FO Rewritability for OMQ using Beth Definability and Interpolation

David Toman and Grant Weddell

Cheriton School of CS, University of Waterloo, Canada
{david,gweddel}@uwaterloo.ca

Abstract. We study first-order (FO) rewritability for query answering in ontology mediated querying (OMQ) in which ontologies are formulated in Horn fragments of description logics (DLs). In general, OMQ approaches for such logics rely on non-FO rewriting of the query or on analogous completion of the data (ABox). In this paper, we study the problem of the existence of FO rewritings in terms of Beth definability, and show how Craig interpolation can then be used to effectively construct the rewritings, when they exist, from the Clark's completion of Datalog-like programs encoding a given DL TBox and optionally a query. We show how our construction can also be seen as an alternative to deriving perfect rewritings in the DL-Lite setting.

1 Introduction

We study *first-order (FO) rewritability* for query answering in the setting of *ontology mediated querying* (OMQ) over a *knowledge base* (KB) formulated in terms of underlying Horn *description logics* (DLs) in the $\mathcal{ALC}$ family.

Typical OMQ approaches generally rely on either reformulating the query by incorporating the KB's terminological knowledge [10,11] and then executing the reformulated query over the explicit data in the KB as a relational query, or, for more expressive logics, on a Datalog completion of the explicit data with respect to the KB's terminological knowledge over which the OMQ is answered [24,25,27,28]. In the latter case, data completion is sometimes expressible in first-order logic. This raises the *FO rewritability* problem: determining if *a particular OMQ instance* can be equivalently expressed as an FO query over the explicit data in the knowledge base.

Earlier work on OMQ for the FunDL family of DLs [31,32,34] has presented what was called a *combined combined approach* to OMQ, and has shown that it is essential to preserve tractability of OMQ in the presence of (limited) value restrictions. In this paper, we focus on the FO rewritability of OMQ when the underlying DL is Horn-$\mathcal{SHIQ}$ and its variants. We show that an adaptation of the combined combined approach leads to efficient OMQ query answering and to a solution to the FO rewritability problem for this family of DLs. In particular, we show how the combined combined approach, with the help of *Beth definability*

[4] applied on the *Clark's completion* [13] of the Datalog program used for the completion of the explicit data in the knowledge base, can be used to characterize FO rewritability of OMQs. We also show how *Craig interpolation* [14] can then be used to construct such an FO rewriting, when it exists. The existence of such a rewriting enables an OMQ front-end to a relational data source that underlies an ABox to operate entirely by a more refined query reformulation of a given *union of conjunctive queries* (UCQ) that yields an SQL query over the relational data source, with no requirement to update tables beforehand.

Our contributions are as follows.

1. We show how to decide uniform FO rewritability of OMQ in Horn-$\mathcal{SHIQ}$ via Clark's completion of Datalog programs and Beth definability;
2. We show how our framework extends to *query specific OMQ* by extending existing results for Horn-$\mathcal{DLFD}$; and
3. We show how a variant of the perfect rewriting approach to OMQ can be synthesized by appeal again to Beth definability and Craig interpolation.

This paper builds on earlier work that was the first to consider FO rewritability of OMQ, but for the above mentioned FunDL family of DLs, via Beth definability and Clark's completion [36]. FO rewritability for Horn logics in the $\mathcal{ALC}$ family has been studied by others, e.g., see [5,7]. This other work has also developed algorithms for generating such rewritings efficiently for logics in the $\mathcal{EL}$ family [19]. Our approach seems to provide an alternative path to detecting rewritability and to generating rewritings. A feature of our approach is its link to interpolation-based query optimization [21,33]. The link to query optimization reveals that minimal sized rewritings are often not optimal for query execution. However, establishing limits on the size of rewriting [6] does provide a guide on what rewritings are reasonable to consider during query optimization.

The use of database constraints, possibly combined with constraints implied by data mapping rules, has been explored in several systems that implement variants of perfect rewriting [11], such as Ontop and MASTRO [3,9] and others. One of the contributions in [36], inherited by this work, also shows how Beth definability and Clark's completion seamlessly accommodate such constraints into the rewriting via interpolation (for space reasons we do not present the details here).

