From Conjunctive Queries to Instance Queries in Ontology-Mediated Querying

Cristina Feier[1], **Carsten Lutz**[1], **Frank Wolter**[2],
[1] Unversity of Bremen, Germany
[2] University of Liverpool, UK
feier@uni-bremen.de, clu@uni-bremen.de, wolter@liverpool.ac.uk

Abstract

We consider ontology-mediated queries (OMQs) based on expressive description logics of the $\mathcal{ALC}$ family and (unions) of conjunctive queries, studying the rewritability into OMQs based on instance queries (IQs). Our results include exact characterizations of when such a rewriting is possible and tight complexity bounds for deciding rewritability. We also give a tight complexity bound for the related problem of deciding whether a given MMSNP sentence is equivalent to a CSP.

1 Introduction

An ontology-mediated query (OMQ) is a database-style query enriched with an ontology that contains domain knowledge, aiming to deliver more complete answers [Calvanese *et al.*, 2009; Bienvenu *et al.*, 2014; Bienvenu and Ortiz, 2015]. In OMQs, ontologies are often formulated in a description logic (DL) and query languages of interest include conjunctive queries (CQs), unions of conjunctive queries (UCQs), and instance queries (IQs). While CQs and UCQs are widely known query languages that play a fundamental role also in database systems and theory, IQs are more closely linked to DLs. In fact, an IQ takes the form $C(x)$ with C a concept formulated in the DL that is also used for the ontology, and thus the expressive power of IQs depends on the ontology language. OMQs based on (U)CQs are more powerful than OMQs based on IQs as the latter only serve to return all objects from the data that are instances of a given class.

It is easy to see that IQs can express tree-shaped CQs with a single answer variable as well as unions thereof. In fact, this observation has been used in many technical constructions in the area, see for example [Calvanese *et al.*, 1998; Glimm *et al.*, 2008; Lutz, 2008; Eiter *et al.*, 2012a]. Intriguingly, though, it was observed by Zolin [2007] that tree-shaped CQs are not the limit of IQ-rewritability when we have an expressive DL such as $\mathcal{ALC}$ or $\mathcal{ALCI}$ at our disposal. For example, the CQ $r(x,x)$, which asks to return all objects from the data that are involved in a reflexive r-loop, can be rewritten into the equivalent $\mathcal{ALC}$-IQ $P \rightarrow \exists r.P(x)$. Here, P behaves like a monadic second-order variable due to the open-world assumption made for OMQs: we are free to interpret P in any possible way and when making P true at an object we are forced to make also $\exists r.P$ true if and only if the object is involved in a reflexive r-loop. It is an interesting question, raised in [Zolin, 2007; Kikot and Zolin, 2013; Kikot *et al.*, 2013], to precisely characterize the class of CQs that are rewritable into IQs. An important step into this direction has been made by Kikot and Zolin [2013] who identify a large class of CQs that are rewritable into IQs: a CQ is rewritable into an $\mathcal{ALCI}$-IQ if it is connected and every cycle passes through the (only) answer variable; for rewritability into an $\mathcal{ALC}$-IQ, one additionally requires that all variables are reachable from the answer variable in a directed sense. It remained open whether these classes are depleting, that is, whether they capture all CQs that are IQ-rewritable.

There are two additional motivations to study the stated question. The first one comes from concerns about the practical implementation of OMQs. When the ontology is formulated in a more inexpressive 'Horn DL', OMQ evaluation is possible in PTIME data complexity and a host of techniques for practically efficient OMQ evaluation is available, see for example [Pérez-Urbina *et al.*, 2010; Eiter *et al.*, 2012b; Trivela *et al.*, 2015; Lutz *et al.*, 2009]. In the case of expressive DLs such as $\mathcal{ALC}$ and $\mathcal{ALCI}$, OMQ evaluation is CONP-complete in data complexity and efficient implementation is much more challenging. In particular, there are hardly any systems that fully support such OMQs when the actual queries are (U)CQs. In contrast, the evaluation of OMQs based on (expressive DLs and) IQs is supported by several systems such as Pellet, Hermit, and PAGOdA [Sirin *et al.*, 2007; Glimm *et al.*, 2014; Zhou *et al.*, 2015]. For this reason, rewriting (U)CQs into IQs has been advocated in [Zolin, 2007; Kikot and Zolin, 2013; Kikot *et al.*, 2013] as an approach towards efficient OMQ evaluation with expressive DLs and (U)CQs. The experiments and optimizations reported in [Kikot *et al.*, 2013] show the potential (and challenges) of this approach.

The second motivation stems from the connection between OMQs and constraint satisfaction problems (CSPs) [Bienvenu *et al.*, 2014; Lutz and Wolter, 2017]. Let $(\mathcal{L}, \mathcal{Q})$ denote the class of OMQs based on ontologies formulated in the DL $\mathcal{L}$ and the query language $\mathcal{Q}$. It was observed in [Bienvenu *et al.*, 2014] that $(\mathcal{ALCI}, \text{IQ})$ is closely related to the complement of CSPs while $(\mathcal{ALCI}, \text{UCQ})$ is closely related to the complement of the logical generalization MMSNP of CSP; we further remark that MMSNP is a notational variant of the complement of (Boolean) monadic disjunctive Datalog. Thus, characterizing OMQs from $(\mathcal{ALCI}, \text{UCQ})$ that are rewritable into $(\mathcal{ALCI}, \text{IQ})$ is related to characterizing MMSNP sentences that are equivalent to a CSP, and we also study the latter problem. In fact, the main differences to the OMQ case are that unary queries are replaced with Boolean ones and that predicates can have unrestricted arity.

The main aim of this paper is to study the rewritability of OMQs from $(\mathcal{L}, \text{(U)CQ})$ into OMQs from $(\mathcal{L}, \text{IQ})$, considering as $\mathcal{L}$ the basic expressive DL $\mathcal{ALC}$ as well as extensions of $\mathcal{ALC}$ with inverse roles, role hierarchies, the universal role, and functional roles. We provide precise characterizations, tight complexity bounds for deciding whether a given OMQ is rewritable, and show how to construct the rewritten query when it exists. In fact, we prove that the classes of CQs from [Kikot and Zolin, 2013] are depleting, but we go significantly beyond that: while [Zolin, 2007; Kikot and Zolin, 2013; Kikot *et al.*, 2013] aim to find IQ-rewritings that work for *any* ontology, we consider the more fine-grained question of rewriting into an IQ an OMQ $(\mathcal{T}, \Sigma, q(x))$ where $\mathcal{T}$ is a DL TBox formalizing the ontology, Σ is an ABox signature, and $q(x)$ is the actual query. The 'any ontology' setup then corresponds to the special case where $\mathcal{T}$ is empty and Σ is full. However, gaving a non-empty TBox or a non-full ABox signature results in additional (U)CQs to become rewritable. While we admit modification of the TBox during rewriting, it turns out that this is mostly unnecessary: only in some rather special cases, a moderate *extension* of the TBox pays off. All this requires non-trivial generalizations of the query classes and IQ-constructions from [Kikot and Zolin, 2013]. Our completeness proofs involve techniques that stem from the connection between OMQs and CSP such as a lemma about ABoxes of high girth due to Feder and Vardi [1998]. The rewritings we construct are of polynomial size when we work with the empty TBox, but can otherwise become exponential in size.

Regarding IQ-rewritability as a decision problem, we show NP-completeness for the case of the empty TBox. This can be viewed as an underapproximation for the case with non-empty TBox and ABox signature. With non-empty TBoxes, complexities are higher. When the ABox signature is full, we obtain 2EXPTIME-completeness for DLs with inverse roles and an EXPTIME lower bound and a coNEXPTIME upper bound for DLs without inverse roles. With unrestricted ABox signature, the problem is 2NEXPTIME-complete for DLs with inverse roles and NEXPTIME-hard (and in 2NEXPTIME) for DLs without inverse roles. All lower bounds hold for CQs and all upper bounds capture UCQs. We also prove that it is 2NEXPTIME-complete to decide whether a given MMSNP sentence is equivalent to a CSP. This problem was known to be decidable [Madelaine and Stewart, 2007], but the complexity was open.

We also consider $\mathcal{ALCIF}$, the extension of $\mathcal{ALCI}$ with functional roles, for which IQ-rewritability turns out to be undecidable and much harder to characterize. We give a rather subtle characterization for the case of the empty TBox and full ABox signature and show that the decision problem is then decidable and NP-complete. Since it is not clear how to apply CSP techniques, we use an approach based on ultrafilters, starting from what was done for $\mathcal{ALC}$ without functional roles in [Kikot and Zolin, 2013].

Full proofs are in the appendix.

2 Preliminaries

We use standard description logic notation and refer to [Baader *et al.*, 2017] for full details. In contrast to the standard DL literature, we carefully distinguish between the *concept language* and the *TBox language*. We consider four concept languages. Recall that $\mathcal{ALC}$*-concepts* are formed according to the syntax rule

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

where A ranges over *concept names* and r over *role names*. As usual, we use $C \rightarrow D$ as an abbreviation for $\neg C \sqcup D$. $\mathcal{ALCI}$*-concepts* additionally admit the use of *inverse roles* r^- in concept constructors $\exists r^-.C$ and $\forall r^-.C$. With a *role*, we mean a role name or an inverse role. $\mathcal{ALC}^u$*-concepts* additionally admit the use of the *universal role* u in concept constructors $\exists u.C$ and $\forall u.C$. In $\mathcal{ALCI}^u$*-concepts*, both inverse roles and the universal role are admitted.

We now introduce several TBox languages. For $\mathcal{L}$ one of the four concept languages introduced above, an $\mathcal{L}$*-TBox* is a finite set of *concept inclusions* $C \sqsubseteq D$ where C and D are $\mathcal{L}$ concepts. So each concept language also serves as a TBox language, but there are additional TBox languages of interest. We include the letter $\mathcal{H}$ in the name of a TBox language to indicate that *role inclusions* $r \sqsubseteq s$ are also admitted in the TBox and likewise for the letter $\mathcal{F}$ and *functionality assertions* $\mathsf{func}(r)$ where in both cases r, s are role names or inverse roles in case that the concept language used admits inverse roles. So it should be understood, for example, what we mean with an $\mathcal{ALCHI}^u$-TBox and an $\mathcal{ALCFI}$-TBox. As usual, the semantics is defined in terms of interpretations, which take the form $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ with $\Delta^\mathcal{I}$ a non-empty *domain* and $\cdot^\mathcal{I}$ an *interpretation function*. An interpretation is a *model* of a TBox $\mathcal{T}$ if it satisfies all inclusions and assertions in $\mathcal{T}$, defined in the usual way. We write $\mathcal{T} \models r \sqsubseteq s$ if every model of $\mathcal{T}$ also satisfies the role inclusion $r \sqsubseteq s$.

An *ABox* is a set of *concept assertions* $A(a)$ and *role assertions* $r(a, b)$ where A is a concept name, r a role name, and a, b are *individual names*. We use $\mathsf{ind}(\mathcal{A})$ to denote the set of all individual names that occur in $\mathcal{A}$. An interpretation is a *model* of an ABox $\mathcal{A}$ if it *satisfies* all concept and role assertions in $\mathcal{A}$, that is, $a \in A^\mathcal{I}$ when $A(a)$ is in $\mathcal{A}$ and $(a, b) \in r^\mathcal{I}$ when $r(a, b)$ is in $\mathcal{A}$. An ABox is *consistent with a TBox* $\mathcal{T}$ if $\mathcal{A}$ and $\mathcal{T}$ have a common model. A *signature* Σ is a set of concept and role names. We use $\mathsf{sig}(\mathcal{T})$ to denote the set of concept and role names that occur in the TBox $\mathcal{T}$, and likewise for other syntactic objects such as ABoxes. A Σ*-ABox* is an ABox $\mathcal{A}$ such that $\mathsf{sig}(\mathcal{A}) \subseteq \Sigma$.

A *conjunctive query (CQ)* is of the form $q(\mathbf{x}) = \exists \mathbf{y}\, \varphi(\mathbf{x}, \mathbf{y})$, where $\mathbf{x}$ and $\mathbf{y}$ are tuples of variables and $\varphi(\mathbf{x}, \mathbf{y})$ is a conjunction of *atoms* of the form $A(x)$ or $r(x, y)$ with A a concept name, r a role name, and $x, y \in \mathbf{x} \cup \mathbf{y}$. We call $\mathbf{x}$ the *answer variables* of $q(\mathbf{x})$ and $\mathbf{y}$ *quantified variables*. For purposes of uniformity, we use $r^-(x, y)$ as an alternative notation to denote an atom $r(y, x)$ in a CQ. In fact, when speaking about *an atom* $r(x,y)$ *in a CQ* $q(\mathbf{x})$, we generally also include the case that $r = s^-$ and $s(y,x)$ is the actual atom in $q(\mathbf{x})$, unless explicitly noted otherwise. Every CQ $q(\mathbf{x}) = \exists \mathbf{y}\, \varphi(\mathbf{x},\mathbf{y})$ gives raise to a directed graph G_q whose nodes are the elements of $\mathbf{x} \cup \mathbf{y}$ and that contains an edge from x to y if $\varphi(\mathbf{x},\mathbf{y})$ contains an atom $r(x,y)$. The corresponding undirected graph is denoted G^u_q (it might contain self loops). We can thus use standard terminology from graph theory to CQs, saying for example that a CQ is *connected*. A *homomorphism* from $q(\mathbf{x})$ to an interpretation $\mathcal{I}$ is a function $h : \mathbf{x} \cup \mathbf{y} \to \Delta^{\mathcal{I}}$ such that $h(x) \in A^{\mathcal{I}}$ for every atom $A(x)$ of $q(\mathbf{x})$ and $(h(x), h(y)) \in r^{\mathcal{I}}$ for every atom $r(x,y)$ of $q(\mathbf{x})$. We write $\mathcal{I} \models q(\mathbf{a})$ and call $\mathbf{a}$ an *answer to* $q(\mathbf{x})$ *on* $\mathcal{I}$ if there is a homomorphism from $q(\mathbf{x})$ to $\mathcal{I}$ with $h(\mathbf{x}) = \mathbf{a}$.

A *union of conjunctive queries (UCQ)* $q(\mathbf{x})$ is a disjunction of one or more CQs that all have the same answer variables $\mathbf{x}$. We say that a UCQ is *connected* if every CQ in it is. The *arity* of a (U)CQ is the number of answer variables in it. For $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALC}^u, \mathcal{ALCI}^u\}$, an *$\mathcal{L}$-instance query ($\mathcal{L}$-IQ)* takes the form $C(x)$ where C is an $\mathcal{L}$ concept and x a variable. We write $\mathcal{I} \models C(a)$ if $a \in C^{\mathcal{I}}$. All instance queries have arity 1.

An *ontology-mediated query (OMQ)* takes the form $Q = (\mathcal{T}, \Sigma, q(\mathbf{x}))$ with $\mathcal{T}$ a TBox, $\Sigma \subseteq \mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(q)$ an ABox signature, and $q(\mathbf{x})$ a query.[1] The *arity* of Q is the arity of $q(\mathbf{x})$ and Q is *Boolean* if it has arity zero. When Σ is $\mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(q)$, then for brevity we denote it with Σ_{full} and speak of the *full ABox signature*. Let $\mathcal{A}$ be a Σ-ABox. A tuple $\mathbf{a} \in \mathsf{ind}(\mathcal{A})$ is an *answer* to Q on $\mathcal{A}$ if $\mathcal{I} \models q(\mathbf{a})$ for all models $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}$. We say that Q is *empty* if for all Σ-ABoxes $\mathcal{A}$, there is no answer to Q on $\mathcal{A}$. Let Q_1, Q_2 be OMQs, $Q_i = (\mathcal{T}_i, \Sigma, q_i(\mathbf{x}))$ for $i \in \{1,2\}$. Then Q_1 is *contained* in Q_2, written $Q_1 \subseteq Q_2$, if for all Σ-ABoxes $\mathcal{A}$, every answer to Q_1 on $\mathcal{A}$ is also an answer to Q_2 on $\mathcal{A}$. Further, Q_1 and Q_2 are *equivalent*, written $Q_1 \equiv Q_2$, if $Q_1 \subseteq Q_2$ and $Q_2 \subseteq Q_1$.

We use $(\mathcal{L}, \mathcal{Q})$ to refer to the *OMQ language* in which the TBox is formulated in the language $\mathcal{L}$ and where the actual queries are from the language $\mathcal{Q}$, such as in $(\mathcal{ALCF}, \text{UCQ})$. For brevity, we generally write $(\mathcal{L}, \text{IQ})$ instead of $(\mathcal{L}, \mathcal{L}'\text{-IQ})$ when $\mathcal{L}'$ is the concept language underlying the TBox language $\mathcal{L}$, so for example $(\mathcal{ALCHI}, \text{IQ})$ is short for $(\mathcal{ALCHI}, \mathcal{ALCI}\text{-IQ})$.

Definition 1. Let $(\mathcal{L}, \mathcal{Q})$ be an OMQ language. An OMQ $Q = (\mathcal{T}, \Sigma, q(\mathbf{x}))$ is *$(\mathcal{L}, \mathcal{Q})$-rewritable* if there is an OMQ Q' from $(\mathcal{L}, \mathcal{Q})$ such that the answers to Q and to Q' are identical on any Σ-ABox that is consistent with $\mathcal{T}$. In this case, we say that *Q is rewritable* into Q' and call Q' a *rewriting* of Q.

Let $(\mathcal{L}, \mathcal{Q})$ be an OMQ-language. *IQ-rewritability in* $(\mathcal{L}, \mathcal{Q})$ is the problem to decide whether a given (unary) OMQ $Q = (\mathcal{T}, \Sigma, q(x))$ from $(\mathcal{L}, \mathcal{Q})$ is $(\mathcal{L}, \text{IQ})$-rewritable; for brevity, we simply speak of *IQ-rewritability* of Q when this is the case. The following examples show that IQ-rewritability of Q depends on several factors. All claims made are sanctioned by results established in this paper.

Example 2. (1) IQ-rewritability depends on the topology of the actual query. Let $q_1(x) = r(x,x)$. The OMQ $(\emptyset, \Sigma_{\mathsf{full}}, q_1(x))$ is rewritable into the OMQ $(\emptyset, \Sigma_{\mathsf{full}}, C(x))$ from $(\mathcal{ALC}, \text{IQ})$ where C is $P \to \exists r.P$. In contrast, let $q_2(x) = \exists y\, s(x,y) \wedge r(y,y)$. The OMQ $(\emptyset, \Sigma_{\mathsf{full}}, q_2(x))$ is not rewritable into an OMQ from $(\mathcal{ALCI}, \text{IQ})$.

(2) If we are not allowed to extend the TBox, IQ-rewritability depends on whether or not inverse roles are available. Let $\Sigma = \{r, s\}$ and $q(x) = \exists y\, r(y,x) \wedge s(y,x)$. The OMQ $Q = (\emptyset, \Sigma, q(x))$ is rewritable into the OMQ $(\emptyset, \Sigma, C(x))$ from $(\mathcal{ALCI}, \text{IQ})$ where C is $P \to \exists r^-.\exists s.P$. Q is also rewritable into the OMQ $(\mathcal{T}, \Sigma, C'(x))$ from $(\mathcal{ALC}, \text{IQ})$ where $\mathcal{T} = \{\exists s.P \sqsubseteq \forall r.P'\}$, and C is $P \to P'$, but it is not rewritable into any OMQ $(\mathcal{T}, \Sigma, C''(x))$ from $(\mathcal{ALC}, \text{IQ})$ with $\mathcal{T} = \emptyset$.

(3) IQ-rewritability depends on the TBox. Let $q(x) = \exists x_1 \exists y_1 \exists y_2 \exists z\, A(x) \wedge r(x,x_1) \wedge r(x_1,y_1) \wedge r(x_1,y_2) \wedge r(y_1,z) \wedge r(y_2,z) \wedge B_1(y_1) \wedge B_2(y_2)$. The OMQ $(\emptyset, \Sigma_{\mathsf{full}}, q(x))$ is not rewritable into an OMQ from $(\mathcal{ALCI}, \text{IQ})$. Let $\mathcal{T} = \{A \sqsubseteq \exists r.\exists r.(B_1 \sqcap B_2 \sqcap \exists r.\top)\}$. The OMQ $(\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$ is rewritable into the OMQ $(\mathcal{T}, \Sigma_{\mathsf{full}}, A(x))$ from $(\mathcal{ALC}, \text{IQ})$.

(4) IQ-rewritability depends on the ABox signature. Let $q(x)$ be the CQ from (3) without the atom $A(x)$ and let $\mathcal{T}$ be as in (3). The OMQ $(\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$ is not rewritable into an OMQ from $(\mathcal{ALCI}, \text{IQ})$. Let $\Sigma = \{A\}$. The OMQ $(\mathcal{T}, \Sigma, q(x))$ is rewritable into the OMQ $(\mathcal{T}, \Sigma, A(x))$ from $(\mathcal{ALC}, \text{IQ})$.

Note that we are allowed to completely rewrite the TBox when constructing IQ-rewritings, which might seem questionable from a practical perspective. Fortunately, though, it turns out the TBox can always be left untouched or, in some rare cases, only needs to be slightly extended. Also note that an alternative definition of IQ-rewritability obtained by dropping the restriction to ABoxes consistent with $\mathcal{T}$ in Definition 1. All results obtained in this paper hold under both definitions. We comment on this throughout the paper and refer to the alternative version as *unrestricted IQ-rewritability*.

3 Characterizations

We aim to provide characterizations of OMQs that are IQ-rewritable. On the one hand, these characterizations clarify which OMQs are IQ-rewritable and which are not. On the other hand, they form the basis for deciding IQ-rewritability. We first concentrate on the case of DLs (and IQs) with inverse roles and then move on to DLs without inverse roles. In the final part of this section, we consider the case where the TBox is empty, both with and without inverse roles.

3.1 The Case With Inverse Roles

To state the characterization, we need some preliminaries. Let $q(x)$ be a CQ. A *cycle* in $q(x)$ is a sequence of non-identical atoms $r_0(x_0,x_1), \dots, r_{n-1}(x_{n-1},x_n)$ in $q(x)$, $n \geq 1$, where[2]

1. $r_0, \dots, r_{n-1}$ are (potentially inverse) roles,
2. $x_i \neq x_j$ for $0 \leq i < j < n$, and $x_0 = x_n$.

[1] The requirement $\Sigma \subseteq \mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(q)$ is harmless since symbols in the ABox that are not from $\mathsf{sig}(\mathcal{T}) \cup \mathsf{sig}(q)$ do not affect answers.

[2] We require the atoms be non-identical to prevent $r(x_0,x_1)$, $r^-(x_1,x_0)$ from being a cycle (both atoms are identical).

The *length* of this cycle is n. We say that $q(x)$ is *x-acyclic* if every cycle in it passes through x and use $q^{\mathsf{con}}(x)$ to denote the result of restricting $q(x)$ to those atoms that only use variables reachable in G^u_q from x. Both notions are lifted to UCQs by applying them to every CQ in the UCQ. A *contraction* of $q(x)$ is a CQ obtained from $q(x)$ by zero or more variable identifications, where the identification of x with any other variable yields x.

