathcal{A}^\Sigma, \mathcal{I}_{\mathcal{T},q})$.*

As a consequence of Theorems 20 and 24, we obtain:

Theorem 25. *There is a dichotomy between* PTIME *and* CONP *for CQ-answering w.r.t. $\mathcal{ALC}$-TBoxes if and only if the Feder-Vardi conjecture is true.*

The same is true for $\mathcal{ALCI}$-TBoxes, for ELIQs, and PEQs. For $\mathcal{ALC}$-TBoxes, it additionally holds for ELQs.

Proof. 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 $\mathcal{T}_\mathcal{I}$ from Theorem 20. By Point 1 of that theorem (and the mentioned reduction of ABox consistency to the complement of ELQ-answering), 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.

Conversely, let $\mathcal{T}$ be an $\mathcal{ALC}$-TBox for which CQ-answering w.r.t. $\mathcal{T}$ is neither in PTIME nor CONP-hard. Then by Theorem 4 and since every ELIQ is a CQ, the same holds for ELIQ-answering w.r.t. $\mathcal{T}$. It follows that there is concrete ELIQ q such that answering q w.r.t. $\mathcal{T}$ is CONP-intermediate. Let $\mathcal{I}_{\mathcal{T},q}$ be the interpretation constructed in Point 1 of Theorem 24. By Point 1 of that theorem, $\mathsf{CSP}(\mathcal{I})$ is not in PTIME; by Point 2, it is not NP-hard. ❑

The construction underlying Theorem 24 cannot be generalized from ELIQs to CQs. To discuss this further, let us consider the simpler 'warmup' Theorem 22 instead. We show that it is impossible to construct an interpretation $\mathcal{I}_{\mathcal{T},q}$ which satisfies (HomDual) for Boolean CQs that are not of the simple form $\exists x.C(x)$. This is true even when the TBox is*empty*. It is thus crucial to use ELIQs even when proving the dichotomy result for CQs and PEQs. The following theorem states this more formally.

Theorem 26. *Let q be a Boolean CQ without individual names, $\mathsf{sig}(q) = \Sigma$, and $\mathcal{T}_\emptyset$ the empty TBox. Then there is a Σ-interpretation $\mathcal{I}_{q,\mathcal{T}_\emptyset}$ that satisfies (HomDual) iff q is logically equivalent to a CQ of the form $\exists x.C(x)$ with C an $\mathcal{ELI}$-concept.*

Proof. This is a consequence of results on *homomorphism dualities* (Nesetril and Tardif 2000), the problem of constructing, for a given Σ-interpretation $\mathcal{I}$, a Σ-interpretation $\overline{\mathcal{I}}$ such that the following duality holds for all Σ-interpretations $\mathcal{J}$:

$$\mathsf{Hom}(\mathcal{I},\mathcal{J}) \text{ iff not } \mathsf{Hom}(\mathcal{J},\overline{\mathcal{I}}).$$

By (Nesetril and Tardif 2000; Nesetril 2009), such an $\overline{\mathcal{I}}$ exists iff the undirected graph induced by $\mathcal{I}$ is a tree. It remains to observe that for any Boolean CQ q without individual names and all Σ-interpretations $\mathcal{J}$, we have $\mathcal{A}_\mathcal{J} \models q$ iff $\mathsf{Hom}(\mathcal{I}_q,\mathcal{J})$, where $\mathcal{I}_q$ is the interpretation with $\Delta^{\mathcal{I}_q}$ the variables in q and in which $x \in A^{\mathcal{I}_q}$ (resp. $(x,y) \in r^{\mathcal{I}_q}$) if $A(x)$ (resp. $r(x,y)$) is a conjunct of q. ❑

Interestingly, (Nesetril 2009) presents five constructions of $\overline{\mathcal{I}}$, one of which resembles our type elimination procedure (but, of course, without taking into account TBoxes).