Beth definability and Craig Interpolation have been used for other purposes, such as query reformulation under FO constraints [8,21,33,35]. That use, however, is orthogonal to the topic of this paper.

The remainder of the paper is organized as follows. Section 2 provides the necessary background and definitions. Here, we review Horn-$\mathcal{SHIQ}$ and the combined combined approach to OMQ. Our main results then follow in Section 3 in which we show how the above-mentioned artifacts, Clark's completion of Datalog programs for example, can be employed to both decide FO rewritability and to synthesize FO rewritings of ABox completion in the combined combined approach to OMQ. In Section 4, we show how our framework is an alternative approach to perfect rewriting of queries for knowledge bases formulated in DL-Lite. In the conclusions we discuss several limitations and possible extensions of our approach.

2 Background and Definitions

Horn-$\mathcal{SHIQ}$ and its variants. Our primary focus will be on KBs with underlying DLs that are a variant of Horn-$\mathcal{SHIQ}$. We begin by defining the relevant roles, concepts and their semantics presumed by such DLs.

Definition 1 (Horn-$\mathcal{SHIQ}$ Concepts and Roles)
Let R, PC and IN be disjoint sets of primitive role names, primitive concept names and individual names respectively. Horn-$\mathcal{SHIQ}$ roles R are of the form P and P^- for $P \in \mathsf{R}$, and concepts C are of the form A for $A \in \mathsf{PC}$, $C_1 \sqcap C_2$, $\bot$, $\top$, $\forall R.C$, $\exists R.C$, $(\geq n\ R.C)$, or $(\leq n\ R.C)$ for $n \geq 0$. The semantics is with respect to a structure $\mathcal{I} = (\triangle^{\mathcal{I}}, \cdot^{\mathcal{I}})$ in which $\triangle^{\mathcal{I}}$ is a domain of objects and $\cdot^{\mathcal{I}}$ an interpretation function seeded by fixing the interpretations of primitive concept names A to be subsets of $\triangle^{\mathcal{I}}$, primitive role names R to be subsets of $\triangle^{\mathcal{I}} \times \triangle^{\mathcal{I}}$, and individual names a to be elements of $\triangle^{\mathcal{I}}$, and is extended to derived concepts C and roles R in the standard way [2]. Subsumption between concepts and roles, assertions, knowledge bases and their consistency, logical implication, and other reasoning problems are also defined in the standard way.

The following definition of a Horn-$\mathcal{SHIQ}$ KB appeals to a simplified normal form for subsumption constraints presented in [16]. For more general but expressively equivalent syntax, e.g., that allows other forms of qualified number restrictions, see [22].

Definition 2 (Horn-$\mathcal{SHIQ}$ TBoxes and ABoxes [16])
A Horn-$\mathcal{SHIQ}$ knowledge base *$\mathcal{K}$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. A TBox $\mathcal{T}$ (in normal form) consists of role subsumptions of the form $R_1 \sqsubseteq R_2$ that define a role hierarchy, transitivity assertions* $\mathsf{trans}(R)$*, and concept subsumptions that adhere to one of the following forms:*[1]

$$\begin{array}{l} A \sqcap B \sqsubseteq C, \\ A \sqsubseteq \forall R.B, \\ A \sqsubseteq \exists R.B, \\ \exists R.A \sqsubseteq B, \textit{ or} \\ A \sqsubseteq (\leq 1\ R.B), \end{array}$$

where $A, B, C \in \mathsf{PC} \cup \{\top, \bot\}$. Roles R are called simple *when neither they nor any of their subroles are transitive. To avoid a well known source of undecidability, we require that any number restriction occurring in $\mathcal{T}$ will mention only simple roles.*

An ABox $\mathcal{A}$ consists of concept assertions, role assertions, equality axioms and inequality axioms with the respective forms $A(a)$, $R(a, b)$, $a = b$ and $a \neq b$.

A Horn-$\mathcal{ALCHQI}$ KB is a Horn-$\mathcal{SHIQ}$ KB without any transitivity assertions.