Let $\mathcal{T}$ be an $\mathcal{ALCHI}^u$-TBox and $q(x)$ a UCQ. We use $q_{\mathsf{acyc}}(x)$ to denote the UCQ that consists of all x-acyclic CQs obtained by starting with a contraction of a CQ from $q(x)$ and then replacing zero or more atoms $r(y,z)$ with $s(y,z)$ when $\mathcal{T} \models s \sqsubseteq r$. We write $q^{\mathsf{con}}_{\mathsf{acyc}}(x)$ to denote $(q_{\mathsf{acyc}})^{\mathsf{con}}(x)$.

Theorem 3. *Let $\mathcal{L} \in \{\mathcal{ALCI}, \mathcal{ALCHI}\}$ and let $Q = (\mathcal{T}, \Sigma, q(x))$ be a unary OMQ from $(\mathcal{L}, UCQ)$ that is non-empty. Then the following are equivalent:*

1. *Q is IQ-rewritable, that is, it is rewritable into an OMQ $Q' = (\mathcal{T}', \Sigma, C(x))$ from $(\mathcal{L}, IQ)$;*
2. *Q is rewritable into an OMQ $Q' = (\mathcal{T}, \Sigma, C(x))$ from $(\mathcal{L}, IQ)$;*
3. *$Q \equiv (\mathcal{T}, \Sigma, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$.*

When $\mathcal{L}$ is replaced with $\mathcal{L}^u$, then the same equivalences hold except that $q^{\mathsf{con}}_{\mathsf{acyc}}$ is replaced with q_{acyc}.

Note that Theorem 3 excludes empty OMQs, but these are trivially IQ-rewritable. It implies that, in the considered cases, it is never necessary to modify the TBox when constructing an IQ-rewriting. Further, it emerges from the proof that it is never necessary to introduce fresh role names in the rewriting (while fresh concept names are crucial). Theorem 3 also applies to unrestricted IQ-rewritability (where also ABoxes are admitted that are inconsistent with the TBox from the OMQ): unrestricted IQ-rewritability trivially implies IQ-rewritability and the converse is an easy consequence of the fact that every OMQ that is IQ-rewritable has an IQ-rewriting based on the same TBox.

We now give some ideas about the proof of Theorem 3. The most interesting implication is "1 $\Rightarrow$ 3". A central step is to show that if $Q = (\mathcal{T}, \Sigma, q(x))$ is IQ-rewritable into an OMQ Q', then $Q \subseteq Q_{\mathsf{acyc}} := (\mathcal{T}, \Sigma, q_{\mathsf{acyc}}(x))$, that is, when $\mathcal{A} \models Q(a)$ for some Σ-ABox $\mathcal{A}$, then $\mathcal{A} \models Q_{\mathsf{acyc}}(a)$. To this end, we first construct from $\mathcal{A}$ a Σ-ABox $\mathcal{A}^g$ of high girth (that is, without small cycles) in a way such that (a) $\mathcal{A}^g$ homomorphically maps to $\mathcal{A}$ and (b) from $\mathcal{A} \models Q'(a)$ it follows that $\mathcal{A}^g \models Q'(a)$, thus $\mathcal{A}^g \models Q(a)$. Due to the high girth of $\mathcal{A}^g$ and exploiting (a variation of the) tree model property for $\mathcal{ALCHI}$, we can then show that $\mathcal{A}^g \models Q(a)$ implies $\mathcal{A}^g \models Q_{\mathsf{acyc}}(a)$. Because of (a), it follows that $\mathcal{A} \models Q_{\mathsf{acyc}}(a)$. In the direction "3 $\Rightarrow$ 2", we construct actual rewritings, based on the following lemma, an extension of a result of Kikot and Zolin [Kikot and Zolin, 2013] with TBoxes and ABox signatures (and UCQs instead of CQs).

Lemma 4. *Let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCHI}^u, UCQ)$. Then*

1. *if $q(x)$ is x-acyclic and connected, then Q is rewritable into an OMQ $(\mathcal{T}, \Sigma, C(x))$ with $C(x)$ an $\mathcal{ALCI}$-IQ and*
2. *if $q(x)$ is x-acyclic, then Q is rewritable into an OMQ $(\mathcal{T}, \Sigma, C(x))$ with $C(x)$ an $\mathcal{ALCI}^u$-IQ.*

The size of the IQs $C(x)$ is polynomial in the size of $q(x)$.

We give the construction of the $\mathcal{ALCI}$-IQ $q'(x)$ in Point 1 of Lemma 4. Let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCHI}^u, \text{UCQ})$ with $q(x)$ x-acyclic and connected. To construct $q'(x)$, we first construct for each CQ $p(x)$ in $q(x)$ an *$\mathcal{ELI}$-concept* C_p, that is, an $\mathcal{ALCI}$-concept that uses only the constructors $\sqcap$, $\exists r.C$, and $\exists r^-.C$. In fact, since $p(x)$ is x-acyclic and connected, we can repeatedly choose and remove atoms of the form $r(x,y)$ that occur in a cycle in $p(x)$ and will eventually end up with a tree-shaped CQ $p'(x)$.[3] Here, *tree-shaped* means that the undirected graph $G^u_{p'}$ is a tree and that there are no multi-edges, that is, if $r(y,z)$ is an atom, then there is no atom $s(y,z)$ with $s \neq r$. Next, extend $p'(x)$ to obtain another tree-shaped CQ $p''(x)$ by taking a fresh concept name $P \notin \Sigma$, and adding $r(x',y)$ and $P(x')$ for each removed atom $r(x,y)$, x' a fresh variable. We can now view $p''(x)$ as an $\mathcal{ELI}$-concept C_p in the obvious way. The desired $\mathcal{ALCI}$-IQ $q'(x)$ is $(P \rightarrow \bigsqcup_{p(x) \text{ a CQ in } q(x)} C_p)(x)$.

3.2 The Case Without Inverse Roles

We consider OMQs whose TBoxes are formulated in a DL $\mathcal{L}$ that does not admit inverse roles. Note that inverse roles are then also not admitted in the IQ used in the rewriting. We first observe that this has less impact than one might expect: inverse roles in the IQ-rewriting can be eliminated and in fact Points 1 and 3 from Theorem 3 are still equivalent. However, there is also a crucial difference: unless the universal role is present, the elimination of inverse roles requires an extension of the TBox and thus the equivalence of Points 1 and 2 of Theorem 3 fails. In fact, this is illustrated by Point (2) of Example 2. We thus additionally characterize IQ-rewritability without modifying the TBox. We also show that, with the universal role, it is not necessary to extend the TBox.

We start with some preliminaries. An *extended conjunctive query (eCQ)* is a CQ that also admits atoms of the form $C(x)$, C a (potentially compound) concept, and *UeCQs* and *extended ABoxes (eABoxes)* are defined analogously. The semantics is defined in the expected way. Every eCQ $q(x)$ gives rise to an eABox $\mathcal{A}_q$ by viewing the variables in $q(x)$ as individual names and the atoms as assertions.

Let $q(x)$ be an eCQ. We use $\mathsf{dreach}(q)$ to denote the set of all variables reachable from x in the directed graph G_q and say that $q(x)$ is *x-accessible* if $\mathsf{dreach}(q)$ contains all variables. For V a set of variables from $q(x)$ that includes x, $q(x)|_V$ denotes the restriction of $q(x)$ to the atoms that use only variables from V.

Let $\mathcal{T}$ be an $\mathcal{ALC}$-TBox. An eCQ $p(x)$ is a *$\mathcal{T}$-decoration* of a CQ $q(x)$ if

1. $p(x)$ is obtained from $q(x)$ by adding, for each $y \in \mathsf{dreach}(q)$ and each subconcept C of $\mathcal{T}$, the atom $C(y)$ or the atom $\neg C(y)$;
2. the eABox $\mathcal{A}_p$ is consistent with $\mathcal{T}$.

[3] Note that x is the answer variable and recall that we might have $r = s^-$ and thus also choose atoms $s(y,x)$.

For a UCQ $q(x)$, we use $q^{\mathsf{deco}}(x)$ to denote the UeCQ that consists of all eCQs $p(x)|_{\mathsf{dreach}(p)}(x)$, where $p(x)$ is a $\mathcal{T}$-decoration of a CQ from $q(x)$. We write $q^{\mathsf{deco}}_{\mathsf{acyc}}(x)$ to denote $(q_{\mathsf{acyc}})^{\mathsf{deco}}(x)$. We now give the results announced above.

Theorem 5. *Let $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCH}\}$ and let $Q = (\mathcal{T}, \Sigma, q(x))$ be a unary OMQ from $(\mathcal{L}, UCQ)$ that is non-empty. Then the following are equivalent:*

1. *Q is rewritable into an OMQ from $(\mathcal{L}, IQ)$;*
2. *Q is rewritable into an OMQ $(\mathcal{T} \cup \mathcal{T}', \Sigma, C(x))$ from $(\mathcal{L}, IQ)$;*
3. *Q is rewritable into an OMQ from $(\mathcal{LI}, IQ)$;*

If $\Sigma = \Sigma_{\mathsf{full}}$, then the following are equivalent:

4. *Q is rewritable into an OMQ $Q' = (\mathcal{T}, \Sigma_{\mathsf{full}}, C(x))$ from $(\mathcal{L}, IQ)$;*
5. *$Q \equiv (\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{deco}}_{\mathsf{acyc}}(x))$.*

If, furthermore, $\mathcal{L}$ is replaced with $\mathcal{L}^u$ and $\mathcal{LI}$ with $\mathcal{LI}^u$, then Conditions 1 to 3 are further equivalent to:

6. *Q is rewritable into an OMQ $Q' = (\mathcal{T}, \Sigma, C(x))$ from $(\mathcal{L}^u, IQ)$.*

Characterizing IQ-rewritability in the case where $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCH}\}$, the TBox (is non-empty) and cannot be extended, and $\Sigma \neq \Sigma_{\mathsf{full}}$ remains an open problem.

In the directions "3 ⇒ 2", "5 ⇒ 4", and "3 ⇒ 6", we have to construct IQ-rewritings. This is done by starting with the rewriting from the proof of Lemma 4 and then modifying it appropriately. As in the case of Theorem 3, it is straightforward to see that all results stated in Theorem 5 also apply to unrestricted IQ-rewritability.

3.3 The Case of Empty TBoxes

We consider OMQs in which the TBox is empty as an important special case. Since it is then not interesting to have an ABox signature, this corresponds to the rewritability of (U)CQs into $\mathcal{L}$-instance queries, for some concept language $\mathcal{L}$ (and thus no OMQs are involved). The importance of this case is due to the fact that it provides an 'underapproximation' of the IQ-rewritability of OMQs, while also being easier to characterize and computationally simpler.

We say that an UCQ $q(x)$ is *$\mathcal{L}$-IQ-rewritable* if there is an $\mathcal{L}$-IQ $q'(x)$ that is equivalent to $q(x)$ in the sense that the OMQs $(\emptyset, \Sigma_{\mathsf{full}}, q(x))$ and $(\emptyset, \Sigma_{\mathsf{full}}, q'(x))$ are equivalent (and in passing, we define the equivalence between two UCQs in exactly the same way). The following proposition makes precise what we mean by underapproximation.

Proposition 6. *Let $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALC}^u, \mathcal{ALCI}^u\}$. If a UCQ $q(x)$ is $\mathcal{L}$-IQ-rewritable, then so is any OMQ $(\mathcal{T}, \Sigma, q(x))$ from $(\mathcal{LH}, UCQ)$.*

Proposition 6 is essentially a corollary of Theorem 7 below. As illustrated by Case (3) of Example 2, its converse fails.

We now characterize IQ-rewritability in the case of the empty TBox. A *subquery* of a CQ $q(x)$ is a CQ $q'(x)$ obtained from $q(x)$ by dropping atoms. A *subquery* of a UCQ $q(x)$ is a UCQ obtained by including as a CQ at most one subquery of each CQ in $q(x)$.

Theorem 7. *Let $q(x)$ be a UCQ. Then*

1. *$q(x)$ is rewritable into an $\mathcal{ALCI}$-IQ iff there is a subquery $q'(x)$ of $q(x)$ that is x-acyclic, connected, and equivalent to $q(x)$;*
2. *$q(x)$ is rewritable into an $\mathcal{ALC}$-IQ iff there is a subquery $q'(x)$ of $q(x)$ that is x-acyclic, x-accessible, and equivalent to $q(x)$.*

When $\mathcal{L}$-IQs are replaced with $\mathcal{L}^u$-IQs, then the same equivalences hold except that connectedness/x-accessibility is dropped.

Note that Theorem 7 also characterizes rewritability of CQs; the query $q'(x)$ is then also a CQ rather than a UCQ. This is in contrast to Theorems 3 and 5 where the queries $q^{\mathsf{con}}_{\mathsf{acyc}}(x)$ and $q^{\mathsf{deco}}_{\mathsf{acyc}}(x)$ are UCQs even when the query $q(x)$ from the OMQ that we start with is a CQ. Another crucial difference is that $q^{\mathsf{con}}_{\mathsf{acyc}}(x)$ and $q^{\mathsf{deco}}_{\mathsf{acyc}}(x)$ can be of size exponential in the size of the original OMQ while the query $q'(x)$ in Theorem 7 is of size polynomial in the size of $q(x)$.

4 Complexity

We determine the complexity of deciding IQ-rewritability in various OMQ languages, based on the established characterizations and starting with the case of empty TBoxes.

Theorem 8. *For every $\mathcal{Q} \in \{CQ, UCQ\}$ and $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALC}^u, \mathcal{ALCI}^u\}$, it is* NP*-complete to decide whether a given query from $\mathcal{Q}$ is $\mathcal{L}$-IQ-rewritable.*

The upper bound in Theorem 8 is by guessing the query $q'(x)$ from Theorem 7 and verifying that it satisfies the properties stated there. The lower bound is by a reduction from 3-colorability.

We next consider the case where TBoxes can be non-empty, starting with the assumption that the ABox signature is full since this results in (slightly) lower complexity.

Theorem 9. *Let $\mathcal{Q} \in \{CQ, UCQ\}$. For OMQs based on the full ABox signature, IQ-rewritability is*

1. ExpTime*-hard in $(\mathcal{ALC}, \mathcal{Q})$ and in* coNExpTime *in $(\mathcal{ALCH}, \mathcal{Q})$ and*
2. 2ExpTime*-complete in $(\mathcal{ALCI}, \mathcal{Q})$ and $(\mathcal{ALCHI}, \mathcal{Q})$.*

The lower bounds are by reduction from OMQ evaluation on ABoxes of the form $\{A(a)\}$, A a concept name, which is ExpTime-complete in $(\mathcal{ALCH}, \text{CQ})$ and 2ExpTime-complete in $(\mathcal{ALCHI}, \text{CQ})$ [Lutz, 2008]. The upper bounds are derived from the OMQ containment checks suggested by Condition 3 of Theorem 3 and Condition 4 of Theorem 5. Since we work with the full ABox signature, the non-emptiness condition from these theorems is void (there are no empty OMQs) and OMQ containment is closely related to OMQ evaluation, which allows us to derive upper bounds for the former from the latter; in fact, these bounds are exactly the ones stated in Theorem 9. We have to exercise some care, for two reasons: first, we admit UCQs as the actual query and thus the trivial reduction of OMQ containment to OMQ evaluation that is possible for CQs (which can be viewed as an ABox) does not apply. And second, we aim for upper bounds that exactly match the complexity of OMQ containment while the UCQs $q^{\mathsf{con}}_{\mathsf{acyc}}(x)$ and $q^{\mathsf{deco}}_{\mathsf{acyc}}(x)$ involved in the containment checks are of exponential size. What rescues us is that each of the CQs in these UCQs is only of polynomial size.

We finally consider the case where the ABox signature is unrestricted.

Theorem 10. *IQ-rewritability is*

1. NExpTime-*hard in* $(\mathcal{ALC}, CQ)$ *and*
2. 2NExpTime-*complete in all of* $(\mathcal{ALCI}, CQ)$, $(\mathcal{ALCI}, UCQ)$, $(\mathcal{ALCHI}, CQ)$, $(\mathcal{ALCHI}, UCQ)$.

The lower bound in Point 1 is by reduction from OMQ emptiness in $(\mathcal{ALC}, \text{CQ})$, which is NExpTime-complete [Baader *et al.*, 2016]. For the one in Point 2, we use a reduction from OMQ containment, which is 2NExpTime-complete in $(\mathcal{ALCI}, \text{CQ})$ [Bourhis and Lutz, 2016]. The upper bounds are obtained by appropriate containment checks as suggested by our characterizations, and we again have to deal with UCQs with exponentially many CQs. Note that Theorem 10 leaves open the complexity of IQ-rewritability in $(\mathcal{ALC}, \text{CQ})$, between NExpTime and 2NExpTime. The same gap exists for OMQ containment [Bourhis and Lutz, 2016] as well as in the related problems of FO-rewritability and Datalog-rewritability [Feier *et al.*, 2017].

5 Functional Roles

We consider DLs with functional roles. A fundamental observation is that for the basic such DL $\mathcal{ALCF}$, IQ-rewritability is undecidable. This can be proved by a reduction from OMQ emptiness in $(\mathcal{ALCF}, \text{IQ})$ [Baader *et al.*, 2016].

Theorem 11. *In* $(\mathcal{ALCF}, CQ)$*, IQ-rewritability is undecidable.*

In the following, we show that decidability is regained in the case where the TBox is empty (apart from functionality assertions). This is challenging because functionality assertions have a strong and subtle impact on rewritability. As before, the only interesting ABox signature to be combined with 'empty' TBoxes is the full ABox signature. We use $\mathcal{F}$ to denote the TBox language in which TBoxes are sets of functionality assertions and concentrate on rewriting into IQs that may use inverse roles.

Example 12. Consider the CQ $p(x) = \exists y(s(x, y) \wedge r(y, y))$ from Point 1 of Example 2. Then $Q_s = (\mathcal{T}_s, \Sigma_{\mathsf{full}}, p(x))$ and $Q_r = (\mathcal{T}_r, \Sigma_{\mathsf{full}}, p(x))$ with $\mathcal{T}_w = \{\mathsf{func}(w)\}$ for $w \in \{r, s\}$ are both rewritable into an OMQ $(\mathcal{T}_w, \Sigma_{\mathsf{full}}, q_w(x))$ with $q_w(x)$ an $\mathcal{ALCI}$-IQ. The rewritings are neither trivial to find nor entirely easy to understand. In fact, for $q_s(x)$ we can use $\forall s.P \rightarrow \exists s.(P \rightarrow \exists r.P)$. For $q_r(x)$, we introduce three fresh concept names rather than a single one and use them in a way inspired by graph colorings:

$$q_r(x) = (\forall s. \bigsqcup_{1 \leq i \leq 3} P_i) \rightarrow (\exists s.(\bigsqcap_{1 \leq i \leq 3} (P_i \rightarrow \exists r.P_i)).$$

Before giving a characterization of rewritable queries, we introduce some preliminaries. Let $q(x)$ be a CQ and $\mathcal{T}$ an $\mathcal{ALCIF}$-TBox. A sequence $x_0, \dots, x_n$ of variables in $q(x)$ is a *functional path in* $q(x)$ *from* x_0 *to* x_n *w.r.t.* $\mathcal{T}$ if for all $i < n$ there is a role r such that $\mathsf{func}(r) \in \mathcal{T}$ and $r(x_i, x_{i+1})$ is in $q(x)$. We say that $q(x)$ is *f-acyclic w.r.t.* $\mathcal{T}$ if for every cycle $r_0(x_0, x_1), \dots, r_{n-1}(x_{n-1}, x_n)$ in $q(x)$, one of the following holds:

- there is a functional path in $q(x)$ from x to some x_i;
- $\mathsf{func}(r_i) \in \mathcal{T}$ or $\mathsf{func}(r_i^-) \in \mathcal{T}$ for all $i < n$ and there is a functional path $y_0, \dots, y_m$ in $q(x)$ with $x_0 = y_0 = y_m$ such that $\{x_0, \dots, x_{n-1}\} \subseteq \{y_0, \dots, y_m\}$.

We are now ready to state the characterization.

Theorem 13. *An OMQ* $Q = (\mathcal{T}, \Sigma_{\mathit{full}}, q(x))$ *from* $(\mathcal{F}, UCQ)$ *is rewritable into an OMQ from* $(\mathcal{F}, \mathcal{ALCI}\text{-}IQ)$ *iff there is a subquery* $q'(x)$ *of* $q(x)$ *that is f-acyclic, connected, and equivalent to* $q(x)$*.*

When $\mathcal{ALCI}$*-IQ is replaced with* $\mathcal{ALCI}^u$*-IQ, the same equivalence holds except that connectedness is dropped.*

The proof of Theorem 13 extends the ultrafilter construction from [Kikot and Zolin, 2013]. We remark that the "if" direction in Theorem 13 even holds for OMQs $Q = (\mathcal{T}, \Sigma, q(x))$ from $(\mathcal{ALCIF}, \text{UCQ})$. Thus, the case of the 'empty' TBox can again be seen as an underapproximation of the general case. We further remark that $\mathcal{T}$ remains unchanged in the construction of the IQ-rewritings and that the constructed rewritings are of polynomial size.

Theorem 14. *For OMQs from* $(\mathcal{F}, UCQ)$*, rewritability into* $(\mathcal{F}, \mathcal{ALCI}\text{-}IQ)$ *is* NP*-complete.*

6 MMSNP and CSP

Recall from the introduction that the OMQ languages studied in this paper are closely related to CSPs and their logical generalization MMSNP. In fact, the techniques used to establish the results in Sections 3 and 4 can be adapted to determine the complexity of deciding whether a given MMSNP sentence is equivalent to a CSP. In a nutshell, we prove that an MMSNP-sentence is equivalent to a CSP iff it is preserved under disjoint union and equivalent to a generalized CSP (a CSP with multiple templates), and that both properties can be reduced to containment between MMSNP sentences which is 2NExpTime-complete [Bourhis and Lutz, 2016]. The latter reduction involves constructing an MMSNP sentence φ_{acyc} that is reminiscent of the query q_{acyc} in Theorem 3. Full details are given in the appendix.

Theorem 15. *It is* 2NExpTime*-complete to decide whether a given MMSNP-sentence is equivalent to a CSP.*

7 Conclusion

We have made a leap forward in understanding the relation between (U)CQs and IQs in ontology-mediated querying. Interesting open problems include a characterization of IQ-rewritability for DLs with functional roles when the TBox is non-empty and characterizations for DLs with transitive roles. The remarks after Theorem 4 and 10 mention further problems left open. In addition, it would be worthwhile to continue the effort from [Kikot *et al.*, 2013] to understand the value of IQ-rewritings for the purposes of efficient practical implementation.

Acknowledgements. Cristina Feier and Carsten Lutz were supported by ERC Consolidator Grant 647289 CODA. Frank Wolter was supported by EPSRC UK grant EP/M012646/1.

References

[Baader *et al.*, 2016] Franz Baader, Meghyn Bienvenu, Carsten Lutz, and Frank Wolter. Query and predicate emptiness in ontology-based data access. *J. Artif. Intell. Res.*, 56:1–59, 2016.

[Baader *et al.*, 2017] Franz Baader, Ian Horrocks, Carsten Lutz, and Ulrike Sattler. *An Introduction to Description Logic*. Cambridge University Press, 2017.

[Bienvenu and Ortiz, 2015] Meghyn Bienvenu and Magdalena Ortiz. Ontology-mediated query answering with data-tractable description logics. In *Proc. of Reasoning Web*, volume 9203 of *LNCS*, pages 218–307. Springer, 2015.