7 Non-Dichotomy in $\mathcal{ALCF}$

We show that the complexity landscape for query answering w.r.t. $\mathcal{ALCF}$-TBoxes is much richer than for $\mathcal{ALCI}$. In particular, we show that for CQ-answering w.r.t. $\mathcal{ALCF}$-TBoxes, there is no dichotomy between PTIME and CONP unless PTIME = NP. This is a consequence of the following, much stronger, result.

Theorem 27. *For every language $L \in$ CONP, there is an $\mathcal{ALCF}$-TBox $\mathcal{T}$ and ELIQ $\mathsf{rej}(a)$, rej a concept name, such that the following holds:*

1. *there is a polynomial reduction of 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 L.*

We use Theorem 27 to establish that there is no PTIME/CONP-dichotomy (unless PTIME = NP). Assume to the contrary of what is to be shown that for every $\mathcal{ALCF}$-TBox $\mathcal{T}$, CQ answering w.r.t. $\mathcal{T}$ is in PTIME or CONP-hard. By Ladner's Theorem, there is a CONP-intermediate language L. Let $\mathcal{T}$ be the TBox from Theorem 27. 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 4 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 27, this is impossible.

The proof of Theorem 27 combines the 'hidden' concepts H_d from the proof of Theorem 20 with a modification of the TBox constructed in (Baader et al. 2010) to prove the undecidability of *query emptiness* in $\mathcal{ALCF}$. Using a similar strategy, one can also establish the following undecidability result.

Theorem 28. *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.*

8 Future Work

Much work remains to be done in order to fully accomplish the general research goal set out in the introduction. We propose four interesting directions.

(1) It would be interesting to consider additional complexity classes such as LOGSPACE, NLOGSPACE, and AC^0. The latter is particularly relevant in the context of FO-rewritability and the implementation of query answering using standard relational database systems, see (Calvanese et al. 2007) for details. Note that even for TBoxes of depth one, the complexity landscape is still rich. Relevant results can be found in (Calvanese et al. 2006): (i) there are $\mathcal{EL}$-TBoxes of depth one for which CQ-answering is PTIME-complete; and (ii) CQ-answering w.r.t. the $\mathcal{EL}$-TBox $\{\exists r.A \sqsubseteq A\}$, which encodes reachability in directed graphs, is NLOGSPACE-complete. We add that CQ-answering w.r.t. the Horn-$\mathcal{ALC}$-TBox $\{\exists r.A \sqsubseteq A, A \sqsubseteq \forall r.A\}$ corresponds to reachability in undirected graphs and can be shown to be LOGSPACE-complete. Also note that every DL-Lite TBox is of depth one, which gives a whole class of TBoxes for which CQ-answering is in AC^0.

(2) We conjecture that for a given $\mathcal{ALCFI}$-TBox of depth one, it is decidable whether CQ-answering is in PTIME, NLOGSPACE, LOGSPACE, and AC^0. A first step towards establishing these result is the observation from (Lutz and Wolter 2011) that FO-rewritability of CQ-answering, which is very closely related to CQ-answering being in AC^0, is decidable for $\mathcal{ALCFI}$-TBoxes of depth one. As an encouraging example of how results from the CSP world can be employed to obtain significant insights into CQ-answering, we note that for $\mathcal{ALCI}$-TBoxes, one can establish this result by using Theorem 24 and the fact that deciding FO-definability of a a CSP is in NP (Larose, Loten, and Tardif 2007). This yields a NEXPTIME upper bound due to the exponential size of the template constructed in the proof of Theorem 24.

(3) To better understand the complexity of TBoxes whose depth is larger than one, it would be interesting to generalize the notion of unraveling tolerance without leaving PTIME. In the CSP world, the corresponding notions of arc consistency and tree obstructions have both been significantly generalized, for example to structures of bounded treewidth (Bulatov, Krokhin, and Larose 2008).

(4) Alternatively to classifying the complexity of TBoxes while quantifying over *all* queries as in our Definition 2, one could also consider pairs $(\mathcal{T}, q)$ and classify the complexity of answering q w.r.t. $\mathcal{T}$, for all such pairs. Although some of our reductions to and from CSP consider such pairs, a systematic study of this problem will require new techniques.