[Bienvenu *et al.*, 2012] Meghyn Bienvenu, Carsten Lutz, and Frank Wolter. Query containment in description logics reconsidered. In *Proc. of KR2012*, 2012.

[Bienvenu *et al.*, 2014] Meghyn Bienvenu, Balder ten Cate, Carsten Lutz, and Frank Wolter. Ontology-based data access: A study through disjunctive Datalog, CSP, and MMSNP. *ACM Trans. Database Syst.*, 39(4):33:1–33:44, 2014.

[Blackburn *et al.*, 2002] Patrick Blackburn, Marten de Rijke, and Yde Venema. *Modal Logic*. Cambridge University Press, 2002.

[Bourhis and Lutz, 2016] Pierre Bourhis and Carsten Lutz. Containment in monadic disjunctive datalog, MMSNP, and expressive description logics. In *Proc. of KR*, pages 207–216. AAAI Press, 2016.

[Calvanese *et al.*, 1998] Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. On the decidability of query containment under constraints. In *Proc. of PODS1998*, pages 149–158. ACM Press, 1998.

[Calvanese *et al.*, 2009] Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Antonella Poggi, Mariano Rodriguez-Muro, and Riccardo Rosati. Ontologies and databases: The DL-Lite approach. In *Proc. of Reasoning Web 2009*, volume 5689 of *LNCS*, pages 255–356. Springer, 2009.

[Eiter *et al.*, 2012a] Thomas Eiter, Magdalena Ortiz, and Mantas Simkus. Conjunctive query answering in the description logic SH using knots. *J. Comput. Syst. Sci.*, 78(1):47–85, 2012.

[Eiter *et al.*, 2012b] Thomas Eiter, Magdalena Ortiz, Mantas Simkus, Trung-Kien Tran, and Guohui Xiao. Query rewriting for Horn-SHIQ plus rules. In *Proc. of AAAI*. AAAI Press, 2012.

[Feder and Vardi, 1998] Tomás Feder and Moshe Y. Vardi. The computational structure of monotone monadic SNP and constraint satisfaction: A study through datalog and group theory. *SIAM J. Comput.*, 28(1):57–104, 1998.

[Feier *et al.*, 2017] Cristina Feier, Antti Kuusisto, and Carsten Lutz. Rewritability in monadic disjunctive datalog, MMSNP, and expressive description logics (invited talk). In *Proc. ICDT*, volume 68 of *LIPIcs*, pages 1:1–1:17. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2017.

[Glimm *et al.*, 2008] Birte Glimm, Carsten Lutz, Ian Horrocks, and Ulrike Sattler. Conjunctive query answering for the description logic SHIQ. *J. Artif. Intell. Res.*, 31:157–204, 2008.

[Glimm *et al.*, 2014] Birte Glimm, Ian Horrocks, Boris Motik, Giorgos Stoilos, and Zhe Wang. Hermit: An OWL 2 reasoner. *J. of Autom. Reasoning*, 53(3):245–269, 2014.

[Kikot and Zolin, 2013] Stanislav Kikot and Evgeny Zolin. Modal definability of first-order formulas with free variables and query answering. *J. Applied Logic*, 11(2):190–216, 2013.

[Kikot *et al.*, 2013] Stanislav Kikot, Dmitry Tsarkov, Michael Zakharyaschev, and Evgeny Zolin. Query answering via modal definability with FaCT++: First blood. In *Proc. DL*, volume 1014 of *CEUR Workshop Proceedings*, pages 328–340. CEUR-WS.org, 2013.

[Lutz and Wolter, 2017] Carsten Lutz and Frank Wolter. The data complexity of description logic ontologies. *Logical Methods in Computer Science*, 13(4), 2017.

[Lutz *et al.*, 2009] Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic $\mathcal{EL}$ using a relational database system. In *Proc. IJCAI*, pages 2070–2075, 2009.

[Lutz, 2008] Carsten Lutz. The complexity of conjunctive query answering in expressive description logics. In *Proc. of IJCAR*, volume 5195 of *LNCS*, pages 179–193. Springer, 2008.

[Madelaine and Stewart, 2007] Florent R. Madelaine and Iain A. Stewart. Constraint satisfaction, logic and forbidden patterns. *SIAM J. Comput.*, 37(1):132–163, 2007.

[Pérez-Urbina *et al.*, 2010] Héctor Pérez-Urbina, Boris Motik, and Ian Horrocks. Tractable query answering and rewriting under description logic constraints. *JAL*, 8(2):186–209, 2010.

[Schild, 1991] Klaus Schild. A correspondence theory for terminological logics: Preliminary report. In *Proc. IJCAI*, pages 466–471, 1991.

[Sirin *et al.*, 2007] Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: A practical OWL-DL reasoner. *J. Web Sem.*, 5(2):51 – 53, 2007.

[Trivela *et al.*, 2015] Despoina Trivela, Giorgos Stoilos, Alexandros Chortaras, and Giorgos B. Stamou. Optimising resolution-based rewriting algorithms for OWL ontologies. *J. Web Sem.*, 33:30–49, 2015.

[Zhou *et al.*, 2015] Yujiao Zhou, Bernardo Cuenca Grau, Yavor Nenov, Mark Kaminski, and Ian Horrocks. Pagoda: Pay-as-you-go ontology query answering using a datalog reasoner. *J. Artif. Intell. Res.*, 54:309–367, 2015.

[Zolin, 2007] Evgeny Zolin. Modal logic applied to query answering and the case for variable modalities. In *Proc. of DL*, volume 250 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2007.

Appendices

A Some Technical Preliminaries

Every interpretation $\mathcal{I}$ can be viewed as an undirected graph $G^u_\mathcal{I}$, analogously to the definition of the undirected graph G^u_q of a CQ q. The universal role does not give rise to edges in $G^u_\mathcal{I}$.

An interpretation is *tree-shaped* or a *tree interpretation* if $G^u_\mathcal{I}$ is a tree and there are no multi-edges, that is, $(d, e) \in r^\mathcal{I}$ implies $(d, e) \notin s^\mathcal{I}$ for all (potentially inverse) roles $s \neq r$. Let $\mathcal{T}$ be an $\mathcal{ALCHI}^u$-TBox and $\mathcal{A}$ an ABox. An interpretation $\mathcal{I}$ is a *forest model of* $\mathcal{A}$ if there are tree interpretations $(\mathcal{I}_a)_{a\in\mathsf{ind}(\mathcal{A})\cup\mathcal{D}}$, where $\mathcal{D}$ is a (potentially empty) set of individuals, with mutually disjoint domains, and

$$\Delta^{\mathcal{I}_a} \cap \mathsf{ind}(\mathcal{A}) = \begin{cases} \{a\}, & \text{if } a \in \mathsf{ind}(\mathcal{A}) \\ \emptyset, & \text{if } a \in \mathcal{D}, \end{cases}$$

such that $\mathcal{I}$ is the (non-disjoint) union of $\mathcal{I}_\mathcal{A}$ and $(\mathcal{I}_a)_{a\in\mathsf{ind}(\mathcal{A})\cup\mathcal{D}}$ where $\mathcal{I}_\mathcal{A}$ is $\mathcal{A}$ viewed as an interpretation. An *extended forest model* $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}$ is a model of $\mathcal{A}$ and $\mathcal{T}$ that can be obtained from a forest model $\mathcal{J}$ of $\mathcal{A}$ by closing under role inclusions from $\mathcal{T}$, that is, adding (d, e) to $r^\mathcal{I}$ when $(d, e) \in s^\mathcal{J}$ and $\mathcal{T} \models s \sqsubseteq r \in \mathcal{T}$. We also say that $\mathcal{J}$ *underlies* $\mathcal{I}$.

Lemmas of the following kind have been widely used in the literature on ontology-mediated querying. The proof of the "if" direction uses a standard unraveling argument and is omitted, see for example [Lutz, 2008].

Lemma 16. *Let $Q = (\mathcal{T}, \Sigma, q(\mathbf{x})$ be an OMQ from $(\mathcal{ALCHI}^u, UCQ)$, $\mathcal{A}$ a Σ-ABox, and $\mathbf{a} \subseteq \mathsf{ind}(\mathcal{A})$. Then $\mathcal{A} \models Q(\mathbf{a})$ iff for all extended forest models $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}$, $\mathcal{I} \models Q(\mathbf{a})$.*

We introduce some more helping lemmas. An ABox $\mathcal{A}$ can be seen as a directed graph $G_\mathcal{A}$ and as an undirected graph $G^u_\mathcal{A}$ in the expected way, analogously to the definition of G_q and G^u_q for a CQ q. For an ABox $\mathcal{A}$ and $a \in \mathsf{ind}(\mathcal{A})$, we use $\mathcal{A}^{\mathsf{con}}_a$ to denote the restriction of $\mathcal{A}$ to the individuals reachable in $G^u_\mathcal{A}$ from a. We also denote with $\mathsf{CON}_\mathcal{A}$ the set of ABoxes induced by the maximal connected components of $G^u_\mathcal{A}$.

Lemma 17. *Let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCHI}, IQ)$. Then $\mathcal{A} \models Q(a)$ implies $\mathcal{A}^{\mathsf{con}}_a \models Q(a)$.*

A *homomorphism* from an ABox $\mathcal{A}$ to an ABox $\mathcal{B}$ is a function $h : \mathsf{ind}(\mathcal{A}) \to \mathsf{ind}(\mathcal{B})$ such that $A(a) \in \mathcal{A}$ implies $A(h(a)) \in \mathcal{B}$ and $r(a, b) \in \mathcal{A}$ implies $r(h(a), h(b)) \in \mathcal{B}$. We write $\mathcal{A} \to \mathcal{B}$ to indicate that there is a homomorphism from $\mathcal{A}$ to $\mathcal{B}$. For $a \in \mathsf{ind}(\mathcal{A})$ and $b \in \mathsf{ind}(\mathcal{B})$, we further write $(\mathcal{A}, a) \to (\mathcal{B}, b)$ to indicate that there is a homomorphism h from $\mathcal{A}$ to $\mathcal{B}$ with $h(a) = b$. The following lemma is well-known, see for example [Bienvenu *et al.*, 2014].

Lemma 18. *Let $Q = (\mathcal{T}, \Sigma, q)$ be a unary OMQ from $(\mathcal{ALCHI}^u, \mathcal{Q})$, with $\mathcal{Q} \in \{UCQ, IQ\}$, $\mathcal{A}$ and $\mathcal{B}$ be Σ-ABoxes, $a \in \mathsf{ind}(\mathcal{A})$, and $b \in \mathsf{ind}(\mathcal{B})$. Then $(\mathcal{A}, a) \to (\mathcal{B}, b)$ and $\mathcal{A} \models Q(a)$ implies $\mathcal{B} \models Q(b)$.*

B Proofs for Section 3

We start with introducing several lemmas concerned with certain constructions on ABoxes. These lemmas are closely related to the connection between ontology-mediated querying and constraint satisfaction problems (CSPs), see for example [Bienvenu *et al.*, 2014; Lutz and Wolter, 2017].

Note that in assertions $r(x, y)$ in an ABox, r must be a role name but cannot be an inverse role. For purposes of uniformity, we use $r^-(x, y)$ as an alternative notation to denote an assertion $r(y, x)$ in an ABox. A *cycle* in an ABox is defined exactly like a cycle in a CQ, repeated here for convenience. A *cycle* in an ABox $\mathcal{A}$ is a sequence of non-identical assertions $r_0(a_0, a_1), \dots, r_{n-1}(a_{n-1}, a_n)$ in $\mathcal{A}$, $n \geq 1$, where

1. $r_0, \dots, r_{n-1}$ are (potentially inverse) roles,
2. $a_i \neq a_j$ for $0 \leq i < j < n$, and $a_0 = a_n$.

The *length* of this cycle is n. The *girth* of $\mathcal{A}$ is the length of the shortest cycle in it and ∞ if $\mathcal{A}$ has no cycle.

The following is a DL formulation of what is often known as the sparse incomparability lemma in CSP [Feder and Vardi, 1998].

Lemma 19. *For every ABox $\mathcal{A}$ and all $g, s \geq 0$, there is an ABox $\mathcal{A}^g$ of girth exceeding g such that*

1. *$\mathcal{A}^g \to \mathcal{A}$ and*
2. *for every ABox $\mathcal{B}$ with $|\mathsf{ind}(\mathcal{B})| \leq s$, $\mathcal{A} \to \mathcal{B}$ iff $\mathcal{A}^g \to \mathcal{B}$.*

We next establish a 'pointed' version of Lemma 19 that is crucial for the subsequent proofs. The *a-girth of* $\mathcal{A}$ is defined exactly like the girth except that we only consider cycles that *do no pass through* a.

Lemma 20. *For all ABoxes $\mathcal{A}$, $a \in \mathsf{ind}(\mathcal{A})$, and $g, s \geq 0$, there is an ABox $\mathcal{A}^g$ of a-girth exceeding g such that*

1. *$(\mathcal{A}^g, a) \to (\mathcal{A}, a)$*
2. *for every ABox $\mathcal{B}$ with $|\mathsf{ind}(\mathcal{B})| \leq s$ and every $b \in \mathsf{ind}(\mathcal{B})$, $(\mathcal{A}, a) \to (\mathcal{B}, b)$ iff $(\mathcal{A}^g, a) \to (\mathcal{B}, b)$.*

Proof. Let $\mathcal{A}$ be an ABox, $a \in \mathsf{ind}(\mathcal{A})$, and $g, s \geq 0$. Further, let $\mathcal{A}_+$ be the ABox obtained from $\mathcal{A}$ by adding the assertion $P(a)$, P a fresh concept name, let $\mathcal{A}^g_+$ the ABox obtained from $\mathcal{A}_+$ by applying Lemma 19 for g and s, and let h be a homomorphism from $\mathcal{A}^g_+$ to $\mathcal{A}_+$. Assume w.l.o.g. that the individual name a does not occur in $\mathcal{A}^g_+$. We use $\mathcal{A}^g$ to denote the ABox obtained from $\mathcal{A}^g_+$ by dropping all facts of the form $P(b)$ and identifying all individual names b with $h(b) = a$, replacing them with a. We show that $\mathcal{A}^g$ is as required:

(a) $\mathcal{A}^g$ has a-girth higher than g.

Every cycle in $\mathcal{A}^g$ that does not pass through a is also in $\mathcal{A}^g_+$, thus is of length exceeding g.

(b) Point 1 of Lemma 20 is satisfied.

Let $h' : \mathsf{ind}(\mathcal{A}^g) \to \mathsf{ind}(\mathcal{A})$ be such that $h'(a) = a$ and $h'(b) = h(b)$ if $a \neq b$. It can be verified that h' is a homomorphism from $\mathcal{A}^g$ to $\mathcal{A}$. It clearly witnesses $(\mathcal{A}^g, a) \to (\mathcal{A}, a)$, as required.

(c) Point 2 of Lemma 20 is satisfied.

Let $\mathcal{B}$ be an ABox with $|\mathsf{ind}(\mathcal{B})| \leq s$ and $b \in \mathsf{ind}(\mathcal{B})$. We have to show that $(\mathcal{A}, a) \to (\mathcal{B}, b)$ iff $(\mathcal{A}^g, a) \to (\mathcal{B}, b)$.

The "only if" direction is immediate by (b). For the "if" direction, assume that $(\mathcal{A}^g, a) \to (\mathcal{B}, b)$. Then $\mathcal{A}^g \cup \{P(a)\} \to \mathcal{B} \cup \{P(b)\}$. This implies $\mathcal{A}^g_+ \to \mathcal{B} \cup \{P(b)\}$ and by Point 2 of Lemma 19 also $\mathcal{A}_+ \to \mathcal{B} \cup \{P(b)\}$. As $P(a)$ is the only assertion of this form in $\mathcal{A}_+$, it follows that $(\mathcal{A}_+, a) \to (\mathcal{B} \cup \{P(b)\}, b)$, thus $(\mathcal{A}, a) \to (\mathcal{B}, b)$. ❑

The following lemma is a straightforward variation of similar lemmas from [Bienvenu *et al.*, 2014]. The constructed ABoxes are called *CSP templates* in [Bienvenu *et al.*, 2014].

Lemma 21. *Let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCHI}^u$, IQ). Then one can find a set Γ of pairs $(\mathcal{B}, b)$ with $\mathcal{B}$ a Σ-ABox and $b \in \mathsf{ind}(\mathcal{B})$ such that for every Σ-ABox $\mathcal{A}$ and all $a \in \mathsf{ind}(\mathcal{A})$,*

1. *$\mathcal{A} \models Q(a)$ iff $(\mathcal{A}, a) \not\to (\mathcal{B}, b)$ for all $(\mathcal{B}, b) \in \Gamma$;*
2. *$\mathcal{A}$ is consistent with $\mathcal{T}$ iff $\mathcal{A} \to \mathcal{B}$ for some $(\mathcal{B}, b) \in \Gamma$.*

When Q is from $(\mathcal{ALCHI}$, IQ), then Γ can be chosen so that all ABoxes in it are identical.

An $\mathcal{ELI}^u$*-concept* is an $\mathcal{ALCI}$-concept that uses only the constructors $\sqcap$, $\exists r.C$, $\exists r^-.C$, and $\exists u.C$ where u is the universal role.

Lemma 4. *Let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCHI}^u$, UCQ). Then*

1. *if $q(x)$ is x-acyclic and connected, then Q is rewritable into an OMQ $(\mathcal{T}, \Sigma, C(x))$ with $C(x)$ an $\mathcal{ALCI}$-IQ and*
2. *if $q(x)$ is x-acyclic, then Q is rewritable into an OMQ $(\mathcal{T}, \Sigma, C(x))$ with $C(x)$ an $\mathcal{ALCI}^u$-IQ.*

The size of the IQs $C(x)$ is polynomial in the size of $q(x)$.

Proof. For Point 1, let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCHI}^u$, UCQ) where $q(x)$ is x-acyclic and connected. Further, let $Q' = (\mathcal{T}, \Sigma, (P \to \bigsqcup_{p(x) \text{ a CQ in } q(x)} C_p)(x))$ be as constructed in the main part of the paper. We have to show the following:

- "$Q \subseteq Q'$": Let $\mathcal{A}$ be a Σ-ABox with $\mathcal{A} \models Q(a)$. Then, for every model $\mathcal{I}$ of $\mathcal{T}$ and $\mathcal{A}$, $\mathcal{I} \models p(a)$, for some CQ $p(x)$ in $q(x)$, and thus $\mathcal{I} \models \neg P(a)$ or $\mathcal{I} \models P(a) \wedge p(a)$. As $\mathcal{A}_{p''} \to \mathcal{A}_p \cup \{P(x)\}$, the latter is the same as $\mathcal{I} \models p''(a)$ or $\mathcal{I} \models C_p(a)$. Thus $\mathcal{I} \models (P \to C_p)(a)$, or $\mathcal{I} \models Q'(a)$.
- "$Q' \subseteq Q$": Let $\mathcal{A}$ be a Σ-ABox with $\mathcal{A} \models Q'(a)$. Then, for every model $\mathcal{I} = (\Delta, \cdot^\mathcal{I})$ of $\mathcal{T}$ and $\mathcal{A}$, $\mathcal{I} \models (P \to \bigsqcup_{p(x) \text{ a CQ in } q(x)} C_p)(a)$. Then, there must be a model $\mathcal{I}' = (\Delta, \cdot^{\mathcal{I}'})$ of $\mathcal{T}$ and $\mathcal{A}$ (possibly the same as $\mathcal{I}$) such that $P^{\mathcal{I}'} = \{a^{\mathcal{I}'}\}$ and $\cdot^\mathcal{I}$ and $\cdot^{\mathcal{I}'}$ coincide on all other symbols. We have $\mathcal{I}' \models (P \to \bigsqcup_{p(x) \text{ a CQ in } q(x)} C_p)(a)$, thus $\mathcal{I}' \models C_p(a)$, for some CQ $p(x)$ in $q(x)$. From the construction of C_p: $\mathcal{I}' \models p''(a)$, and from the fact that P is interpreted as a singleton in $\mathcal{I}'$, $\mathcal{I}' \models p(a)$, or $\mathcal{I}' \models \bigvee_{p(x) \text{ a CQ in } q(x)} p(a)$. As P is fresh, and in particular does not occur in q, and $\mathcal{I}$ and $\mathcal{I}'$ might differ only w.r.t. the interpretation of P: $\mathcal{I} \models \bigvee_{p(x) \text{ a CQ in } q(x)} p(a)$. Thus, $\mathcal{A} \models Q(a)$.

For Point 2, let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCHI}^u$, UCQ) where $q(x)$ is x-acyclic. Let $p_0(x), p_1(), \dots, p_m()$ be the maximal connected components of $q(x)$. Note that $p_0(x)$ is x-acyclic and each $p_i()$ is acyclic in the sense that it contains no cycles at all. We can view $p_0(x)$ as an $\mathcal{ELI}$-concept and each $p_i()$ as an $\mathcal{ELI}^u$-concept C_{p_i} of the form $\exists u.C$ with u the universal role and C an $\mathcal{ELI}$-concept. Let $C_p = C_{p,0} \sqcap C_{p,1} \sqcap \ldots \sqcap C_{p,m}$ and $Q' = (\mathcal{T}, \Sigma, (P \to \bigsqcup_{p(x) \text{ a CQ in } q(x)} C_p)(x))$. One can show that $Q \equiv Q'$. ❑

Example 22. Let Q be an OMQ $(\mathcal{T}, \Sigma, q(x))$ with $\mathcal{T} = \emptyset$, $\Sigma = \{r, s, t, v\}$, and $q(x) = \exists y_1 \exists y_2 \exists y_3\, r(x, y_1) \wedge s(x, y_2) \wedge t(y_2, y_1) \wedge v(y_2, y_3)$. It is easy to see that $q(x)$ is x-acyclic and connected. Towards obtaining an $\mathcal{ALCI}$-IQ rewriting, we construct a tree-shaped CQ $p''(x)$ from $q(x)$ by first removing the atom $s^-(y_2, x)$ and then adding atoms $s^-(y_2, x')$ and $P(x')$, with x' a fresh variable: $p''(x) = \exists y_1 \exists y_2 \exists y_3 \exists x'\, P(x') \wedge r(x, y_1) \wedge s^-(y_2, x') \wedge t(y_2, y_1) \wedge v(y_2, y_3)$. The concept C_q corresponding to $p''(x)$ is $\exists r.\exists t^-.(\exists v.\top \sqcap \exists s^-.P)$, and thus the desired rewriting is the OMQ $Q' = (\mathcal{T}, \Sigma, q'(x))$ with $q'(x)$ the $\mathcal{ALCI}$-IQ: $(P \to C_q)(x)$.

Theorem 3. *Let $\mathcal{L} \in \{\mathcal{ALCI}, \mathcal{ALCHI}\}$ and let $Q = (\mathcal{T}, \Sigma, q(x))$ be a unary OMQ from $(\mathcal{L}$, UCQ) that is non-empty. Then the following are equivalent:*

1. *Q is IQ-rewritable, that is, it is rewritable into an OMQ $Q' = (\mathcal{T}', \Sigma, C(x))$ from $(\mathcal{L}$, IQ);*
2. *Q is rewritable into an OMQ $Q' = (\mathcal{T}, \Sigma, C(x))$ from $(\mathcal{L}$, IQ);*
3. *$Q \equiv (\mathcal{T}, \Sigma, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$.*

When $\mathcal{L}$ is replaced with $\mathcal{L}^u$, then the same equivalences hold except that $q^{\mathsf{con}}_{\mathsf{acyc}}$ is replaced with q_{acyc}.

Proof. The implication "2 $\Rightarrow$ 1" is trivial. For "3 $\Rightarrow$ 2", assume that $Q \equiv (\mathcal{T}, \Sigma, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$. Since $q^{\mathsf{con}}_{\mathsf{acyc}}$ is connected and x-acyclic, we can apply Lemma 4.

For "1 $\Rightarrow$ 3", we show that whenever an OMQ Q from $(\mathcal{L}$, UCQ) is IQ-rewritable, then (a) $Q \equiv Q_{\mathsf{acyc}}$ where $Q_{\mathsf{acyc}} = (\mathcal{T}, \Sigma, q_{\mathsf{acyc}}(x))$ and (b) $Q \equiv Q^{\mathsf{con}}$ where $Q^{\mathsf{con}} := (\mathcal{T}, \Sigma, q^{\mathsf{con}}(x))$. This yields $Q \equiv (\mathcal{T}, \Sigma, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$ as desired: if Q is IQ-rewritable, then (a) yields $Q \equiv Q_{\mathsf{acyc}}$, thus Q_{acyc} is IQ-rewritable and we can apply (b).

Thus, let Q from $(\mathcal{L}$, UCQ) be IQ-rewritable. Thus there is an OMQ $Q' = (\mathcal{T}', \Sigma, C(x))$ from $(\mathcal{L}$, IQ) that is equivalent to Q. By Lemma 21, one can find a Σ-ABox $\mathcal{B}$ and $b_1, \dots, b_k \in \mathsf{ind}(\mathcal{B})$ such that for every Σ-ABox $\mathcal{A}$ and $a \in \mathsf{ind}(\mathcal{A})$,

1. $\mathcal{A} \models Q'(a)$ iff $(\mathcal{A}, a) \not\to (\mathcal{B}, b_i)$ for $1 \le i \le k$;
2. $\mathcal{A}$ is consistent with $\mathcal{T} \cup \mathcal{T}'$ iff $\mathcal{A} \to \mathcal{B}$.

We show Points (a) and (b) from above.

(a) We have $Q_{\mathsf{acyc}} \subseteq Q$ by definition of Q_{acyc}, no matter whether Q is IQ-rewritable or not, and thus it remains to show that $Q \subseteq Q_{\mathsf{acyc}}$. If all CQs in $q(x)$ are x-acyclic, the result clearly holds. In the following we assume that at least one CQ in $q(x)$ is not x-acyclic.

Let $\mathcal{A}$ be a Σ-ABox with $\mathcal{A} \models Q(a)$. Thus $(\mathcal{A}, a) \not\to (\mathcal{B}, b_i)$ for $1 \leq i \leq k$. We apply Lemma 20 with g the maximum between 2 and the girths of CQs from $q(x)$ which are not x-acyclic, obtaining a Σ-ABox $\mathcal{A}^g$ of a-girth exceeding g such that $(\mathcal{A}^g, a) \to (\mathcal{A}, a)$ and $(\mathcal{A}^g, a) \not\to (\mathcal{B}, b_i)$ for $1 \leq i \leq k$. The latter yields $\mathcal{A}^g \models Q(a)$. We aim to show that $\mathcal{A}^g \models Q_{\mathsf{acyc}}(a)$. Since $(\mathcal{A}^g, a) \to (\mathcal{A}, a)$, it follows by Lemma 18 that $\mathcal{A} \models Q_{\mathsf{acyc}}(a)$, as desired.

By Lemma 16, it suffices to show that for every extended forest model $\mathcal{I}$ of $\mathcal{A}^g$ and $\mathcal{T}$, we have $\mathcal{I} \models q_{\mathsf{acyc}}(a)$. Thus let $\mathcal{I}$ be such a model. Since $\mathcal{A}^g \models Q(a)$, we have $\mathcal{I} \models q(a)$ and thus there is a CQ $p(x)$ in $q(x)$ such that $\mathcal{I} \models p(a)$. Consequently, there is a homomorphism h from $p(x)$ to $\mathcal{I}$ with $h(x) = a$. Let $p'(x)$ be the contraction of $p(x)$ obtained by identifying all variables y_1 and y_2 such that $h(y_1) = h(y_2)$. As witnessed by h, $\mathcal{I} \models p'(a)$. Note that the x-girth of $p'(x)$ is either ∞ or it is bounded from above by g since the x-girth of $p(x)$ is. Also note that h is an injective homomorphism from $p'(x)$ to $\mathcal{I}$. By definition of extended forest models, all cycles in $\mathcal{I}$ are either cyles from $\mathcal{A}^g$, or they are cycles of the form $r(y,z), s(z,y)$. This together with the fact that the girth of $\mathcal{A}^g$ exceeds g implies that every cycle in $p'(x)$ passes through x or is of the latter kind. In fact, $p'(x)$ is x-acyclic when $\mathcal{T}$ contains no role inclusions since then $\mathcal{I}$ is a forest model of $\mathcal{A}^g$. Since $p'(x)$ is a CQ in $q_{\mathsf{acyc}}(x)$, we are done in that case.

Now for the case where $\mathcal{T}$ contains role inclusions. Let $\mathcal{J}$ be the forest model of $\mathcal{A}^g$ underlying $\mathcal{I}$. Construct a CQ $p''(x)$ from $p'(x)$ as follows: for all distinct variables y, z, with $y \neq x$ and $z \neq x$, whenever $r_1(y,z), \ldots, r_k(y,z)$, $s_1(z,y), \ldots, s_\ell(z,y)$ are all atoms of this form in $p'(x)$, then replace them with $r(x,y)$ if $(h(x), h(y)) \in r^{\mathcal{J}}$ and with $r(y,x)$ if $(h(y), h(x)) \in r^{\mathcal{J}}$. Note that by definition of extended forest models and due to the fact that g, the girth of $\mathcal{A}^g$, is greater than 2, such an r always exists. As witnessed by h, $\mathcal{I} \models p''(a)$. Moreover, $p''(x)$ is x-acyclic and a CQ in $q_{\mathsf{acyc}}(x)$, thus we are again done.

(b) It is immediate by definition of Q^{con} that $Q \subseteq Q^{\mathsf{con}}$. We thus have to show that $Q^{\mathsf{con}} \subseteq Q$. Assume the contrary. Then, there is a Σ-ABox $\mathcal{A}$ and an $a \in \mathsf{ind}(\mathcal{A})$ such that $\mathcal{A} \models Q^{\mathsf{con}}(a)$ and $\mathcal{A} \not\models Q(a)$. Note that $\mathcal{A}$ must be consistent with $\mathcal{T}$. Since Q is non-empty, there is a Σ-ABox $\mathcal{A}_Q$ such that $\mathcal{A}_Q \models Q(b)$ for some $b \in \mathsf{ind}(\mathcal{A}_Q)$. Let $\mathcal{A}'$ be the disjoint union of $\mathcal{A}$ and $\mathcal{A}_Q$. We get $\mathcal{A}' \models Q(a)$ from Lemma 18 and thus $\mathcal{A}' \models Q'(a)$. Since Q' is from $(\mathcal{ALCI}, \text{IQ})$ and ${\mathcal{A}'}^{\mathsf{con}}_a = \mathcal{A}$, the latter and Lemma 17 implies $\mathcal{A} \models Q'(a)$, thus $\mathcal{A} \models Q(a)$, a contradiction.

When the OMQ language $\mathcal{L}$ is replaced by $\mathcal{L}^u$, we can show that $Q \equiv Q_{\mathsf{acyc}}$ exactly as above. The second part of the proof showing that $Q \equiv Q^{\mathsf{con}}$ (does not go through and) is no longer needed. ❑

Lemma 23. *Let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCH}, UCQ)$ such that $q(x)$ is x-acyclic and connected. Then Q is rewritable into an OMQ $(\mathcal{T} \cup \mathcal{T}', \Sigma, q(x))$ from $(\mathcal{ALCH}, IQ)$ whose size is polynomial in the size of Q.*

Proof. Let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCH}, \text{UCQ})$ such that $q(x)$ is x-acyclic and connected. From Lemma 4, we know that there is an OMQ $Q' = (\mathcal{T}, \Sigma, C(x))$ that is equivalent to Q, with $C(x)$ an $\mathcal{ALCI}$-IQ. From the proof of the lemma, we further know that C has the form $P \to \bigsqcup_{p(x) \text{ a CQ in } q(x)} C_p$ where each C_p is an $\mathcal{ELI}$-concept. We show how to transform Q' into an equivalent OMQ $(\mathcal{T}', \Sigma, C'(x))$ from $(\mathcal{ALCH}, \text{IQ})$.

We start with setting $\mathcal{T}' := \mathcal{T}$ and $C' := C$ and apply the following modification step until no further changes are possible: if D is a subconcept of C' that is of the form $\exists r^-.E$ with E an $\mathcal{EL}$-concept, then let P_D be a fresh concept name that is not in Σ and

- set $\mathcal{T}' = \mathcal{T}' \cup \{E \sqsubseteq \forall r.P_D\}$ and
- replace $\exists r^-.E$ in C' with P_D.

At the end of the transformation, C' will contain no inverse roles anymore, so the constructed OMQ is from $(\mathcal{ALCH}, \text{IQ})$. Moreover, it is straightforward to show that the described modification step preserves equivalence of the OMQ.

In fact, assume that $Q_2 = (\mathcal{T}_2, \Sigma, C_2(x))$ was obtained by a single modification step from $Q_1 = (\mathcal{T}_1, \Sigma, C_1(x))$. Let $\mathcal{A}$ be a Σ-ABox and $a \in \mathsf{Ind}(\mathcal{A})$. First assume that $\mathcal{A} \not\models Q_1(a)$. Then there is a model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}_1$ with $a \notin C_1^{\mathcal{I}}$. Extend $\mathcal{I}$ to the concept name P_D by setting $P_D^{\mathcal{I}} = (\exists r^-.E)^{\mathcal{I}}$. Clearly, $\mathcal{I}$ is then a model of $\mathcal{T}_2$. Moreover, by construction of C_2 we have $a \notin C_2^{\mathcal{I}}$. Conversely, assume that $\mathcal{A} \not\models Q_1(a)$. Then there is a model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}_2$ with $a \notin C_2^{\mathcal{I}}$. Clearly, $\mathcal{I}$ is a model of $\mathcal{T}_1$. Since $\mathcal{I}$ is a model of $\mathcal{T}_2$, we have $(\exists r^-.E)^{\mathcal{I}} \subseteq P_D^{\mathcal{I}}$. We can modify $\mathcal{I}$ by setting $P_D^{\mathcal{I}} = (\exists r^-.E)^{\mathcal{I}}$ and the resulting $\mathcal{I}$ will still be a model of $\mathcal{T}_1$ and still satisfy $a \notin C_2^{\mathcal{I}}$ since all occurrences of P_D in C_2 are positive. Moreover, by construction of C_2 it also satisfies $a \notin C_1^{\mathcal{I}}$. ❑

The proof of the following lemma is a much simplified and slightly extended version of a construction from [Kikot and Zolin, 2013].

Lemma 24.

1. *Every OMQ $Q = (\mathcal{T}, \Sigma, q(x))$ from $(\mathcal{ALCH}, UeCQ)$ with $q(x)$ x-acyclic and x-accessible is rewritable into an OMQ $Q = (\mathcal{T}, \Sigma, C(x))$ with $C(x)$ an $\mathcal{ALC}$-IQ and*
2. *Every OMQ $Q = (\mathcal{T}, \Sigma, q(x))$ from $(\mathcal{ALCH}^u, UeCQ)$ with $q(x)$ x-acyclic is rewritable into an OMQ $Q = (\mathcal{T}, \Sigma, C(x))$ with $C(x)$ an $\mathcal{ALC}^u$-IQ.*

The size of the IQs $C(x)$ is polynomial in the size of $q(x)$.

Proof. We first observe that Lemma 4 extends to the case where the actual query is a UeCQ rather than a UCQ. One simply "carries through" atoms $C(x)$ with C a compound concept in the construction of the IQ.

We start with Point 2 since its proof is simpler and prepares for the proof of Point 1. Thus, let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCH}^u, \text{UeCQ})$ with $q(x)$ x-acyclic. By (the extended) Lemma 4, there is an equivalent OMQ $Q' = (\mathcal{T}, \Sigma, C(x))$ with $C(x)$ an $\mathcal{ALCI}^u$-IQ. In fact, the IQ $C(x)$ constructed in the proof of Lemma 4 is of the form $P \to \bigsqcup_{p(x) \text{ a CQ in } q(x)} C_p(x)$ where each C_p is an $\mathcal{ELI}^u$-*concept decorated with $\mathcal{ALC}^u$-concepts*, that is, built according to the syntax rule

$$C, D ::= \top \mid A \mid C \sqcap D \mid \exists r.D \mid \exists u.D \mid E$$

where A ranges over all concept names, r over all (potentially inverse) roles, and E over all $\mathcal{ALC}^u$-concepts. Note that every $\mathcal{ELI}^u$-concept decorated with $\mathcal{ALC}^u$-concepts is an $\mathcal{ALCI}^u$-concept, but that the converse is false.

We construct from Q' an $(\mathcal{ALCH}^u, \text{IQ})$-rewriting $(\mathcal{T}, \Sigma, C'(x))$ of Q where C' has the form $C_{\mathsf{pre}} \rightarrow C_{\mathsf{con}}$. To start, let $D_1 = \exists r_1^-.P, \dots, D_\ell = \exists r_\ell^-.P$ be all subconcepts of C that are of this form and let

- C_{con} be obtained from C by replacing each concept D_i with a fresh concept name $P_{D_i} \notin \Sigma$ and
- $C_{\mathsf{pre}} = \forall r_1.P_{D_1} \sqcap \cdots \sqcap \forall r_\ell.P_{D_\ell}$.

Next, exhaustively apply the following transformation step: if $D = \exists r^-.E$ is a subconcept of C_{con} where E is an $\mathcal{ALC}^u$-concept (that is, does not contain any inverse roles), then

- replace D in C_{con} with a fresh concept name $P_D \notin \Sigma$ and
- set $C_{\mathsf{pre}} = C_{\mathsf{pre}} \sqcap \forall u.(E \rightarrow \forall r.P_D)$.

We end up with C_{con} being an $\mathcal{ALC}^u$-concept because if there is a subconcept $\exists r^-.E$ of C_{con} left, then in the innermost such subconcept E must be an $\mathcal{ALC}^u$-concept and thus the transformation rule applies. It can be proved that the initial IQ $C_{\mathsf{pre}} \rightarrow C_{\mathsf{con}}(x)$ is equivalent to $C(x)$ and that the transformation step is equivalence preserving. We omit details, please see the proof of Lemma 23 for very similar arguments.

We now turn to Point 1. Let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{ALCH}, \text{UeCQ})$ with $q(x)$ x-acyclic and x-accessible. Then $q(x)$ is also connected. By (the extended) Lemma 4, there is an equivalent OMQ $Q' = (\mathcal{T}, \Sigma, C(x))$ with $C(x)$ an $\mathcal{ALCI}$-IQ. In fact, the IQ $C(x)$ constructed in the proof of Lemma 4 is of the form $P \rightarrow \bigsqcup_{p(x) \text{ a CQ in } q(x)} C_p(x)$ where each C_p is an *$\mathcal{ELI}$-concept decorated with $\mathcal{ALC}$-concepts*, that is, an $\mathcal{ELI}^u$-concept decorated with $\mathcal{ALC}^u$-concepts that does not mention the universal role. However, the syntactic structure of C is even more restricted.

Claim. *In each subconcept $\exists r^-.D$ of C, $D = P$ or D has the form $D_0 \sqcap \exists r_1^-.(D_1 \sqcap \exists r_2^-.(\dots \sqcap \exists r_n^-.P))\dots)$, $n \geq 1$.*

Proof of claim. Let $p(x)$ be a CQ in $q(x)$. Recall that, when constructing $C(x)$ in the proof of Lemma 4, we first remove atoms of the form $r(x, y)$ from $p(x)$ to obtain a tree-shaped CQ $p'(x)$, then add back $r^-(y, u)$ and $P(u)$ for each removed $r(x, y)$ where u is a fresh variable producing a CQ $p''(x)$, and finally view $p''(x)$ as an $\mathcal{ELI}$-concept C_p decorated with $\mathcal{ALC}$-concepts.[4]

Let $\exists r^-.D$ be a subconcept of C_p. Then there is a variable y in $p''(x)$ and an atom $r^-(y, z)$ such that D describes the subtree of $p''(x)$ rooted at z and z is a successor of z in the tree-shaped $p''(x)$, that is, y is on the path from the root x of $p''(x)$ to z. First assume that $r^-(y, z)$ was one of the atoms added back in the construction of $p''(x)$. Then $D = P$ and we are done. Now assume that $r^-(y, z)$ was already in $p'(x)$ and thus in $p(x)$. Since $p(x)$ is x-accessible, z is reachable from x in the directed graph G_p. Since z is not reachable from x in the directed graph $G_{p''}$ it follows from the construction of $p'(x)$ and $p''(x)$ that z is reachable in $G_{p''}$ from a leaf node labeled with P. Consequently, D must have the stated form. This finishes the proof of the claim.

We construct from Q' an $(\mathcal{ALCH}, \text{IQ})$-rewriting $(\mathcal{T}, \Sigma, C'(x))$ of Q where C' has the form $C_{\mathsf{pre}} \rightarrow C_{\mathsf{con}}$. To start, let $D_1 = \exists r_1^-.P, \dots, D_\ell = \exists r_\ell^-.P$ be all subconcepts of C that are of this form and let

- C_{con} be obtained from C by replacing each concept D_i with a fresh concept name P_{D_i} and
- $C_{\mathsf{pre}} = \forall r_1.P_{D_1} \sqcap \cdots \sqcap \forall r_\ell.P_{D_\ell}$.

It is easy to see that the following condition is satisfied:

($*$) in every subconcept $D = \exists r^-.E$ of C_{con} with E an $\mathcal{ALC}$-concept, E is of the form $F \sqcap P_{D'}$.

Next, exhaustively apply the following transformation step, which preserves ($*$): if $D = \exists r^-.(F \sqcap P_{D'})$ is a subconcept of C_{con} where F is an $\mathcal{ALC}$-concept, then

- replace D in C_{con} with a fresh concept name P_D and
- replace $P_{D'}$ in C_{pre} with $F \rightarrow \forall R.P_D$.

It can be verified that, because of the claim, the transformation step indeed preserves ($*$). It can also be seen that all subconcepts of the form $\exists r^-.E$ will eventually be eliminated. Finally, it can be shown that the initial IQ $C_{\mathsf{pre}} \rightarrow C_{\mathsf{con}}(x)$ is equivalent to $C(x)$ and that the transformation step is equivalence preserving. We omit details. ❑

Example 25. Let Q be the OMQ from Example 22. Towards obtaining an $\mathcal{ALC}$-IQ rewriting, we start with the $\mathcal{ALCI}$-IQ rewriting Q' described in the same example. The only subconcept of the form $\exists r_i^-.P$ in C_q is $D = \exists s^-.P$. We thus introduce a fresh concept name P_D and initialize C_{pre} and C_{con} with $\forall s.P_D$ and $\exists r.\exists t^-.(\exists v.\top \sqcap P_D)$. We next consider concepts of the form $\exists r^-.(F \sqcap P_{D'})$, with F an $\mathcal{ALC}$ concept and $P_{D'}$ previously introduced. The only such concept is $E = \exists t^-.(\exists v.\top \sqcap P_D)$. We replace E in C_{con} with P_E and P_D in C_{pre} with $\neg\exists v.\top \sqcup \forall t.P_E$. At this point both $C_{\mathsf{pre}} = \forall s.(\forall t.(\neg\exists v.\top \sqcup P_E))$ and $C_{\mathsf{con}} = \exists r.P_E$ are $\mathcal{ALC}$ concepts, thus no further transformation is possible (and neither needed): Q can can be rewritten into an OMQ $Q'' = (\mathcal{T}, \Sigma, C'(x))$ with C' the $\mathcal{ALC}$ concept $\forall s.(\forall t.(\neg\exists v.\top \sqcup P_E)) \rightarrow \exists r.P_E$.

Theorem 5. *Let $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCH}\}$ and let $Q = (\mathcal{T}, \Sigma, q(x))$ be a unary OMQ from $(\mathcal{L}, \text{UCQ})$ that is non-empty. Then the following are equivalent:*

1. *Q is rewritable into an OMQ from $(\mathcal{L}, \text{IQ})$;*
2. *Q is rewritable into an OMQ $(\mathcal{T} \cup \mathcal{T}', \Sigma, C(x))$ from $(\mathcal{L}, \text{IQ})$;*
3. *Q is rewritable into an OMQ from $(\mathcal{LI}, \text{IQ})$;*

If $\Sigma = \Sigma_{\mathsf{full}}$, then the following are equivalent:

4. *Q is rewritable into an OMQ $Q' = (\mathcal{T}, \Sigma_{\mathsf{full}}, C(x))$ from $(\mathcal{L}, \text{IQ})$;*
5. *$Q \equiv (\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{deco}}_{\mathsf{acyc}}(x))$.*

[4]The first two steps can together be viewed as an unfolding construction.

If, furthermore, $\mathcal{L}$ is replaced with $\mathcal{L}^u$ and $\mathcal{LI}$ with $\mathcal{LI}^u$, then Conditions 1 to 3 are further equivalent to:

6. *Q is rewritable into an OMQ $Q' = (\mathcal{T}, \Sigma, C(x))$ from $(\mathcal{L}^u, IQ)$.*

Proof. "2 ⇒ 1" and "1 ⇒ 3" are trivial.

"3 ⇒ 2". We know from Theorem 3 that $(\mathcal{LI}, \text{IQ})$-rewritability of Q implies that $q(x)$ is x-acyclic and connected. By Lemma 23, Q is rewritable into an OMQ from $(\mathcal{L}, \text{IQ})$ that is of the desired shape.

"4 ⇒ 5". Let $Q = (\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$ be an OMQ from $(\mathcal{ALCH}, \text{UCQ})$ and assume that Q is rewritable into an OMQ $Q' = (\mathcal{T}, \Sigma_{\mathsf{full}}, C(x))$ with $C(x)$ an $\mathcal{ALC}$-IQ. Let $Q_{\mathsf{acyc}} = (\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$. It is established in the proof of the "1 ⇒ 3" direction of Theorem 3 that, since Q is rewritable into $(\mathcal{ALCHI}, \text{IQ})$, $Q \equiv Q_{\mathsf{acyc}}$. It thus suffices to show that $Q_{\mathsf{acyc}} \equiv Q^{\mathsf{deco}}_{\mathsf{acyc}}$.

Using the definition of $Q^{\mathsf{deco}}_{\mathsf{acyc}}$, it can be shown that $Q_{\mathsf{acyc}} \subseteq Q^{\mathsf{deco}}_{\mathsf{acyc}}$. To establish the converse direction, assume towards a contradiction that there is a Σ_{full}-ABox $\mathcal{A}$ such that $\mathcal{A} \models Q^{\mathsf{deco}}_{\mathsf{acyc}}(a)$ but $\mathcal{A} \not\models Q_{\mathsf{acyc}}(a)$. Then $\mathcal{A} \not\models Q'(a)$. Take a model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}$ such that $\mathcal{I} \not\models C(a)$. We have $\mathcal{I} \models q^{\mathsf{deco}}_{\mathsf{acyc}}(a)$, thus $\mathcal{I} \models p(x)|_{\mathsf{dreach}(p}(a)$ for some $\mathcal{T}$-decoration $p(x)$ of a CQ in $q_{\mathsf{acyc}}(x)$. Let h be a homomorphism from $p(x)|_{\mathsf{dreach}(p)}$ to $\mathcal{I}$ with $h(x) = a$.