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

References

Allender, E.; Bauland, M.; Immerman, N.; Schnoor, H.; and Vollmer, H. 2005. The complexity of satisfiability problems: Refining Schaefer's theorem. In *MFCS*, 71–82.

Artale, A.; Calvanese, D.; Kontchakov, R.; and Zakharyaschev, M. 2009. The DL-Lite family and relations. *J. Artif. Intell. Res. (JAIR)* 36:1–69.

Baader, F.; Calvanese, D.; McGuiness, D.; Nardi, D.; and Patel-Schneider, P. 2003. *The Description Logic Handbook: Theory, implementation and applications*. Cambridge University Press.

Baader, F.; Bienvenu, M.; Lutz, C.; and Wolter, F. 2010. Query and predicate emptiness in description logics. In *KR*.

Baader, F.; Brandt, S.; and Lutz, C. 2005. Pushing the $\mathcal{EL}$ envelope. In *IJCAI*, 364–369. Professional Book Center.

Baget, J.-F.; Mugnier, M.-L.; Rudolph, S.; and Thomazo, M. 2011. Walking the complexity lines for generalized guarded existential rules. In *IJCAI*, 712–717.

Barany, V.; Gottlob, G.; and Otto, M. 2010. Querying the guarded fragment. In *LICS*, 1–10.

Bulatov, A. A.; Jeavons, P.; and Krokhin, A. A. 2005. Classifying the complexity of constraints using finite algebras. *SIAM J. Comput.* 34(3):720–742.

Bulatov, A. A.; Krokhin, A. A.; and Larose, B. 2008. Dualities for constraint satisfaction problems. In *Complexity of Constraints*, 93–124.

Bulatov, A. A. 2002. A dichotomy theorem for constraints on a three-element set. In *FOCS*, 649–658.

Bulatov, A. A. 2011. On the CSP dichotomy conjecture. In *CSR*, 331–344.

Calvanese, D.; Giacomo, G. D.; Lenzerini, M.; and Vardi, M. Y. 2000. View-based query processing and constraint satisfaction. In *LICS*, 361–371.

Calvanese, D.; Giacomo, G. D.; Lenzerini, M.; and Vardi, M. Y. 2003a. Reasoning on regular path queries. *SIGMOD Record* 32(4):83–92.

Calvanese, D.; Giacomo, G. D.; Lenzerini, M.; and Vardi, M. Y. 2003b. View-based query containment. In *PODS*, 56–67.

Calvanese, D.; Giacomo, G. D.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2006. Data complexity of query answering in description logics. In *KR*, 260–270.

Calvanese, D.; De Giacomo, G.; Lembo, D.; Lenzerini, M.; and Rosati, R. 2007. Tractable reasoning and efficient query answering in description logics: The *DL-Lite* family. *J. of Autom. Reasoning* 39(3):385–429.

Chang, C. C., and Keisler, H. J. 1990. *Model Theory*, volume 73 of *Studies in Logic and the Foundations of Mathematics*. Elsevier.

Cosmadakis, S. S.; Gaifman, H.; Kanellakis, P. C.; and Vardi, M. Y. 1988. Decidable optimization problems for database logic programs (preliminary report). In *STOC*, 477–490.

Dechter, R. 2003. *Constraint processing*. Elsevier Morgan Kaufmann.

Dolby, J.; Fokoue, A.; Kalyanpur, A.; Ma, L.; Schonberg, E.; Srinivas, K.; and Sun, X. 2008. Scalable grounded conjunctive query evaluation over large and expressive knowledge bases. In *ISWC*, 403–418.

Eiter, T.; Gottlob, G.; Ortiz, M.; and Simkus, M. 2008. Query answering in the description logic Horn-$\mathcal{SHIQ}$. In *JELIA*, 166–179.

Feder, T., and Vardi, M. Y. 1993. Monotone monadic SNP and constraint satisfaction. In *STOC*, 612–622.