To finish the proof, it suffices to show that we can construct from $\mathcal{I}$ a model $\mathcal{I}'$ of $\mathcal{T}$ such that $\mathcal{I}' \not\models C(a)$ and $\mathcal{I}' \models q_{\mathsf{acyc}}(a)$. In fact, we can then take a homomorphism h' from a CQ in $q_{\mathsf{acyc}}(x)$ to $\mathcal{I}'$ with $h'(x) = a$ and let $\mathcal{A}'$ be $\mathcal{I}'$ restricted to the range of h', viewed as an ABox. Clearly, $\mathcal{A}' \models Q_{\mathsf{acyc}}(a)$ since already $\mathcal{A}' \models (\emptyset, \Sigma_{\mathsf{full}}, q_{\mathsf{acyc}}(x))(a)$. Moreover, $\mathcal{I}'$ is a model of $\mathcal{A}'$ and thus $\mathcal{A}' \not\models Q'(a)$, in contradiction to Q' being equivalent to Q_{acyc}.

It thus remains to construct $\mathcal{I}'$. Informally, we do this by adding to $\mathcal{I}$ the part of $p(x)$ that is not reachable from the answer variable along a directed path. By the second condition of $\mathcal{T}$-decorations, there is a model $\mathcal{J}$ of $\mathcal{T}$ and a homomorphism h' from $p(x)$ to $\mathcal{J}$. We can assume that $\mathcal{I}$ and $\mathcal{J}$ have disjoint domains.

Let $\mathcal{I}'$ be the disjoint union of $\mathcal{I}$ and $\mathcal{J}$, extended as follows: for every atom $r(y_1, y_2)$ in $p(x)$ with $y_1 \notin \mathsf{dreach}(p)$ and $y_2 \in \mathsf{dreach}(p)$, add $(h'(y_1), h(y_2))$ to $r^{\mathcal{I}'}$. It can be verified that the map h'' defined by setting $h''(y) = h(y)$ for all $y \in \mathsf{dreach}(p)$ and $h''(y) = h'(y)$ for all variables y in p that are not in $\mathsf{dreach}(p)$ is a homomorphism from $p(x)$ to $\mathcal{I}'$ with $h''(x) = a$. Thus, $\mathcal{I}' \models q_{\mathsf{acyc}}(a)$ as desired. It thus remains to show that $\mathcal{I}'$ is a model of $\mathcal{T}$ and that $\mathcal{I}' \not\models C(a)$. This is a consequence of the following:

(a) for all $\mathcal{ALC}$-concepts C and all $d \in \Delta^{\mathcal{I}}$, $d \in C^{\mathcal{I}}$ iff $d \in C^{\mathcal{I}'}$;

(b) for all subconcepts C of a concept in $\mathcal{T}$ and all $d \in \Delta^{\mathcal{J}}$, $d \in C^{\mathcal{J}}$ iff $d \in C^{\mathcal{I}'}$.

Both points are proved by induction on the structure of C. This is straightforward for (a) since for every element $d \in \Delta^{\mathcal{I}}$, the subinterpretation of $\mathcal{I}$ induced by the set of elements reachable from d in $\mathcal{I}$ by traveling roles in the forwards direction is identical to the corresponding subinterpretation of $\mathcal{I}'$ (and since $\mathcal{ALC}$-concepts do not admit inverse roles). For (b), it is important to observe that if we have added $(h'(y_1), h(y_2))$ to $r^{\mathcal{I}'}$ in the construction of $\mathcal{I}'$, then then $h'(y_1)$ has an r-successor d in $\mathcal{J}$ such that for all subconcepts C of a concept in $\mathcal{T}$, $d \in C^{\mathcal{J}}$ iff $h(y_2) \in C^{\mathcal{I}}$. In fact, this is a consequence of the decoration of every variable in $p(x)$ with such concepts: when choosing $d = h'(y_2)$, the stated condition must be satisfied.

"5 ⇒ 4". We have that $Q \equiv (\mathcal{T}, \Sigma, q'(x))$, where each q'_i is an x-acyclic, accessible eCQ. We can apply Lemma 24 to obtain an $(\mathcal{L}, \text{IQ})$-rewriting.

For the case with the universal role, it is enough to show that 3 ⇒ 6. Again, we can apply Lemma 24. ❑

Theorem 7. *Let $q(x)$ be a UCQ. Then*

1. *$q(x)$ is rewritable into an $\mathcal{ALCI}$-IQ iff there is a subquery $q'(x)$ of $q(x)$ that is x-acyclic, connected, and equivalent to $q(x)$;*
2. *$q(x)$ is rewritable into an $\mathcal{ALC}$-IQ iff there is a subquery $q'(x)$ of $q(x)$ that is x-acyclic, x-accessible, and equivalent to $q(x)$.*

When $\mathcal{L}$-IQs are replaced with $\mathcal{L}^u$-IQs, then the same equivalences hold except that connectedness/x-accessibility is dropped.

Proof. For the statement at Point 1, the "if" direction is a consequence of Lemma 4, while for the statement at Point 2, the same direction is a consequence of Lemma 24 (and similarly for the cases where the universal role is present). We will thus show the "only if" direction in each case. We first show that $\mathcal{L}$-IQ rewritability of a UCQ $q(x)$, for every $\mathcal{L} \in \{\mathcal{ALCI}, \mathcal{ALC}, \mathcal{ALCI}^u, \mathcal{ALC}^u\}$, implies the existence of a subquery $q'(x)$ of $q(x)$ that is x-acyclic and equivalent to $q(x)$.

A *homomorphism minimal CQ* (also *hom-minimal*) is a CQ which does not admit any equivalent strict subquery.

Claim 1. *Let q and q' be two CQs such that $q \equiv q'$ and q' is hom-minimal. Then:*

1. *q' is a subquery of q;*
2. *q' is a contraction of q.*

Proof of claim. Consider any homomorphisms h_1 and h_2 from q' to q and from q to q', respectively. Then h_1 must be injective and h_2 must be surjective (otherwise $h_1 \circ h_2$ is a non-injective homomorphism from q' to itself, and thus q' is not hom-minimal). The existence of h_1 implies that q' is a subquery of q, while the existence of h_2 implies that q' is a contraction of q.

Claim 2. *Let $p(x)$ be a CQ in $q_{\mathsf{acyc}}(x)$. Then, there exists a hom-minimal CQ $p'(x)$ in $q_{\mathsf{acyc}}(x)$ such that $p(x) \equiv p'(x)$.*

Proof of claim. We show that, in fact, every hom-minimal subquery $p'(x)$ of $p(x)$ which is equivalent to $p(x)$ is a CQ in $q_{\mathsf{acyc}}(x)$. From Claim 1, $p'(x) \equiv p(x)$ and $p'(x)$ being hom-minimal, implies that $p'(x)$ is a contraction of $p(x)$. As $p(x)$ is a contraction of a CQ in $q(x)$, it follows that $p'(x)$ is itself a contraction of some CQ in $q(x)$. As $p(x)$ is x-acyclic and

$p'(x)$ is a subquery of $p(x)$, it follows that $p'(x)$ is x-acyclic. Thus, $p'(x)$ is an x-acyclic contraction of some CQ in $q(x)$, or, in other words, $p'(x)$ is a CQ in $q_{\mathsf{acyc}}(x)$.

Assume now $\mathcal{L}$-IQ rewritability of $q(x)$. By inspecting Point (a) in the proof of direction "1 ⇒ 3" of Theorem 3, we observe that $q(x) \equiv q_{\mathsf{acyc}}(x)$.

For didactic purposes, we first consider the case where $q(x)$ is a CQ. Then, there must be a CQ $p(x)$ in the UCQ $q_{\mathsf{acyc}}(x)$ such that $q(x) \equiv p(x)$. From Claim 2, there must be some CQ $p'(x)$ in $q_{\mathsf{acyc}}(x)$ which is hom-minimal and equivalent to $p(x)$, and thus also to $q(x)$. From Claim 1, it follows that $p'(x)$ is a subquery of $q(x)$, and from the fact that $p'(x)$ is a CQ in $q_{\mathsf{acyc}}(x)$, it follows that $p'(x)$ is x-acyclic.

We now consider the case where $q(x)$ is a UCQ. Let $q_1(x), \ldots, q_k(x)$ be the CQs in $q(x)$ that are *minimal* in $q(x)$ in the following sense: for all CQs $p(x)$ in $q(x)$, $q_i(x) \subseteq p(x)$ implies $q_i(x) \equiv p(x)$. Take such a minimal CQ $q_i(x)$. Since $q(x) \equiv q_{\mathsf{acyc}}(x)$, there must be a CQ $p_i(x)$ in $q_{\mathsf{acyc}}(x)$ such that $q_i(x) \subseteq p_i(x)$. By construction of $q_{\mathsf{acyc}}(x)$, $p_i(x)$ must be the contraction of some CQ $\widehat{q}_i(x)$ in $q(x)$ and thus $p_i(x) \subseteq \widehat{q}_i(x)$. We obtain $q_i(x) \subseteq \widehat{q}_i(x)$ and thus $\widehat{q}_i(x) \equiv q_i(x)$ and consequently $q_i(x) \equiv p_i(x)$. From Claim 2, there must be some hom-minimal query $p_i'(x) \in q_{\mathsf{acyc}}(x)$ such that $p_i'(x) \equiv p(x)$. Then, $q_i(x) \equiv p_i'(x)$ and from Claim 1, $p_i'(x)$ is a sub-query of $q_i(x)$. Let $q^-_{\mathsf{acyc}}(x)$ be the restriction of $q_{\mathsf{acyc}}(x)$ to the chosen CQs $p_1'(x), \ldots, p_k'(x)$. Clearly, $q^-_{\mathsf{acyc}}(x)$ is equivalent to $q(x)$.

Now we concentrate on the "only if" direction for Point 1, i.e. the case where $\mathcal{L}$ is $\mathcal{ALCI}$, and thus $q(x)$ is $\mathcal{ALCI}$-IQ rewritable. We already know that $q^-_{\mathsf{acyc}}(x)$ is equivalent to $q(x)$, thus $q^-_{\mathsf{acyc}}(x)$ is also $\mathcal{ALCI}$-IQ rewritable. From the proof of direction "1 ⇒ 3" Point (b) in Theorem 3 , $\mathcal{ALCI}$-IQ rewritability implies $q(x) \equiv q^{\mathsf{con}}(x)$, and thus also $q^{\mathsf{con}}(x) \equiv (q^-_{\mathsf{acyc}})^{\mathsf{con}}(x)$ and $q(x) \equiv (q^-_{\mathsf{acyc}})^{\mathsf{con}}(x)$. It is easy to see that $(q^-_{\mathsf{acyc}})^{\mathsf{con}}(x)$ is an x-acyclic connected subquery of $q(x)$. Point 2 and the cases with universal roles are treated similarly.

❏

Proposition 6. *Let $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALC}^u, \mathcal{ALCI}^u\}$. If a UCQ $q(x)$ is $\mathcal{L}$-IQ-rewritable, then so is any OMQ $(\mathcal{T}, \Sigma, q(x))$ from $(\mathcal{LH}, UCQ)$.*

Proof. We show the result in the case where $\mathcal{L} = \mathcal{ALCI}$. All other cases follow similarly.

Assume that $q(x)$ is $\mathcal{ALCI}$-IQ-rewritable and let $Q = (\mathcal{T}, \Sigma, q(x))$ be an OMQ from $(\mathcal{LH}, \text{UCQ})$. As a consequence of Theorem 7, there is a subquery $q'(x)$ of $q(x)$ that is x-acyclic, connected and equivalent to $q(x)$. Let $Q' = (\mathcal{T}, \Sigma, q'(x))$. Then, $Q \equiv Q'$ and according to Lemma 4, Q' (and thus also Q) is rewritable into an OMQ from $(\mathcal{ALCI}, \text{IQ})$. ❏

C Proofs for Section 4

For the proofs in this section, we recall that every CQ q can be viewed in a straightforward way as an ABox $\mathcal{A}_q$ by viewing the atoms as assertions and the variables as individual names.

Theorem 8. *For every $\mathcal{Q} \in \{CQ, UCQ\}$ and $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALC}^u, \mathcal{ALCI}^u\}$, it is* NP*-complete to decide whether a given query from $\mathcal{Q}$ is $\mathcal{L}$-IQ-rewritable.*

Proof. Let $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}, \mathcal{ALC}^u, \mathcal{ALCI}^u\}$. We start with the upper bound, that is, given a UCQ $q(x)$, it is in NP to decide whether $q(x)$ is $\mathcal{L}$-IQ-rewritable.

We guess a subquery $q'(x)$ of the original query $q(x)$ and check whether $q'(x) \equiv q(x)$. This is the case when for every CQ $p(x)$ in $q(x)$ there exists a CQ $p'(x)$ in $q'(x)$ such that $p(x) \subseteq p'(x)$ and vice versa. We guess for every CQ $p(x)$ in $q(x)$ a target CQ $p'(x)$ in $q'(x)$ and a potential homomorphism $h_p : \mathsf{ind}(\mathcal{A}_p) \to \mathsf{ind}(\mathcal{A}_{p'})$. We also guess for every CQ $p'(x)$ in $q'(x)$ a target CQ $p(x)$ in $q(x)$ and a potential homomorphism $h_{p'} : \mathsf{ind}(\mathcal{A}_{p'}) \to \mathsf{ind}(\mathcal{A}_p)$. We then check that every h_p and every h_p' is an actual homomorphism. If this is the case, $q'(x) \equiv q(x)$ and, provided that $q'(x)$ fulfills the additional conditions in the $\mathcal{L}$-IQ characterisation from Theorem 8 (x-acyclicity, connectedness and/or x-accesibility), $q(x)$ is $\mathcal{L}$-IQ rewritable. As the size of our guess is polynomial in the size of $q(x)$ and all checks can be performed in polynomial time, we obtain the desired upper bound.

To show NP-hardness of whether a given CQ is $\mathcal{L}$-IQ-rewritable, we employ a reduction from the 3-colorability problem (3COL). Let $G = (V, E)$ be an undirected graph, let q_G be G viewed as a conjunctive query where every $\{v_1, v_2\} \in E$ is represented by two atoms $r(v_1, v_2), r(v_2, v_1)$, and choose a $v \in V$. Let

$$\begin{aligned} q(x_0) \quad = \quad & \exists \mathbf{y}\, q_G \wedge r(x_0, v) \wedge r(v, x_0) \wedge \\ & \bigwedge \{ r(x_i, x_j) \mid i, j \leq 2 \text{ with } i \neq j \} \end{aligned}$$

where $\mathbf{y}$ contains all elements of V (as variables) as well as the fresh variables x_1 and x_2.

Claim. *$q(x_0)$ is $\mathcal{L}$-IQ-rewritable iff G is 3-colorable.*

Proof of claim. For the "if" direction, assume that G is 3-colorable. Then G admits a homomorphism into the 3-clique (without reflexive loops). Consequently, $q_0(x)$ is homomorphically equivalent to the restriction $q_{3C}(x_0)$ of $q(x_0)$ to the variables x_0, x_1, x_2. In particular, $q(x_0)$ and $q_{3C}(x_0)$ are then equivalent in the sense of query containment. Since $q_{3C}(x_0)$ is x_0-acyclic and x_0-accessible, by Lemma 24 it is rewritable into an $\mathcal{ALC}$-IQ.

Conversely, assume that G is not 3-colorable. By Lemma 24, it suffices to show that any subquery $p(x_0)$ of $q(x_0)$ that is equivalent to $q(x_0)$ is not x_0-acyclic. Thus let $p(x_0)$ be such a subquery. There is no homomorphism h from $p(x_0)$ to $q_{3C}(x_0)$ since the equivalence of $p(x_0)$ and $q(x_0)$ implies the existence of a homomorphism h' from $q(x_0)$ to $p(x_0)$ and composing h' with h would establish 3-colorability of G. It is easy to verify, though, that when $p(x_0)$ contains no cycle that does not pass through any of x_0, x_1, x_2, then there is such a homomorphism h. Consequently, $p(x_0)$ is not x_0-acyclic. ❏

Theorem 9. *Let $\mathcal{Q} \in \{CQ, UCQ\}$. For OMQs based on the full ABox signature, IQ-rewritability is*

1. ExpTime-*hard in* $(\mathcal{ALC}, \mathcal{Q})$ *and in* coNExpTime *in* $(\mathcal{ALCH}, \mathcal{Q})$ *and*
2. 2ExpTime-*complete in* $(\mathcal{ALCI}, \mathcal{Q})$ *and* $(\mathcal{ALCHI}, \mathcal{Q})$.

Proof. We start with the lower bounds. Points 1 and 2 are treated uniformly. In fact, for $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}\}$, we reduce a special case of OMQ evaluation in $(\mathcal{L}, \text{CQ})$ to IQ-rewritability in $(\mathcal{L}, \text{CQ})$ where *OMQ evaluation* in $(\mathcal{L}, \mathcal{Q})$ means to decide, given an OMQ $Q = (\mathcal{T}, \Sigma_{\mathsf{full}}, q(\mathbf{x}))$ from $(\mathcal{L}, \mathcal{Q})$, an ABox $\mathcal{A}$, and a tuple $\mathbf{a}$ whether $\mathcal{A} \models Q(\mathbf{a})$. The mentioned special case is that OMQs are Boolean and $\mathcal{A}$ takes the form $\{A(a)\}$ and we refer to this as *singleton BOMQ evaluation*.

Singleton BOMQ evaluation is 2ExpTime-hard in $\mathcal{ALCI}$ [Lutz, 2008]. We observe that it is ExpTime-hard in $\mathcal{ALC}$ since concept (un)satisfiability w.r.t. $\mathcal{ALC}$-TBoxes is ExpTime-hard [Schild, 1991] and an $\mathcal{ALC}$-concept C is unsatisfiable w.r.t. an $\mathcal{ALC}$-TBox $\mathcal{T}$ iff $\{A(a)\} \models (\mathcal{T} \cup \{A \sqsubseteq C\}, \Sigma_{\mathsf{full}}, \exists y\, D(y))$ where A and D are fresh concept names.

Now for the reduction to IQ-rewritability. Let $Q = (\mathcal{T}, \Sigma_{\mathsf{full}}, q())$ be an OMQ from $(\mathcal{L}, \text{CQ})$, $\mathcal{L} \in \{\mathcal{ALC}, \mathcal{ALCI}\}$, and let $\mathcal{A} = \{A(a)\}$ be an ABox. Further, let $q'(x)$ is the extension of $q()$ with the atom $A(x)$, x a fresh answer variable. It is important to note that $q'(x)$ is a disconnected CQ.

Claim. $\mathcal{A} \models Q$ iff $Q' = (\mathcal{T}, \Sigma_{\mathsf{full}}, q'(x))$ is IQ-rewritable.

Proof of claim. If $\mathcal{A} \models Q$, then Q' is equivalent to $(\mathcal{T}, \Sigma_{\mathsf{full}}, A(x))$ which is from $(\mathcal{L}, \text{IQ})$. Conversely, assume that $\mathcal{A} \not\models Q$. The query Q^{con} from Point (b) in the proof of the "1 ⇒ 3" direction of Theorem 3, applied to Q', is exactly $(\mathcal{T}, \Sigma_{\mathsf{full}}, A(x))$. As shown there, IQ-rewritability of Q implies $Q \equiv Q^{\mathsf{con}}$, in contradiction to $\mathcal{A} \not\models Q$. This finishes the proof of the claim.

For the upper bounds, we use the characterizations from Theorem 3 and Theorem 5: deciding IQ-rewritability in $(\mathcal{ALCHI}, \text{UCQ})$ amounts to checking containment between Q and $(\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$ while deciding IQ-rewritability in $(\mathcal{ALCH}, \text{UCQ})$ amounts to checking containment between Q and respectively $(\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{deco}}_{\mathsf{acyc}}(x))$. Note that the two involved OMQs share the same TBox and are based on the full ABox signature. There is also an initial emptiness check, which however is just another containment check. We thus have to argue that these containment checks can be carried out in 2ExpTime and NExpTime, respectively.

We start with the case of $(\mathcal{ALCHI}, \text{UCQ})$ and first observe that containment in $(\mathcal{ALCHI}, \text{UCQ})$ is in 2ExpTime. In fact, it is shown in [Lutz, 2008] that OMQ evaluation in $(\mathcal{ALCHI}, \text{CQ})$ is in 2ExpTime and the algorithm given there is straightforwardly extended to $(\mathcal{ALCHI}, \text{UCQ})$. It follows that containment between an OMQ $Q_1 = (\mathcal{T}, \Sigma_{\mathsf{full}}, q_1(x_1))$ from $(\mathcal{ALCHI}, \text{CQ})$ in an OMQ $Q_2 = (\mathcal{T}, \Sigma_{\mathsf{full}}, q_2(x_2))$ from $(\mathcal{ALCHI}, \text{UCQ})$ is in 2ExpTime since $Q_1 \subseteq Q_2$ iff $\mathcal{A}_{q_1} \models Q_2(x_1)$.

We next observe how this can be lifted to containment in $(\mathcal{ALCHI}, \text{UCQ})$. In fact, it suffices to show that for $Q_i = (\mathcal{T}, \Sigma_{\mathsf{full}}, q_i)$ from $(\mathcal{ALCHI}, \text{UCQ})$, $i \in \{1, 2\}$, and $q_1 = p_1 \vee \cdots \vee p_k$, we have $Q_1 \subseteq Q_2$ iff $(\mathcal{T}, \Sigma_{\mathsf{full}}, p_i) \subseteq Q_2$ for all $i \in \{1, \ldots, k\}$. The "if" direction is trivial. For the "only if" direction, we argue as follows. Assume that $(\mathcal{T}, \Sigma_{\mathsf{full}}, p_i) \subseteq Q_2$ for all $i \in \{1, \ldots, k\}$. Let $\mathcal{A}$ be an ABox and $a \in \text{ind}(\mathcal{A})$ such that $\mathcal{A} \models Q_1(a)$. It suffices to show that $\mathcal{I} \models q_2(a)$ for every finite model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}$. Let $\mathcal{I}$ be such a model and let $\mathcal{A}_\mathcal{I}$ be $\mathcal{I}$ viewed as an ABox. Since $\mathcal{A} \models Q_1(a)$, we must have $\mathcal{I} \models p_j(a)$ for some $j \in \{1, \ldots, k\}$. Then clearly also $\mathcal{A}_\mathcal{I} \models p_j(a)$. Since $(\mathcal{T}, \Sigma, p_j) \subseteq Q_2$, this yields $\mathcal{A}_\mathcal{I} \models Q_2(a)$. Since $\mathcal{I}$ is a model of $\mathcal{A}_\mathcal{I}$ and $\mathcal{T}$, from this we obtain $\mathcal{I} \models q_2(a)$, as required.

The argument is not yet complete since the UCQs $q^{\mathsf{con}}_{\mathsf{acyc}}(x)$ can be exponentially large. In fact, it may contain exponentially many CQs, but each CQ is only of polynomial size. For checking $(\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{con}}_{\mathsf{acyc}}(x)) \subseteq Q$, using the above argument we can use exponentially many containment checks between an OMQ from $(\mathcal{ALCHI}, \text{CQ})$ and an OMQ from $(\mathcal{ALCHI}, \text{UCQ})$, both of polynomial size. The overall complexity is thus 2ExpTime, as required. For checking $Q \subseteq (\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$, we observe that, by Lemma 4, $(\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$ is rewritable into an equivalent OMQ $(\mathcal{T}, \Sigma_{\mathsf{full}}, C(x))$ with $C(x)$ an $\mathcal{ALCHI}$-IQ (independently of the properties of Q) and such that the size of $C(x)$ is polynomial in the size of $q^{\mathsf{con}}_{\mathsf{acyc}}(x)$, which in turn is single exponential in the size of Q. We can thus replace the check $Q \subseteq (\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{con}}_{\mathsf{acyc}}(x))$ with $Q \subseteq (\mathcal{T}, \Sigma_{\mathsf{full}}, C(x))$. This boils down to deciding OMQ entailment in $(\mathcal{ALCHI}, IQ)$, which is in ExpTime. So despite $C(x)$ being of (single) exponential size, we achieve 2ExpTime overall complexity.

For the case of $(\mathcal{ALCH}, \text{UCQ})$, the argument is essentially the same. However, as also shown in [Lutz, 2008] OMQ evaluation in $(\mathcal{ALCH}, \text{UCQ})$ is in ExpTime and thus so is our basic containment check between an OMQ from $(\mathcal{ALCH}, \text{CQ})$ and an OMQ from $(\mathcal{ALCHI}, \text{UCQ})$. Therefore, the check $(\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{deco}}_{\mathsf{acyc}}(x)) \subseteq Q$ can be implemented in ExpTime despite the exponential number of CQs in $q^{\mathsf{deco}}_{\mathsf{acyc}}$. It is not clear, however, how to implement the containment check $Q \subseteq (\mathcal{T}, \Sigma_{\mathsf{full}}, q^{\mathsf{deco}}_{\mathsf{acyc}}(x))$ in ExpTime. We give a sketch of how it can be implemented in coNExpTime. In fact, what we have to implement in coNExpTime is the evaluation of an OMQ $Q = (\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$ where $q(x)$ is a UCQ with exponentially many connected CQs, each of polynomial size. Let $\mathcal{A}$ be an ABox and $a \in \mathsf{Ind}(\mathcal{A})$. By Lemma 16, $\mathcal{A} \not\models Q(a)$ iff there is an extended forest model $\mathcal{I}$ of $\mathcal{A}$ and $\mathcal{T}$ such that $\mathcal{I} \not\models q(a)$. It is easy to see that we can further demand that (the tree-shaped parts of) $\mathcal{I}$ be of outdegree polynomial in the size of $\mathcal{T}$. Our NExpTime algorithm for the complement of OMQ evaluation is as follows. Let m be the maximum number of variables of a CQ in $q(x)$. We guess an initial piece of the extended forest model $\mathcal{I}$ that consists of the 'ABox part' of $\mathcal{A}$ together with the tree-shaped parts restricted to depth $m + 1$, along with a type adornment, that is, a function μ that assigns a $\mathcal{T}$-type to every element of the guessed initial part in a way that is consistent with the initial part. Note that we guess an object of single exponential size here. Since the CQ in $q(x)$ are connected and thus are independent of the part of $\mathcal{I}$ that lies beyond the guessed initial part, we can verify that $\mathcal{I}$ can be extended to a full model by considering the type $\mu(d)$ for every leaf d in the initial part on level $m + 1$ and verifying that $\mu(d)$ is satisfiable with $\mathcal{T}$.

This can be implemented in ExpTime. ❑

We introduce a preparatory lemma and notation for the proof of Theorem 5. An *atomic query (AQ)* is an IQ of the form $A(x)$, with A a concept name. A *Boolean atomic query (BAQ)* is a query of the form $\exists x\, A(x)$, with A a concept name, and a *Boolean conjunctive query (BCQ)* is a CQ of arity zero.

Lemma 26. *Let $Q = (\mathcal{T}, \Sigma, \exists x\, C(x))$ be an OMQ from $(\mathcal{ALCI}, BAQ)$. Then, there exists an OMQ $Q' = (\mathcal{T}', \Sigma, M(x))$ from $(\mathcal{ALCI}, AQ)$, such that for all Σ-ABoxes $\mathcal{A}$, $\mathcal{A} \models Q$ iff there is an $a \in \mathsf{ind}(\mathcal{A})$ such that $\mathcal{A} \models Q'(a)$.*

Proof. We first observe that $\mathcal{A} \models Q$ iff $\mathcal{A}' \models Q$, for some $\mathcal{A}' \in \mathsf{CON}_{\mathcal{A}}$. Thus, it is enough to show the statement of the lemma for connected Σ-ABoxes.

Let $\mathcal{A}'$ be a connected Σ-ABox and M a fresh concept name. Let $\mathcal{T}'$ be the TBox obtained from $\mathcal{T}$ by adding $C \sqsubseteq M$ and $\exists r.M \sqsubseteq M$ for every role r such that r or the inverse of r occurs in $\mathcal{T}$. It can be verified that $Q' = (\mathcal{T}', \Sigma, M(x))$ is as required. ❑

We also introduce a more fine-grained version of a complexity result from [Bourhis and Lutz, 2016] which highlights that the complexity of containment is double exponential only in the maximum size of CQs in the input OMQs, but not in their number. This only requires a careful analysis of the constructions in [Bourhis and Lutz, 2016].

Theorem 27. *Containment between OMQs from $(\mathcal{ALCHI}, UCQ)$ is in* 2NExpTime. *More precisely, for OMQs $Q_1 = (\mathcal{T}_1, \Sigma, q_1)$ and $Q_2 = (\mathcal{T}_2, \Sigma, q_2)$ with arity a and where n_i is the number of CQs in q_1 and n_i the maximum size of a CQ in q_i, $i \in \{1, 2\}$, it can be decided in time $2^{2^{p(\log n_1 + s_1 + \log n_2 + s_2 + \log|\mathcal{T}| + \log\log a)}}$, p a polynomial.*

Theorem 10. *IQ-rewritability is*

1. NExpTime*-hard in $(\mathcal{ALC}, CQ)$ and*
2. 2NExpTime*-complete in all of $(\mathcal{ALCI}, CQ)$, $(\mathcal{ALCI}, UCQ)$, $(\mathcal{ALCHI}, CQ)$, $(\mathcal{ALCHI}, UCQ)$.*

Proof. We start with the lower bound for Point 1, using a reduction from OMQ emptiness in $(\mathcal{ALC}, \text{AQ})$ which is known to be NExpTime-hard. Let $Q_0 = (\mathcal{T}, \Sigma, A(x))$ be an OMQ from this language. Also let

$$q(x) = \exists y\, A(x) \wedge r(x, y) \wedge r(y, y),$$

where r is a role name that does not occur in $\mathcal{T}$ and let $Q = (\mathcal{T}, \Sigma \cup \{r\}, q(x))$. It suffices to show that Q_0 is empty iff Q is IQ-rewritable.

In fact, emptiness of Q_0 implies emptiness of Q and thus $(\mathcal{ALC}, \text{AQ})$-rewritability. Conversely, assume that Q_0 is non-empty. To show that Q is not IQ-rewritable, by Theorem 5 it suffices to show that $Q \not\equiv Q^{\mathsf{deco}}_{\mathsf{acyc}} := (\mathcal{T}, \Sigma \cup \{r\}, q^{\mathsf{deco}}_{\mathsf{acyc}}(x))$ where $q^{\mathsf{deco}}_{\mathsf{acyc}}(x)$ is a UCQ in which every CQ contains the subquery $r(x, x)$. Since Q_0 is non-empty, there is a Σ-ABox $\mathcal{A}$ and an $a \in \mathsf{ind}(\mathcal{A})$ such that $\mathcal{A} \models Q_0(a)$. Since r does not occur in $\mathcal{T}$, we can assume w.l.o.g. that it does not occur in $\mathcal{A}$ as well. Let $\mathcal{A}' = \mathcal{A} \cup \{r(a, b), r(b, b)\}$. By definition of $Q(a)$, clearly $\mathcal{A}' \models Q(a)$. Moreover, $\mathcal{A}' \not\models Q^{\mathsf{deco}}_{\mathsf{acyc}}(a)$ because $\mathcal{A}'$ does not contain the assertion $r(a, a)$.

To establish the lower bound for Point 2, we use a reduction from containment between an OMQ from $(\mathcal{ALCI}, \text{BAQ})$ and an OMQ from $(\mathcal{ALCI}, \text{BCQ})$. This problem has been shown to be 2NExpTime-hard in [Bourhis and Lutz, 2016]. The reduction presented there uses different TBoxes in the two involved OMQs. However, by Theorem 3 in [Bienvenu *et al.*, 2012], we can assume w.l.o.g. that they both share the same TBox.

Now for the reduction to IQ-rewritability. Let $Q_1 = (\mathcal{T}, \Sigma, \exists x\, A(x))$ be an OMQ from $(\mathcal{ALCI}, \text{BAQ})$ and $Q_2 = (\mathcal{T}, \Sigma, q())$ an OMQ from $(\mathcal{ALCI}, \text{BCQ})$. We first show that Q_1 and Q_2, which are Boolean, can be replaced with unary OMQs. By Lemma 26, we find an OMQ $Q_1' = (\mathcal{T}', \Sigma, M(x))$ from $(\mathcal{ALCI}, \text{AQ})$ such that for all Σ-ABoxes $\mathcal{A}$, $\mathcal{A} \models Q_1$ iff $\mathcal{A} \models Q_1'(a)$ for some $a \in \mathsf{ind}(\mathcal{A})$. Clearly, the construction from the proof of Lemma 26 is such that Q_1 is equivalent to $(\mathcal{T}', \Sigma, \exists x\, A(x))$ and by choosing the fresh concept M to also not occur in $q()$ we can further ensure that Q_2 is equivalent to $(\mathcal{T}', \Sigma, q())$. Thus, we can assume that Q_1, Q_2, and Q_1' all use the same TBox $\mathcal{T}$ and we can further assume $\mathcal{T}$ contains a CI $\top \sqsubseteq N$ where N is a concept name not occurring anywhere else, including Σ. Set $Q_2' = (\mathcal{T}, \Sigma, q'(x))$ where $q'(x)$ is $q()$ extended with the atom $N(x)$, x a fresh (answer) variable. It can be verified that $Q_1 \subseteq Q_2$ iff $Q_1' \subseteq Q_2'$.

Now let $q_0(x)$ be $q()$ extended with the atom $M(x)$, x a fresh (answer) variable. It is important to note that $q_0(x)$ is a disconnected CQ.

Claim. $Q_1' \subseteq Q_2'$ iff $Q = (\mathcal{T}, \Sigma, q_0(x))$ is IQ-rewritable.

Proof of claim. If $Q_1' \subseteq Q_2'$, then Q is equivalent to Q_1', which is from $(\mathcal{ALCI}, \text{IQ})$. Conversely, assume that $Q_1' \not\subseteq Q_2'$. The query Q^{con} from Point (b) in the proof of the "1 ⇒ 3" direction of Theorem 3, applied to Q, is exactly Q_1'. As shown there, IQ-rewritability of Q implies $Q \equiv Q^{\mathsf{con}}$, in contradiction to $Q_1' \not\subseteq Q_2'$.

The upper bounds are a consequence of the characterization of IQ-rewritability in $(\mathcal{ALCHI}, \text{UCQ})$ from Theorem 5 in terms of query containment. Containment in $(\mathcal{ALCHI}, \text{UCQ})$ is in 2NExpTime [Bourhis and Lutz, 2016]. Note that our characterizations use UCQs with exponentially many CQs, each of which is of polynomial size so we cannot apply the containment complexity result as a black box. However, by perusing the refined OMQ containment complexity result from Theorem 27, we observe that containment checking for OMQs is double exponential only in the size of the CQs in the UCQs while it is only exponential in the number of CQs.

Note that we also need an emptiness check beforehand: if the check succeeds and the OMQ is empty, it is also rewritable and so we answer 'yes', if not we proceed to perform the containment check. Emptiness is simply a special case of containment, so we end up in the right complexity class. ❑

D Proofs for Section 5

We postpone the proof of Theorem 11 as we need the ultrafilter technique introduced in the proof of Theorem 13. We start by discussing the rewritings given in Example 12 in more detail and present an additional example. Recall that $p(x) = \exists y(s(x,y) \wedge r(y,y))$ and that we consider the OMQ $Q = (\mathcal{T}_r, \Sigma_{\mathsf{full}}, p(x))$ with $\mathcal{T}_r = \{\mathsf{func}(r)\}$. We claim that $Q_r = (\mathcal{T}_r, \Sigma_{\mathsf{full}}, q_r(x))$ with

$$q_r(x) = (\forall s. \bigsqcup_{1 \leq i \leq 3} P_i) \rightarrow (\exists s. (\bigsqcap_{1 \leq i \leq 3} (P_i \rightarrow \exists r. P_i))$$

is a rewriting of Q. To prove this claim one can use the following straightforward three colorability argument: for every set X of individual names in an ABox $\mathcal{A}$ which does not contain an atom of the form $r(c,c)$ with $c \in X$ and in which r is functional (in the sense that for any a there is at most one b with $r(a,b) \in \mathcal{A}$) one can color the individual names in X with three different colors P_1, P_2, P_3 without having distinct c_1, c_2 with $r(c_1, c_2) \in \mathcal{A}$ such that $P_i(c_1)$ and $P_i(c_2)$ for some $1 \leq i \leq 3$. We give an additional example illustrating this technique which is fundamental for our approach to rewritability for $\mathcal{ALCIF}$ TBoxes.

Example 28. Let $\mathcal{T} = \{\mathsf{func}(s_1), \mathsf{func}(s_2)\}$ and

$$p(x) = \exists y, z(r(x,y) \wedge s_1(y,z) \wedge s_2(y,z))$$

and obtain $p'(x)$ from $p(x)$ by adding the atom $s_1(z,y)$. Then $(\mathcal{T}, \Sigma_{\mathsf{full}}, p(x))$ is not rewritable into an OMQ in $(\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$ with $q(x)$ a CI but $(\mathcal{T}, \emptyset, \Sigma_{\mathsf{full}}, p'(x))$ is rewritable into the OMQ $(\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$, where

$$\begin{aligned} q(x) &= \forall r. ((\bigsqcup_{1 \leq i \leq 3} P_i) \sqcap (\forall r. \forall s_1. \bigsqcup_{1 \leq i \leq 3} Q_i) \\ &\rightarrow (\exists r. (\bigsqcap_{1 \leq i \leq 3} (P_i \rightarrow \exists s_1.(C \sqcap \exists s_1. P_i)) \end{aligned}$$

where

$$C = \bigsqcap_{1 \leq i \leq 3} (Q_i \rightarrow \exists s_1. \exists s_2. Q_i)$$

We split the proof of Theorem 13 into two parts and state the claims in such a way that they cover the extensions discussed in the main text. Call a role r *functional w.r.t.* $\mathcal{T}$ if $\mathsf{func}(r) \in \mathcal{T}$. The *functional closure* $FC_p(x_0)$ *of a variable* x_0 *in a CQ* $p(x)$ is the set of all variables x_n in $p(x)$ such that there is a functional path from x_0 to x_n in $p(x)$. Let $\mathrm{var}(p(x))$ denote the set of all variables in $p(x)$ and set

$$\mathrm{nFC}_p(x) = \mathrm{var}(p(x)) \setminus \mathrm{FC}_p(x)$$

Lemma 29. *Let $Q = (\mathcal{T}, \Sigma_{full}, q(x))$ be an OMQ from $(\mathcal{ALCIF}, UCQ)$. Then Q is rewritable into an OMQ $(\mathcal{T}, \Sigma_{full}, q'(x))$ with $q'(x)$ an $\mathcal{ALCI}$-IQ if there is a subquery $q'(x)$ of $q(x)$ that is f-acyclic, connected, and equivalent to $q(x)$.*

When $\mathcal{ALCIF}$ is replaced with $\mathcal{ALCIF}^u$, then the implication holds without the connectedness assumption.

Proof. Fix an f-acyclic and connected UCQ $q(x)$ and let $\mathcal{T}$ be a $\mathcal{ALCIF}$ TBox. Let $Q = (\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$. We construct an $\mathcal{ALCI}$-IQ $C(x)$ such that $Q \equiv Q'$ for $Q' = (\mathcal{T}, \Sigma_{\mathsf{full}}, C(x))$. Let $p(x)$ be a CQ in $q(x)$. A *cluster* in $p(x)$ is any maximal subset X of $\mathrm{nFC}_p(x)$ such that there is a functional path from any $y \in X$ to any $y' \in X$. Denote the set of clusters in $p(x)$ by $\mathfrak{C}$. For any two clusters X_1, X_2 we set $(X_1, X_2) \in E$ if there exist $y_1 \in X_1$ and $y_2 \in X_2$ such that there is an atom $r(y_1, y_2)$ in $p(x)$. We obtain an undirected graph $(\mathfrak{C}, E)$ without self-loops. As $p(x)$ is f-acyclic, we obtain that

- $(\mathfrak{C}, E)$ is acyclic;
- for any $(X_1, X_2) \in E$ there is exactly one atom $r(y_1, y_2)$ in $p(x)$ with $y_1 \in X_1$ and $y_2 \in X_2$;
- there does not exist an atom $r(y, y')$ in $p(x)$ such that neither r nor r^- are functional w.r.t. $\mathcal{T}$ and $y, y' \in X$ for a single cluster X in $p(x)$.

It follows that we obtain from $p(x)$ a new CQ $p'(x)$ by repeatedly choosing and removing

- atoms $r(x', y')$ with $x' \in \mathrm{FC}_p(x)$ and $y' \in \mathrm{nFC}_p(x)$ and
- atoms $r(y', y'')$ with y', y'' contained in the same cluster X and
- atoms $r(x', x'')$ with $x', x'' \in \mathrm{FC}_p(x)$

such that $p'(x)$ is a tree-shaped CQ with answer variable x still containing all variables in $p(x)$ and

- every x' in $\mathrm{FC}_p(x)$ is still reachable in $p'(x)|_{\mathrm{FC}_q(x)}$ from x along a functional path;
- for every cluster X there exists a $y_X \in X$ such that every $y \in X$ can be reached in $p'(x)|_X$ from y_X along a functional path.

Now obtain a CQ $p_{\mathsf{plain}}(x)$ from $p'(x)$ by adding for every y in $p'(x)$ a fresh atom $A_y(y)$ to $p'(x)$. Obtain an eCQ $p_{\mathsf{deco}}(x)$ from $p_{\mathsf{plain}}(x)$ by adding

- for every atom $r(x_1, x_2) \in p(x) \setminus p'(x)$ with $x_1, x_2 \in \mathrm{FC}_p(x)$ the compound 'atom' $\exists r. A_{x_2}(x_1)$ and
- for every atom $r(y_1, x_1) \in p(x) \setminus p'(x)$ with $y_1 \in \mathrm{nFC}_p(x)$ and $x_1 \in \mathrm{FC}_p(x)$ the compound atom $\exists r. A_{x_1}(y_1)$.

Replace in the $\mathcal{ELI}$ concept corresponding to $p_{\mathsf{plain}}(x)$ all occurrences of the symbol '$\exists$' by the symbol '$\forall$' and denote the resulting $\mathcal{ALCI}$ concept by C_{plain}. Take the $\mathcal{ELI}$ concept D_{deco} corresponding to $p_{\mathsf{deco}}(x)$. Consider the $\mathcal{ALCI}$ concept $C_{\mathsf{plain}} \rightarrow D_{\mathsf{deco}}$. It should be clear that $(\mathcal{T}, \Sigma_{\mathsf{full}}, C_{\mathsf{plain}} \rightarrow D_{\mathsf{deco}}(x))$ is a rewriting of $(\mathcal{T}, \Sigma_{\mathsf{full}}, p(x))$ if all clusters in $p(x)$ are degenerate in the sense that they consist of a single variable y such that there is no atom of the form $r(y,y)$ in $p(x)$. More generally, $(\mathcal{T}, \Sigma_{\mathsf{full}}, C_{\mathsf{plain}} \rightarrow D_{\mathsf{deco}}(x))$ is a rewriting of $(\mathcal{T}, \Sigma_{\mathsf{full}}, p^*(x))$ for the CQ $p^*(x)$ obtained from $p(x)$ by removing all atoms $r(y_1, y_2)$ with variables y_1, y_2 from a single cluster which are not in $p'(x)$. To obtain the rewriting we are after we thus still have to take care of those atoms.

Consider a non-degenerate cluster X. We find an ordering $r_1(x_1^1, x_2^1), \ldots r_n(x_1^n, x_2^n)$ of the atoms in $p(x)|_X \setminus p'|_X$ such that all r_i are functional (recall that no atoms $r(y, y')$ with neither r nor r^- functional and $y, y' \in X$ exist) and for inductively defined sets of atoms

$$\begin{aligned} p^0|_X &= p'|_X \\ p^{i+1}|_X &= p^i|_X \cup \{r_{i+1}(x_1^{i+1}, x_2^{i+1})\} \end{aligned}$$

the following holds: for all $0 \leq i < n$, there is a functional path $y_0^i, \ldots, y_k^i$ in $p^i|_X$ such that $y_0^i = y_k^i$, $y_{k-1}^i = x_1^{i+1}$ and $y_k^i = x_2^{i+1}$. Now take for every $0 \leq i < n$ fresh concept names $P_{X,i}^1, P_{X,i}^2, P_{X,i}^3$ and obtain $p_{\mathsf{plain}}^X(x)$ from $p_{\mathsf{plain}}(x)$ by adding the compound atoms

$$P_{X,i}^1 \sqcup P_{X,i}^2 \sqcup P_{X,i}^3(y_0^i)$$

Take functional roles $s_0^i, \ldots, s_{k-2}^i$ and consider the query

$$s_0^i(y_0^i, y_1^i), \ldots, s_{k-2}^i(y_{k-2}^i, y_{k-1}^i) \in p^i|_X$$

and consider the CQ $p_X^i(y_0^i)$ defined by taking the conjunction of the atoms $s_0^i(y_0^i, y_1^i), \ldots, s_{k-2}^i(y_{k-2}^i, y_{k-1}^i)$ and taking y_0^i as the answer variable. Define for $1 \leq j \leq 3$ an eCQ $p_X^{i,j}(y_0^i)$ by adding to $p_X^i(y_0^i)$ the compound atom

$$\exists r_{i+1}.P_{X,i}^j(y_{k-1}^i)).$$

Take the $\mathcal{ELI}$ concept $D_{X,i}^j$ corresponding to $p_X^{i,j}(y_0^i)$, for $j = 1, 2, 3$. Obtain the CQ $p_{\mathsf{deco}}^X(x)$ from $p_{\mathsf{deco}}(x)$ by adding the compound atoms

$$\bigsqcap_{1 \leq j \leq 3} (P_{X,i}^j \rightarrow D_{X,i}^j)(y_0^i)$$

We do this for all non-degenerate clusters X and obtain eCQs $p_{\mathsf{plain}}^*(x)$ and $p_{\mathsf{deco}}^*(x)$ by taking the conjunction of all $p_{\mathsf{plain}}^X(x)$ and $p_{\mathsf{deco}}^X(x)$, respectively. Now define the concepts C_{plain}^* and D_{deco}^* in the obvious way. It is not difficult to prove that $(\mathcal{T}, \Sigma_{\mathsf{full}}, C_{\mathsf{plain}}^* \rightarrow D_{\mathsf{deco}}^*(x))$ is a rewriting of $(\mathcal{T}, \Sigma_{\mathsf{full}}, p(x))$. By constructing C_{plain}^* and D_{deco}^* for all CQs $p(x)$ in $q(x)$ and taking the disjunction of all $C_{\mathsf{plain}}^* \rightarrow D_{\mathsf{deco}}^*(x)$ we obtain a rewriting of $q(x)$. The extension to $\mathcal{ALCIF}^u$ without the connectedness assumption is straightforward. ❏

For the proof of the other direction of Theorem 13, we require some preparation. We use standard notation and results for ultrafilter extensions of interpretations [Blackburn *et al.*, 2002]. For $U \subseteq \Delta^{\mathcal{I}}$ for an interpretation $\mathcal{I}$ we set $\overline{U} = \Delta^{\mathcal{I}} \setminus U$.