Gaifman, H.; Mairson, H. G.; Sagiv, Y.; and Vardi, M. Y. 1987. Undecidable optimization problems for database logic programs. In *LICS*, 106–115.

Glimm, B.; Lutz, C.; Horrocks, I.; and Sattler, U. 2008. Conjunctive query answering for the description logic $\mathcal{SHIQ}$. *JAIR* 31:157–204.

Hell, P., and Nesetril, J. 1990. On the complexity of h-coloring. *J. Comb. Theory, Ser. B* 48(1):92–110.

Hustadt, U.; Motik, B.; and Sattler, U. 2007. Reasoning in description logics by a reduction to disjunctive datalog. *J. Autom. Reasoning* 39(3):351–384.

Kazakov, Y. 2009. Consequence-driven reasoning for Horn-$\mathcal{SHIQ}$ ontologies. In Boutilier, C., ed., *IJCAI*, 2040–2045.

Kontchakov, R.; Lutz, C.; Toman, D.; Wolter, F.; and Zakharyaschev, M. 2010. The combined approach to query answering in DL-Lite. In *KR*.

Krisnadhi, A., and Lutz, C. 2007. Data complexity in the $\mathcal{EL}$ family of description logics. In *LPAR*, 333–347.

Krokhin, A. A. 2010. Tree dualities for constraint satisfaction. In *CSL*, 32–33.

Krötzsch, M.; Rudolph, S.; and Hitzler, P. 2007. Complexity boundaries for Horn description logics. In *AAAI*, 452–457.

Krötzsch, M. 2010. Efficient inferencing for OWL EL. In *JELIA*, 234–246.

Kun, G., and Szegedy, M. 2009. A new line of attack on the dichotomy conjecture. In *STOC*, 725–734.

Ladner, R. E. 1975. On the structure of polynomial time reducibility. *J. ACM* 22(1):155–171.

Larose, B.; Loten, C.; and Tardif, C. 2007. A characterisation of first-order constraint satisfaction problems. *Logical Methods in Computer Science* 3(4).

Lutz, C., and Wolter, F. 2011. Non-uniform data complexity of query answering in description logics. In *Description Logics*.

Lutz, C.; Piro, R.; and Wolter, F. 2011. Description logic tboxes: Model-theoretic characterizations and rewritability. In *IJCAI*.

Lutz, C.; Toman, D.; and Wolter, F. 2009. Conjunctive query answering in the description logic $\mathcal{EL}$ using a relational database system. In *IJCAI*, 2070–2075.

Makowsky, J. A. 1987. Why Horn formulas matter in computer science: Initial structures and generic examples. *J. Comput. Syst. Sci.* 34(2/3):266–292.

Malcev, A. 1971. *The metamathematics of algebraic systems, collected papers:1936-1967*. North-Holland.

Meseguer, J., and Goguen, J. 1985. Initiality, induction, and computability. In *Algebraic Methods in Semantics*. Cambridge University Press. 459–541.

Nesetril, J., and Tardif, C. 2000. Duality theorems for finite structures (characterising gaps and good characterisations). *J. Comb. Theory, Ser. B* 80(1):80–97.

Nesetril, J. 2009. Many facets of dualities. In *Research Trends in Combinatorial Optimization*. 285–302.

Otto, M.; Blumensath, A.; and Weyer, M. 2010. Decidability results for the boundedness problem. Technical report, TU Darmstadt.

Poggi, A.; Lembo, D.; Calvanese, D.; Giacomo, G. D.; Lenzerini, M.; and Rosati, R. 2008. Linking data to ontologies. *J. Data Semantics* 10:133–173.

Pratt, V. R. 1979. Models of program logics. In *FOCS*, 115–122.

Schaefer, T. J. 1978. The complexity of satisfiability problems. In *STOC*, 216–226.

Schaerf, A. 1993. On the complexity of the instance checking problem in concept languages with existential quantification. *J. of Intel. Inf. Systems* 2:265–278.