Definition 30. Let $\mathcal{I}$ be an interpretation. A set $\mathfrak{U} \subseteq 2^{\Delta^{\mathcal{I}}}$ is an *ultrafilter* over $\Delta^{\mathcal{I}}$ if the following conditions hold for all $U, V \subseteq \Delta^{\mathcal{I}}$:

- if $U, V \in \mathfrak{U}$, then $U \cap V \in \mathfrak{U}$;
- if $U \in \mathfrak{U}$ and $U \subseteq V$, then $V \in \mathfrak{U}$;
- $U \in \mathfrak{U}$ iff $\overline{U} \notin \mathfrak{U}$.

For every $d \in \Delta^{\mathcal{I}}$, the set

$$\mathfrak{U}_d = \{X \subseteq \Delta^{\mathcal{I}} \mid d \in X\}$$

is an ultrafilter, called the *principal ultrafilter* generated by d in $\Delta^{\mathcal{I}}$. An ultrafilter $\mathfrak{U}$ for which there exists no $d \in \Delta^{\mathcal{I}}$ with $\mathfrak{U} = \mathfrak{U}_d$ is called a *non-principal* ultrafilter. It is known that for every set $\mathfrak{V} \subseteq 2^{\Delta^{\mathcal{I}}}$ with the *finite intersection property* (if $U_1, \ldots, U_n \in \mathfrak{V}$, then $U_1 \cap \cdots \cap U_n \neq \emptyset$) there exists an ultrafilter $\mathfrak{U} \supseteq \mathfrak{V}$. For a set $U \subseteq \Delta^{\mathcal{I}}$ and role r we set

$$(\exists r.U)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid \text{there exists } d' \in U \text{ with } (d, d') \in r^{\mathcal{I}}\}$$

We are in the position now to define ultrafilter extensions of interpretations.

Definition 31. Let $\mathcal{I}$ be an interpretation. The *ultrafilter extension* $\mathcal{I}^{\mathsf{ue}}$ of $\mathcal{I}$ is defined as follows:

- $\Delta^{\mathcal{I}^{\mathsf{ue}}}$ is the set of ultrafilters over $\Delta^{\mathcal{I}}$;
- $\mathfrak{U} \in A^{\mathcal{I}^{\mathsf{ue}}}$ iff $A^{\mathcal{I}} \in \mathfrak{U}$, for all concept names A;
- $(\mathfrak{U}_1, \mathfrak{U}_2) \in r^{\mathcal{I}^{\mathsf{ue}}}$ iff $(\exists r.U)^{\mathcal{I}} \in \mathfrak{U}_1$ for all $U \in \mathfrak{U}_2$, for all role names r.

Observe the following equivalence for all principal ultrafilters $\mathfrak{U}_d, \mathfrak{U}_e$ and all roles r:

$$(d, e) \in r^{\mathcal{I}} \quad \Leftrightarrow \quad (\mathfrak{U}_d, \mathfrak{U}_e) \in r^{\mathcal{I}^{\mathsf{ue}}}$$

The fundamental property of ultrafilter extensions is the following *anti-preservation result*:

Lemma 32. *For all interpretations $\mathcal{I}$, $\mathcal{ALCI}$ concept C, and roles r:*

- *if $\mathcal{I} \models C(a)$, then $\mathcal{I}^{ue} \models C(U_a)$.*
- *if $r^{\mathcal{I}}$ is functional, then $r^{\mathcal{I}^{ue}}$ is functional.*

We are now in the position to prove the second part of Theorem 13.

Lemma 33. *If an OMQ $Q = (\mathcal{T}, \Sigma_{full}, q(x))$ from $(\mathcal{F}, UCQ)$ is rewritable into an OMQ from $(\mathcal{F}, \mathcal{ALCI}$-IQ), then there is a subquery $q'(x)$ of $q(x)$ that is f-acyclic, connected, and equivalent to $q(x)$.*

When $\mathcal{ALCI}$-IQ is replaced with $\mathcal{ALCI}^u$-IQ, then the same equivalence holds except that connectedness is dropped.

Proof. Assume that $\mathcal{T}$ contains functionality assertions only and $q(x)$ is a UCQ such that there does not exist an equivalent subquery $q'(x)$ of $q(x)$ which is f-acyclic and connected. We may assume that

- there is no homomorphism from any disjunct $p(x)$ of $q(x)$ to another disjunct $p'(x)$ of $q(x)$;
- every homomorphism from any disjunct $p(x)$ into itself is surjective.

Take a disjunct $p(x)$ of $q(x)$ which is not f-acyclic or not connected. We consider the case that $p(x)$ is not f-acyclic but connected. The case that $p(x)$ is not connected is straightforward. We have a cycle $r_0(x_0, x_1), \ldots, r_{n-1}(x_{n-1}, x_n)$ in $p(x)$ such that $\text{FC}_q(x) \cap \{x_0, \ldots, x_{n-1}\} = \emptyset$ and

1. r_i or r_i^- is not functional w.r.t. $\mathcal{T}$ for some $i < n$ or
2. there exists no functional path $y_0, \ldots, y_m$ in $p(x)$ with $x_0 = y_0 = y_m$ such that $\{x_0, \ldots, x_{n-1}\} \subseteq \{y_0, \ldots, y_m\}$.

The basic idea of the proof for both Point 1 and Point 2 is as follows. Assume there exists a a rewriting $(\mathcal{T}, \Sigma_{\mathsf{full}}, C(x))$ of $(\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$.

(a) Using the CQ $p(x)$ we construct an infinite ABox $\mathcal{A}$ such that $\mathcal{T}, \mathcal{A} \not\models q(a)$;

(b) By compactness of FO there exists a forest model $\mathcal{I}$ of $\mathcal{T}$ and $\mathcal{A}$ with $\mathcal{I} \models \neg C(a)$;

(c) Then $\mathcal{I}^{\mathsf{ue}} \models \neg C(\mathfrak{U}_a)$ by Lemma 32;

(d) Moreover, $\mathcal{I}^{\text{ue}}$ is a model of a finite ABox $\mathcal{A}'$ with individual name $\mathfrak{U}_a$ such that $\mathcal{T}, \mathcal{A}' \models q(\mathfrak{U}_a)$. But this contradicts $\mathcal{I}^{\text{ue}} \models \neg C(\mathfrak{U}_a)$.

For Point 1, the construction of $\mathcal{A}$ is a variant of a construction given in [Kikot and Zolin, 2013]. Consider the cycle $r_0(x_0, x_1), \ldots, r_{n-1}(x_{n-1}, x_n)$ in $p(x)$ such that for some i neither r nor r^- are functional w.r.t. $\mathcal{T}$. We may assume that $i = 0$. Let V be the connected component of $\{x_0, \ldots, x_{n-1}\}$ in $\text{nFC}_p(x)$. Regard the variables of $p(x)$ as individual names. Define an ABox $\mathcal{A}$ with individuals

$$\text{FC}_p(x) \cup (\text{nFC}_p(x) \setminus V) \cup (V \times \mathbb{N})$$

by setting:

- for all concept names A and variables $y \in \text{FC}_p(x) \cup (\text{nFC}_p(x) \setminus V)$: $A(y) \in \mathcal{A}$ iff $A(y)$ is in $p(x)$;
- for all concept names A, variables $y \in V$, and $i \in \mathbb{N}$: $A(y, i) \in \mathcal{A}$ iff $A(y) \in q(x)$;
- for all roles r and variables $y, z \in \text{FC}_p(x) \cup (\text{nFC}_p(x) \setminus V)$: $r(y, z) \in \mathcal{A}$ iff $r(y, z)$ is in $p(x)$;
- for all roles r, variables $y \in \text{FC}_p(x) \cup (\text{nFC}_p(x) \setminus V)$ and $z \in V$, and $i \in \mathbb{N}$: $r(y, (z, i)) \in \mathcal{A}$ iff $r(y, z)$ is in $p(x)$;
- for all roles r, variables $y, y' \in V$, and $i \in \mathbb{N}$: $r((y, i), (y', i)) \in \mathcal{A}$ iff $r(y, y')$ is in $p(x)$ and $r_0(x_0, x_1) \neq r(y, y')$;
- for all roles r, variables $y, y' \in V$, and $i \neq j \in \mathbb{N}$, $r((y, i), (y', j)) \in \mathcal{A}$ iff $i < j$ and $r_0(x_0, x_1) = r(y, y')$.

We now check that $\mathcal{A}$ satisfies Points (a) to (d).

Point (a). Clearly all r functional w.r.t. $\mathcal{T}$ are functional in $\mathcal{A}$. Thus, $\mathcal{T}, \mathcal{A} \models q(x)$ iff $\emptyset, \mathcal{A} \models p'(x)$ for some disjunct $p'(x)$ of $q(x)$. It therefore suffices to show that there is no homomorphism from any disjunct $p'(x)$ of $q(x)$ to $\mathcal{A}$ mapping x to x. It has been observed in [Kikot and Zolin, 2013] already that the mapping π from $\mathcal{A}$ to $p(x)$ mapping every variable to itself and every $(y, i) \in V \times \mathbb{N}$ to y is a homomorphism. Using this observation it has been shown that there is no homomorphism from $p(x)$ to $\mathcal{A}$ as the composition of π with such a homomorphism would be a non-surjective homomorphism mapping x to x which contradicts our assumptions. It also follows that there is no homomorphism from any disjunct $p'(x)$ of $q(x)$ distinct from $p(x)$ to $\mathcal{A}$ as the composition of π with such a homomorphism would be a homomorphism from $p'(x)$ to $p(x)$ mapping x to x.

Point (b). Assume there exists no model of $\mathcal{T}$ and $\mathcal{A}$ with $\mathcal{I} \models \neg C(x)$. Then, by compactness, there exists a finite subset $\mathcal{A}'$ of $\mathcal{A}$ such that there exists no model of $\mathcal{T}$ and $\mathcal{A}'$ with $\mathcal{I} \models \neg C(x)$. But then $\mathcal{T}, \mathcal{A}' \models q(x)$ which contradicts Point (a) and the assumption that $(\mathcal{T}, \Sigma_{\mathsf{full}}, C(x))$ is a rewriting of $(\mathcal{T}, \Sigma_{\mathsf{full}}, q(x))$.

Point (c). This is by Lemma 32.

Point (d). It follows directly from [Kikot and Zolin, 2013] that $\mathcal{I}^{\text{ue}}$ contains a homomorphic image of $p(x)$ under a homomorphism mapping x to $\mathfrak{U}_x$. Regard this image as an ABox $\mathcal{A}'$. Then $\mathcal{T}, \mathcal{A}' \models q(\mathfrak{U}_a)$. This finishes the proof if Point 1 holds.

Now suppose that Point 2 does not hold. Thus, there exists no functional path $y_0, \ldots, y_m$ in $p(x)$ with $x_0 = y_0 = y_m$ such that $\{x_0, \ldots, x_{n-1}\} \subseteq \{y_0, \ldots, y_m\}$. Let $p^n = p|_{\text{nFC}_p(x)}$. We observe the following

Claim 1. There exist $V_1 \subseteq \text{nFC}_p(x)$ of the form $V_1 = \text{FC}_{p^n}(x')$ for some x' in $p(x)$ such that for $V_2 := \text{nFC}_q(x) \setminus V_1$ there are $y_1, y_2 \in V_2$ such that there is a path from y_1 to y_2 in V_2 and $z_1, z_2 \in V_1$ such that there are distinct $s_1(y_1, z_1), s_2(y_2, z_2) \in q(x)$ with s_1, s_2 functional w.r.t.$\mathcal{T}$.

For the proof of Claim 1 take the cycle $r_0(x_0, x_1), \ldots, r_{n-1}(x_{n-1}, x_n)$ in $p(x)$. There must exist x_i such that some x_j with $x_i \neq x_j$ is not in $\text{FC}_p(x_i)$. Let $V_1 = \text{FC}_{p^n}(x_i)$. Then we find $x_j \in \text{FC}_{p^n}(x_i)$ such that $x_{j+1} \notin \text{FC}_{p^n}(x_i)$ and we find a path (possibly of length 0) from x_{j+1} to some $x_{j'}$ within V_2 such that there exist r' and $x_{j''} \in V_1$ with $r'(x_{j'}, x_{j''}) \in p(x)$. Then $s_1(y_1, z_1) := r_j^-(x_{j+1}, x_j)$ and $s_2(y_2, z_2) := r'(x_{j'}, x_{j''})$ are as required.

We define an ABox $\mathcal{A}$ with individual names

$$\text{FC}_p(x) \cup (V_1 \times \mathbb{N}) \cup (V_2 \times I)$$

where

$$I = \{(\beta, E) \mid E \subseteq \mathbb{N}, |E| = |\mathcal{S}|, \beta : E \to \mathcal{S} \text{ bijective } \}$$

and

$$\mathcal{S} = \{r(y, z) \in p(x) \mid z \in V_1, y \in V_2\}$$

as follows:

- for all concept names A and variables $y \in \text{FC}_p(x)$: $A(y) \in \mathcal{A}$ iff $A(y)$ is in $p(x)$;
- for all roles r and variables $y, z \in \text{FC}_p(x)$: $r(y, z) \in \mathcal{A}$ iff $r(y, z)$ is in $p(x)$;
- for all concept names A, variables $y \in V_1$, and $i \in \mathbb{N}$: $A(y, i) \in \mathcal{A}$ iff $A(y)$ is in $p(x)$;
- for all concept names A, variables $y \in V_2$, and $i \in I$: $A(y, i) \in \mathcal{A}$ iff $A(y)$ is in $p(x)$;
- for all roles r, variables $y, z \in V_1$ and $i \in \mathbb{N}$: $r((y, i), (z, i)) \in \mathcal{A}$ iff $r(y, z)$ is in $p(x)$;
- for all roles r, variables $y, z \in V_2$ and $i \in I$: $r((y, i), (z, i)) \in \mathcal{A}$ iff $r(y, z)$ is in $p(x)$;
- for all roles r, variables $y \in \text{FC}_p(x)$ and $z \in V_1$, and $i \in \mathbb{N}$: $r(y, (z, i)) \in \mathcal{A}$ iff $r(y, z)$ is in $p(x)$;
- for all roles r, variables $y \in \text{FC}_p(x)$ and $z \in V_2$, and $i \in I$: $r(y, (z, i)) \in \mathcal{A}$ iff $r(y, z)$ is in $p(x)$;
- for all roles r, $(y, (\beta, E)) \in (V_2 \times I)$ and $(z, i) \in V_1 \times \mathbb{N}$: $r((y, (\beta, E)), (z, i)) \in \mathcal{A}$ iff $i \in E$ and $\beta(i) = r(y, z)$.

This construction of the ABox $\mathcal{A}$ achieves the following:

- for all copies V_2' of V_2 and any $r(y, z) \in \mathcal{S}$, there is a copy V_1' of V_1 such that $r(y', z') \in \mathcal{A}$ for the copies y', z' of y and z in V_2' and V_1', respectively;

- for all copies V_2' of V_2 and copies V_1' of V_1 there is at most one atom $r(y,z) \in \mathcal{A}$ with $y \in V_2$ and $z \in V_1$;
- Let $V_2^1, \dots, V_2^n$ be distinct copies of V_2 and $r_1(y_1, z_1), \dots, r_n(y_n, z_n)$ be distinct atoms in $\mathcal{S}$. Then there is a single copy V_1' of V_1 such that $r_1(y_1', z_1^1), \dots, r_n(y_n', z_n^n) \in \mathcal{A}$ for the copies $y_1', \dots, y_n'$ of $y_1, \dots, y_n$ in $V_2^1, \dots, V_2^n$, respectively, and the copies $z_1^1, \dots, z_n^n$ of $z_1, \dots, z_n$ in V_1'.

We first show Point (a) above.

Point (a). $\mathcal{T}, \mathcal{A} \not\models q(a)$.

Proof of Point (a). By construction, all r functional w.r.t. $\mathcal{T}$ are functional in $\mathcal{A}$. Thus, $\mathcal{T}, \mathcal{A} \models q(x)$ iff $\emptyset, \mathcal{A} \models p'(x)$ for some disjunct $p'(x)$ of $q(x)$. It therefore suffices to show that there is no homomorphism from any disjunct $p'(x)$ of $q(x)$ to $\mathcal{A}$ mapping x to x. Consider the mapping

$$\pi : \mathcal{A} \to p(x)$$

mapping every variable in $\text{FC}_p(x)$ to itself and every $(y,i) \in (V_1 \times \mathbb{N}) \cup (V_2 \times I)$ to y. It is easy to see that π is a homomorphism. It also follows that there is no homomorphism from any disjunct $p'(x)$ of $q(x)$ distinct from $p(x)$ to $\mathcal{A}$ as the composition of π with such a homomorphism would be a homomorphism from $p'(x)$ to $p(x)$ mapping x to x. It remains to prove that there is no homomorphism from $p(x)$ to $\mathcal{A}$ mapping x to x. Assume there is such a homomorphism h. Then $\pi \circ h$ is a homomorphism from $p(x)$ to $p(x)$ mapping x to x. We obtain a contradiction if we can show that $\pi \circ h$ is not surjective. To this end assume that $\pi \circ h$ is surjective. As $p(x)$ is finite, it is an isomorphism. Let $h[p(x)] = \{h(y) \mid y \in p(x)\}$ be the image of h in $\mathcal{A}$. Then h is an isomorphism from $p(x)$ onto the restriction $\mathcal{A}|_{h[p(x)]}$ of $p(x)$ to $h[p(x)]$ and the restriction $\pi|_{h[p(x)]}$ of π to $h[p(x)]$ is an isomorphism onto $p(x)$. It follows that $h[p(x)]$ contains for every y in $p(x)$ exactly one individual a with $\pi(a) = y$ and

- $h[p(x)]$ contains $\text{FC}_p(x)$;
- as V_1 is connected, there exists $i \in \mathbb{N}$ such that $h([p(x)]) \supseteq V_1 \times \{i\}$ and $h([p(x)]) \cap V_1 \times \{j\} = \emptyset$ for all $j \neq i$;
- for every connected component V of V_2 of there exists $i \in I$ such that $h([p(x)]) \supseteq V \times \{i\}$ and $h([p(x)]) \cap V \times \{j\} = \emptyset$ for all $j \neq i$.

Now recall that there there are distinct atoms $s_1(y_1, z_1) \in p(x)$ and $s_2(y_2, z_2) \in p(x)$ such that y_1, y_2 are in the same connected component in V_2 and $z_1, z_2 \in V_1$. Thus, there exists $i \in \mathbb{N}$ such that $(y_1, i), (y_2, i) \in h[p(x)]$ and for some $j \in I$, $s_1((y_1, i), (z_1, j)) \in \mathcal{A}$ and $s_2((y_2, i), (z_2, j)) \in \mathcal{A}$. As observed above, no such two atoms exist in $\mathcal{A}$ and we have derived a contradiction. This finishes the proof of Point (a).

Example 34. Let $\mathcal{T} = \{\mathsf{func}(s_1), \mathsf{func}(s_2)\}$ and consider the CQ

$$q(x) = \exists y, z(r(x,y) \wedge s_1(y,z) \wedge s_2(y,z))$$

Then

$$\text{FC}_q(x) = \{x\}, \quad V_1 = \{z\}, \quad V_2 = \{y\}$$

Thus,

$$\mathcal{S} = \{s_1(y,z), s_2(y,z)\}$$

and so the individuals of $\mathcal{A}$ are

$$\{x\} \cup (\{z\} \times \mathbb{N}) \cup (\{y\} \times I)$$

and the essential properties of $\mathcal{A}$ are:

- $r(x, (y,i)) \in \mathcal{A}$ for all $i \in I$;
- for every $(y,i) \in I$ there are distinct $(z, i_1), (z, i_2)$ with $s_1((y,i),(z,i_1)), s_2((y,i),(z,i_2)) \in \mathcal{A}$;
- for any two $(z, i_1), (z, i_2)$ there exists (y,i) such that $s_1((y,i),(z,i_1)), s_2((y,i),(z,i_2)) \in \mathcal{A}$.

Observe that $\mathcal{A} \not\models q(x)$.

Points (b) and (c) are as before. It remains to show Point (d). Let $\mathcal{I}$ be a model of $\mathcal{T}$ and $\mathcal{A}$ with $\mathcal{I} \models \neg C(a)$ and consider the ultrafilter extension $\mathcal{I}^{\text{ue}}$. We define a homomorphism h from $p(x)$ to $\mathcal{I}^{\text{ue}}$ mapping x to $\mathfrak{U}_x$. For every $y \in \text{FC}_p(x)$, we set $h(y) = \mathfrak{U}_y$. To define h for the remaining variables, fix a non-principal ultrafilter $\mathfrak{N}$ over $\mathbb{N}$. For every variable $z \in V_1$ we obtain an ultrafilter $\mathfrak{N}(z)$ over $\Delta^{\mathcal{I}}$ by setting $U \in \mathfrak{N}(z)$ iff $\{i \mid (z,i) \in U \cap (V_1 \times \mathbb{N})\} \in \mathfrak{N}$. Observe that for $z_1, z_2 \in V_1$, $r(z_1, z_2) \in p(x)$ implies $(\mathfrak{N}(z_1), \mathfrak{N}(z_2)) \in r^{\mathcal{I}^{\text{ue}}}$. Observe as well that for $y \in \text{FC}_p(x)$ and $z \in V_1$, $r(y,z) \in p(x)$ implies $(\mathfrak{U}_y, \mathfrak{N}(z)) \in r^{\mathcal{I}^{\text{ue}}}$. We set $h(z) = \mathfrak{N}(z)$ for $z \in V_1$. It remains to define h for variables in V_2.

Let for $y \in V_2$, and $X \subseteq \Delta^{\mathcal{I}}$, $\rho_y(X) = \{i \in I \mid (y,i) \in X\}$. By construction, the set

$$\mathfrak{X} = \{\rho_y(\exists r.Z)^{\mathcal{I}} \mid \ r(y,z) \in \mathcal{S} \text{ and } Z \in \mathfrak{N}(z)\}$$

has the finite intersection property. Thus, there exists an ultrafilter $\mathfrak{I}$ over I containing $\mathfrak{X}$. For every variable $y \in V_2$ we obtain an ultrafilter $\mathfrak{I}(y)$ over $\Delta^{\mathcal{I}}$ by setting $U \in \mathfrak{I}(y)$ iff $\{i \mid (y,i) \in U \cap (V_2 \times I)\} \in \mathfrak{I}$. Observe that for $y_1, y_2 \in V_1$, $r(y_1, y_2) \in p(x)$ implies $(\mathfrak{I}(y_1), \mathfrak{I}(y_2)) \in r^{\mathcal{I}^{\text{ue}}}$. Observe as well that for $y \in \text{FC}_p(x)$ and $y' \in V_2$, $r(y, y') \in p(x)$ implies $(\mathfrak{U}_y, \mathfrak{I}(y')) \in r^{\mathcal{I}^{\text{ue}}}$. Finally, observe that by construction $r(y,z) \in \mathcal{S}$ implies that $(\mathfrak{I}(y), \mathfrak{N}(z)) \in r^{\mathcal{I}^{\text{ue}}}$. We set $h(y) = \mathfrak{I}(y)$ for $y \in V_2$. It follows that h is a homomorphism from $p(x)$ to $\mathcal{I}^{\text{ue}}$, as required.

Now let $\mathcal{A}'$ be the image of $p(x)$ under h, regarded as an ABox. Then $\mathcal{T}, \mathcal{A}' \models q(\mathfrak{U}_x)$, as required.

The proof when $\mathcal{ALCI}$-IQ is replaced with $\mathcal{ALCI}^u$-IQ and connectedness is dropped is a straightforward variation of the proof above. ❑

Theorem 11. *In $(\mathcal{ALCF}, CQ)$, IQ-rewritability is undecidable.*

Proof. We use a reduction from emptiness checking in $(\mathcal{ALCF}, \text{AQ})$ which is known to be undecidable [Baader *et al.*, 2016]. Let $Q = (\mathcal{T}, \Sigma, A(x))$ be an OMQ from this language and let $q(x) = \exists y\, A(x) \wedge r(x,y) \wedge r(y,y)$, where r is a role name that does not occur in $\mathcal{T}$. We show that Q is empty iff $Q' = (\mathcal{T}, \Sigma \cup \{r\}, q(x))$ is IQ-rewritable. Clearly, if Q is empty, then Q' is empty and, therefore, IQ-rewritable. Conversely, if Q is not empty, then we show that

Q' is not IQ-rewritable. To this end we take a Σ-ABox $\mathcal{A}_0$ and $a \in \mathsf{ind}(\mathcal{A})$ such that $\mathcal{A}_0 \models Q(a)$ and $\mathcal{A}_0$ is consistent with $\mathcal{T}$. Assume for a proof by contradiction that there is a rewriting $Q'' = (\mathcal{T}', \Sigma \cup \{r\}, C(x))$ of Q'. We use a minor modification of the construction in the proof of Theorem 13:

(a) Using the CQ $q(x)$ and $\mathcal{A}_0$ we construct an infinite $(\Sigma \cup \{r\})$-ABox $\mathcal{A} \supseteq \mathcal{A}_0$ such that $\mathcal{A} \not\models Q'(a)$;

(b) By compactness of FO there exists a forest model $\mathcal{I}$ of $\mathcal{T}'$ and $\mathcal{A}$ with $\mathcal{I} \models \neg C(a)$;

(c) Then $\mathcal{I}^{\text{ue}} \models \neg C(\mathfrak{U}_a)$ and $\mathcal{I} \models \mathcal{T}'$ by Lemma 32;

(d) Moreover, $\mathcal{I}^{\text{ue}}$ is a model of a finite $(\Sigma \cup \{r\})$-ABox $\mathcal{A}'$ with individual name $\mathfrak{U}_a$ such that $\mathcal{A}' \models Q'(\mathfrak{U}_a)$. But this contradicts $\mathcal{I}^{\text{ue}} \models \neg C(\mathfrak{U}_a)$.

The construction of $\mathcal{A}$ is the same as in Point 1 of the proof of Theorem 13 above using the query $q(x)$ and its cycle $r(y, y)$ except that we also attach $\mathcal{A}_0$ to the ABox constructed by identifying x and a. Now the proof of Points (a) to (d) is exactly as before. ❑

E Proofs for Section 6

We start with definitions of MMSNP and CSP together with some preliminaries. We consider signatures $\mathbf{S}$ that consist of predicate symbols with unrestricted arity, known as *schemas*. An $\mathbf{S}$-*fact* is an expression of the form $S(a_1, \ldots, a_n)$ where $S \in \mathbf{S}$ is an n-ary predicate symbol, and $a_1, \ldots, a_n$ are elements of some fixed, countably infinite set const of *constants*. An $\mathbf{S}$-*instance* I is a set of $\mathbf{S}$-facts. The *domain* of I, denoted $\mathsf{dom}(I)$, is the set of constants that occur in some fact in I. The notions of cycles, girth, acyclicity, and connectedness can be lifted from ABoxes to $\mathbf{S}$-instances, for details see [Feier *et al.*, 2017]. We use CON_I to denote the set of $\mathbf{S}$-instances that are the maximal connected components of the $\mathbf{S}$-instance I.

An *MMSNP sentence* φ *over schema* $\mathbf{S}$ has the form

$$\exists X_1 \cdots \exists X_n \forall x_1 \cdots \forall x_m \, \psi,$$

with $X_1, \ldots X_n$ monadic second-order (SO) variables, $x_1, \ldots, x_m$ first-order (FO) variables, and ψ a conjunction of quantifier-free formulas of the form

$$\alpha_1 \wedge \cdots \wedge \alpha_n \rightarrow \beta_1 \vee \cdots \vee \beta_m \text{ with } n, m \geq 0, \qquad (\dagger)$$

where each α_i is of the form $X_i(x_j)$ or $R(\mathbf{x})$ with R from $\mathbf{S}$, and each β_i is of the form $X_i(x_j)$. We refer to a formula of the form (†) as a *rule* in φ, to the conjunction $\alpha_1 \wedge \cdots \wedge \alpha_n$ as its *body*, and to $\beta_1 \vee \cdots \vee \beta_m$ as its *head*. A rule body can be seen as an $\mathbf{S} \cup \{X_1, \ldots, X_n\}$-instance in the obvious way, which we shall sometimes do implicitly. An MMSNP sentence φ is *connected* if the body of every rule in φ is connected. The *rule size* of φ is the maximum size of a rule in φ.

Every MMSNP sentence φ can be seen as a Boolean query in the obvious way, that is, for an $\mathbf{S}$-instance I, $I \models \varphi$ whenever φ evaluates to true on I. We also consider *disjunctions of MMSNP sentences over schema* $\mathbf{S}$, that is, sentences of the form $\bigvee_i \varphi_i$, where each φ_i is an MMSNP sentence over $\mathbf{S}$. For an $\mathbf{S}$-instance I, $I \models \bigvee_i \varphi_i$, whenever there exists some i such that φ_i evaluates to true on I. An MMSNP sentence φ_1 over $\mathbf{S}$ is *contained* in an MMSNP sentence φ_2 over $\mathbf{S}$, written $\varphi_1 \subseteq \varphi_2$, if for every $\mathbf{S}$-instance I, $I \models \varphi_1$ implies $I \models \varphi_2$. We say that φ_1 and φ_2 are *equivalent* if $\varphi_1 \subseteq \varphi_2$ and $\varphi_2 \subseteq \varphi_1$. Containment and equivalence are defined in the same way for disjunctions of MMSNP sentences.

A *constraint satisfaction problem (CSP)* is defined by an $\mathbf{S}$-instance T that is called the *template*. The problem associated with T is to decide whether an input $\mathbf{S}$-instance I admits a homomorphism to T, denoted $I \rightarrow T$. An MMSNP sentence φ over schema $\mathbf{S}$ is said to be *CSP-definable* if there exists an $\mathbf{S}$-template T such that for every $\mathbf{S}$-instance I, $I \models \varphi$ iff $I \rightarrow T$. A *generalized CSP* over schema $\mathbf{S}$ is defined by a finite set $\mathsf{T} = \{T_1, \ldots T_n\}$ of $\mathbf{S}$-templates. For an $\mathbf{S}$-instance I, we write $I \rightarrow \mathsf{T}$ if $I \rightarrow T_i$ for some i. An MMSNP sentence φ over schema $\mathbf{S}$ is *definable by a generalized CSP* T if for every $\mathbf{S}$-instance I, $I \models \varphi$ iff $I \rightarrow \mathsf{T}$.

For two $\mathbf{S}$-instances I_1 and I_2 with disjoint domains, we use $I_1 \uplus I_2$ to denote the disjoint union of I_1 and I_2.[5] An MMSNP sentence φ is *preserved under disjoint union* if for all $\mathbf{S}$-instances I_1 and I_2 (with disjoint domains), $I_1 \models \varphi$ and $I_2 \models \varphi$ implies $I_1 \uplus I_2 \models \varphi$.

Lemma 35. *An MMSNP sentence φ is CSP-definable iff it is definable by a generalized CSP and preserved under disjoint union.*

Proof. "⇒". Clear.

"⇐". Assume that φ is definable by a generalized CSP $\mathsf{T} = \{T_1, \ldots, T_n\}$ and preserved under disjoint union. For every i, $T_i \rightarrow \mathsf{T}$ and thus $T_i \models \varphi$. Assume w.l.o.g. that the domains of all templates in T are mutually disjoint. As φ is preserved under disjoint union, $\biguplus_{1 \leq i \leq n} T_i \models \varphi$. Thus, $\biguplus_{1 \leq i \leq n} T_i \rightarrow \mathsf{T}$ and consequently $\biguplus_{1 \leq i \leq n} T_i \rightarrow T_j$ for some j. This implies that $T_i \rightarrow T_j$ for every i and thus the generalized CSP T is equivalent to the CSP T_j, which finishes the proof. ❑

We next determine the complexity of deciding preservation under disjoint union for MMSNP sentences. For our final aim, we only need the upper bound, but we also observe a lower bound for the sake of completeness.

Theorem 36. *Deciding whether an MMSNP sentence is preserved under disjoint union is* 2NEXPTIME*-complete.*

Proof. For the upper bound, we reduce preservation under disjoint union to a series of (exponentially many) containment checks between MMSNP sentences (of polynomial size) and then invoke the result from [Bourhis and Lutz, 2016] that MMSNP containment can be decided in 2NEXPTIME.

Let φ be an MMSNP sentence over schema $\mathbf{S}$ and $\mathcal{N}$ the set of nullary predicate symbols in φ. We assume w.l.o.g. that the number of first-order variables in φ is bounded from below by the largest arity of a predicate in $\mathbf{S}$. For all $\mathcal{N}_1, \mathcal{N}_2 \subseteq \mathcal{N}$, we construct an MMSNP sentence $\varphi_{\mathcal{N}_1,\mathcal{N}_2}$ as follows:

- $\varphi_{\mathcal{N}_1,\mathcal{N}_2}$ has the same quantifiers as φ except for two new existentially quantified second-order variables C_1 and C_2;

[5] We do not assume here that I_1 and I_2 contain the same nullary predicate symbols; $I_1 \uplus I_2$ contains the union of the nullary symbols in I_1 and I_2.

- $\varphi_{\mathcal{N}_1,\mathcal{N}_2}$ has the following rules:
  - $\mathsf{true} \to C_1(x) \vee C_2(x)$;
  - $C_1(x) \wedge C_2(x) \to \mathsf{false}$;
  - $R(y_1,\dots,y_n) \wedge C_i(y_j) \to C_i(y_k)$ whenever $R \in \mathbf{S}$ is n-ary, $i \in \{0,1\}$, and $j,k \in \{1,\dots,n\}$ and where $y_1,\dots,y_n$ are the first n FO variables in φ;
  - $C_i(x_1) \wedge \cdots \wedge C_i(x_n) \wedge \mathsf{body} \to \mathsf{head}$ whenever $\mathsf{body} \to \mathsf{head}$ is a rule in φ with no predicate symbol from $\mathcal{N} \setminus \mathcal{N}_i$ occuring in body and $i \in \{0,1\}$ and where $x_1,\dots,x_n$ are the FO variables in body.

Intuitively, an $\mathbf{S}$-instance I satisfies $\varphi_{\mathcal{N}_1,\mathcal{N}_2}$ iff there is a coloring of I with the two colors C_1 and C_2 such that elements from the same maximal connected components receive the same color and each of the resulting two monochromatic subinstances of I satisfies φ. Note that I is the disjoint union of I_1 and I_2. The sets $\mathcal{N}_1, \mathcal{N}_2$ help to disentangle the nullary predicate symbols: $\mathcal{N}_1$ contains the predicates true in the monochromatic subinstance colored C_1 and likewise for $\mathcal{N}_2$ and C_2.

Claim. *φ is preserved under disjoint union iff $\varphi \equiv \bigvee_{\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}} \varphi_{\mathcal{N}_1,\mathcal{N}_2}$.*

"⇒'. Assume that φ is preserved under disjoint union. We have to show the following inclusions:

- $\varphi \subseteq \bigvee_{\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}} \varphi_{\mathcal{N}_1,\mathcal{N}_2}$. Let $I \models \varphi$. Further let I_1 be the extension of I in which every element is colored C_1 and let $\mathcal{N}_1$ be the set of nullary predicates true in I and $\mathcal{N}_2 = \emptyset$. Clearly, $I_1 \models \varphi_{\mathcal{N}_1,\mathcal{N}_2}$, witnessing $I \models \bigvee_{\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}} \varphi_{\mathcal{N}_1,\mathcal{N}_2}$.
- $\bigvee_{\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}} \varphi_{\mathcal{N}_1,\mathcal{N}_2} \subseteq \varphi$. Let $I \models \bigvee_{\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}} \varphi_{\mathcal{N}_1,\mathcal{N}_2}$. Then $I \models \varphi_{\mathcal{N}_1,\mathcal{N}_2}$ for some $\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}$. By construction of $\varphi_{\mathcal{N}_1,\mathcal{N}_2}$, there is thus a partition of $\mathsf{dom}(I)$ into two sets S_1, S_2 such that $I_1 \models \varphi$ and $I_2 \models \varphi$ where I_i is the restriction of I to domain S_i and makes exactly the nullary predicates in $\mathcal{N}_i$ true, $i \in \{1,2\}$. As I is the disjoint union of I_1 and I_2, $I \models \varphi$.

"⇐". Assume that $\varphi \equiv \bigvee_{\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}} \varphi_{\mathcal{N}_1,\mathcal{N}_2}$. Let I_1 and I_2 be $\mathbf{S}$-instances with disjoint domain such that $I_1 \models \varphi$ and $I_2 \models \varphi$. Then $I_1 \uplus I_2 \models \varphi_{\mathcal{N}_1,\mathcal{N}_2}$, where $\mathcal{N}_i$ is the set of nullary predicates true in I_i, $i \in \{1,2\}$. Thus, $I_1 \uplus I_2 \models \bigvee_{\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}} \varphi_{\mathcal{N}_1,\mathcal{N}_2}$ and from the original assumption $I_1 \uplus I_2 \models \varphi$. This finishes the proof of the claim.

It remains to note that the inclusion $\varphi \subseteq \bigvee_{\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}} \varphi_{\mathcal{N}_1,\mathcal{N}_2}$ holds even when φ is not preserved under disjoint union (as shown by the proof above) and thus deciding whether φ is preserved under disjoint union amounts to checking that $\varphi \supseteq \varphi_{\mathcal{N}_1,\mathcal{N}_2}$ for all $\mathcal{N}_1,\mathcal{N}_2 \subseteq \mathcal{N}$. This gives the desired upper bound.

For the lower bound, we consider the (polynomial time) reduction from [Bourhis and Lutz, 2016] from a 2NEXPTIME-hard torus tiling problem to OMQ containment: there, two OMQs Q_1 and Q_2 are constructed, Q_1 from $(\mathcal{ALCI}, \text{BAQ})$ and Q_2 from $(\mathcal{ALCI}, \text{BCQ})$, such that a double exponentially large torus can be tiled iff $Q_1 \subseteq Q_2$. We sketch a polynomial time reduction from the containment problem for two such OMQs to the preservation under disjoint union of an MMSNP sentence φ. For the sake of proving the correctness of our reduction, we note that Q_1 and Q_2 are such that $Q_2 \not\subseteq Q_1$.

It was shown in [Bienvenu *et al.*, 2014] that every OMQ from $(\mathcal{ALCI}, \text{BAQ})$ such as Q_1 is equivalent to the complement of a CSP T in the sense that for every Σ-ABox $\mathcal{A}$, $\mathcal{A} \models Q$ iff $\mathcal{A} \not\to \mathcal{B}$ where Σ is the signature of the OMQs Q_1 and Q_2. Note that this is a variation of Lemma 21 for the case of BAQs. It has also been shown in [Bienvenu *et al.*, 2014] that for every OMQs from $(\mathcal{ALCI}, \text{BCQ})$, one can construct in polynomial time an MMSNP sentence whose complement is equivalent to the OMQ and thus for Q_1 and Q_2 we find two such sentences φ_1 and φ_2. The size of φ_1 and φ_2 is polynomial in that of Q_1 and Q_2. Summing up, the 2-exp torus can be tiled iff $\varphi_2 \subseteq \varphi_1$, φ_1 is equivalent to a CSP, and $\varphi_1 \not\subseteq \varphi_2$. We next construct an MMSNP sentence φ such that for all $\mathbf{S}$-instances I,

$$I \not\models \varphi \text{ iff } I \not\models \varphi_1 \text{ and } I \not\models \varphi_2. \qquad (\ddagger)$$

Towards constructing φ, we start by standardizing apart all FO and SO variables from φ_1 and φ_2. For each $i \in \{1,2\}$, let ψ_i^- be the quantifier-free part of φ_i with all rules of the form $\mathsf{body} \to \mathsf{false}$ removed. Then φ is the MMSNP sentence which has as SO/FO variables the union of SO/FO variables from φ_1 and φ_2 and the following rules:

- all rules from ψ_1^- and from ψ_2^-,
- all rules of the form $\mathsf{body}_1 \wedge \mathsf{body}_2 \to \mathsf{false}$, where $\mathsf{body}_1 \to \mathsf{false}$ is a rule in φ_1 and $\mathsf{body}_2 \to \mathsf{false}$ a rule in φ_2.

It can be verified that φ satisfies (‡).

Claim. *$\varphi_2 \subseteq \varphi_1$ iff φ is preserved under disjoint union.*

"⇒". Assume that $\varphi_2 \subseteq \varphi_1$. Then, φ is equivalent to φ_1 and thus to a CSP, consequently it is preserved under disjoint union.

"⇐". Assume that $\varphi_2 \not\subseteq \varphi_1$. Thus there exist instances I_1 and I_2 such that $I_1 \models \varphi_1$, $I_1 \not\models \varphi_2$, $I_2 \models \varphi_2$, $I_2 \not\models \varphi_1$. From (‡), we obtain $I_1 \models \varphi$ and $I_2 \models \varphi$. We next observe that $I_1 \not\models \varphi_1$ and $I_2 \not\models \varphi_2$ implies $I_1 \uplus I_2 \not\models \varphi_1$ and $I_1 \uplus I_2 \not\models \varphi_2$. Thus $I_1 \uplus I_2 \not\models \varphi$ by (‡). Consequently, I_1 and I_2 witness that φ is not preserved under disjoint union. ❑

We next characterize the equivalence of MMSNP sentences to a generalized CSP and analyze the complexity of deciding this property.

For an MMSNP sentence φ, let φ_{acyc} be the MMSNP sentence with the same quantifier prefix that contains all rules which have an acyclic body and can be obtained from a rule in φ by zero or more identifications of variables.

Theorem 37. *An MMSNP sentence φ is definable by a generalized CSP iff $\varphi \equiv \varphi_{\mathsf{acyc}}$.*

Proof. "only if". Assume that φ is definable by a generalized CSP $\mathsf{T} = \{T_1,\dots,T_n\}$. Using the construction of φ_{acyc}, it can be verified that $\varphi \subseteq \varphi_{\mathsf{acyc}}$. It thus remains to be shown that $\varphi_{\mathsf{acyc}} \subseteq \varphi$. If the body of each rule in φ is acyclic, then this is clearly the case. Otherwise, let g be the maximum girth of a cyclic rule body from φ. Take an $\mathbf{S}$-instance I such that $I \not\models \varphi$. We have to show that $I \not\models \varphi_{\mathsf{acyc}}$. Since φ is equivalent to T, $I \not\rightarrow T_i$ for $1 \leq i \leq n$. From Lemma 19[6], we obtain an $\mathbf{S}$-instance I^g of girth exceeding g such that $I^g \rightarrow I$ and $I^g \not\rightarrow T_i$ for $1 \leq i \leq n$. Thus $I^g \not\models \varphi$. As the girth of I^g is higher than the girth of every cyclic rule body in φ, it follows that $I^g \not\models \varphi_{\mathsf{acyc}}$. Since $I^g \rightarrow I$, $I \not\models \varphi_{\mathsf{acyc}}$.

"if". Assume that $\varphi \equiv \varphi_{\mathsf{acyc}}$. Since the rule bodies in φ_{acyc} are acyclic, it is easy to convert φ_{acyc} into an equivalent MMSNP sentence in which each rule body contains at most one atom that uses a predicate symbol from $\mathbf{S}$; see [Feder and Vardi, 1998]. It is implicit in that paper (see also [Bienvenu *et al.*, 2014]) that MMSNP sentences of this kind have the same expressive power as generalized CSPs. Thus, φ is equivalent to a generalized CSP. ❏

Before showing the main complexity result of this section, we state a slightly refined version of a theorem from [Bourhis and Lutz, 2016] regarding the complexity of MMSNP containment. It emphasizes that the complexity of containment is double exponential only in the size of the rules, but not in their number. This only requires a careful analysis of the constructions in [Bourhis and Lutz, 2016].

Theorem 38. *Containment between MMSNP sentences is in* 2NExpTime. *More precisely, for MMSNP sentences φ_1 and φ_2 where φ_i has n_i rules and rule size r_i, $i \in \{1,2\}$, it can be decided in time* $2^{2^{p(\log n_1 + r_1 + \log\log n_2 + \log r_2)}}$.

Theorem 15. *It is 2*NExpTime*-complete to decide whether a given MMSNP-sentence is equivalent to a CSP.*

Proof. We start with the upper bound. Lemma 35 and Theorem 37 suggest an algorithm for deciding CSP-definability of an MMSNP sentence φ: check whether φ is preserved under disjoint union and $\varphi \equiv \varphi_{\mathsf{acyc}}$. The first condition can be decided in 2NExpTime according to Theorem 36. As for the second check, we note that the size of φ_{acyc} might be exponential in the size of φ so we cannot apply the MMSNP containment result from [Bourhis and Lutz, 2016] straightaway. However, the rule size of φ_{acyc} is polynomial in the size of φ and thus by Theorem 38 the second condition can be decided in 2NExpTime as well.

For showing that CSP-definability of MMSNP sentences is 2NExpTime-hard, we can apply the same reduction as in the proof of Theorem 36: the MMSNP sentences φ_1, φ_2, and φ, constructed in the reduction are such that $\varphi_2 \subseteq \varphi_1$ iff φ is equivalent to a CSP. ❏

[6]Note that the lemma in its original formulation in [Feder and Vardi, 1998] applies to instances over schemas of any arity, not just to ABoxes.