[1] Note that subsumptions of the form "$A_1 \sqcap \cdots \sqcap A_n \sqsubseteq B$" are also allowed in [16]. Here, we are appealing to an obvious conservative extension to replace such subsumptions with strictly binary use of conjunction to further simplify our presentation.

Conjunctive queries and OMQ. Conjunctive queries are, as usual, formed from atomic queries (or *atoms*) of the form “$A(x)$” and “$R(x, y)$”, where x and y are variables, using conjunction and existential quantification (in prenex normal form). As usual, we conflate conjunctive queries with the set of its constituent atoms and a list of *answer variables* to simplify notation.

Definition 3 (Conjunctive Query) *Let ψ now be a set of atoms $A(x_i)$ and $R(x_{i_1}, x_{i_2})$, where A is a primitive concept name or $\top$, R a role name, and $\bar{x}$ a tuple of variables. We call the expression $\varphi = \{\bar{x} \mid \psi\}$ a* conjunctive query *(CQ).*

A CQ φ is also a notational variant of the formula “$\exists \bar{y}. \bigwedge_{\phi \in \psi} \phi$” in which $\bar{y}$ contains all variables appearing in ψ but not in $\bar{x}$.[2] We also omit set braces when explicitly listing atoms in ψ to improve readability. With this understanding, the usual definition of certain answers is assumed and given as follows:

Definition 4 (Certain Answer) *Let $\mathcal{K}$ be a Horn-$\mathcal{SHIQ}$ knowledge base and $\varphi = \{\bar{x} \mid \psi\}$ a CQ. A* certain answer *to φ over $\mathcal{K}$ is a tuple of constant symbols $\bar{a}$, such that $\mathcal{K} \models \varphi(\bar{a})$ (where $\varphi(\bar{a})$ is short for $\varphi[\bar{x} \mapsto \bar{a}]$).*

Our primary concern is then given by the following problem:

Definition 5 ((uniform) FO Query Rewritability)
Given a Horn-$\mathcal{SHIQ}$ TBox $\mathcal{T}$, the problem of uniform query rewritability *is to determine if there is a query reformulation $\varphi_{\mathcal{T}}$ for* every *CQ φ such that, for every ABox $\mathcal{A}$ and tuple of constant symbols $\bar{a}$, $(\mathcal{T}, \mathcal{A}) \models \varphi(\bar{a})$ iff $\mathcal{A} \models \varphi_{\mathcal{T}}(\bar{a})$.*

Later in the paper, we briefly consider a query-specific variant of this problem: whether such a rewriting exists for a given CQ. The following observations will also be useful in regard to this problem.

Observation 6 (Transitivity) *Consider a Horn-$\mathcal{SHIQ}$ knowledge base with a TBox $\{\mathsf{trans}(R)\}$. Then the CQ $\{(x, y) \mid R(x, y)\}$ cannot be FO rewritable since this would one allow to answer the* connectivity *question with respect to any ABox considered as a graph of R-edges.*

Analogously to transitive roles, allowing equality between ABox objects, and therefore not adopting the *unique name assumption* (UNA), leads immediate to non-rewritability:

Observation 7 (Equality) *Consider a Horn-$\mathcal{SHIQ}$ KB in which $\mathcal{T} = \emptyset$ and a CQ $\{(x, x) \mid \top(x)\}$. Again, this query solves the (undirected) connectivity problem in an ABox with explicit equalities between individuals and thus cannot have an FO rewriting.*

[2] Note that it is not necessary to place any restrictions on the variables $\bar{x}$. Indeed, one can add additional atoms $\top(x_i)$ to ensure variables in $\bar{x}$ also appear in ψ, if desired, without any impact on the remaining results.

Hence, hereon, we focus on the Horn-$\mathcal{ALCHQI}$ sub-dialect of Horn-$\mathcal{SHIQ}$ without transitive roles, and also adopt UNA.

Observation 8 (Boolean Queries) *Consider a Boolean CQ φ. Such a query, when equivalent to a concept, can be entailed not only due to* matches *in an ABox, but also due to matches in the* anonymous *part of the models of the knowledge base. However, these matches only depend on the existence of certain patterns (types) in the given ABox that can be enumerated and this way converted to a UCQ [31,32,34].*

Hence, hereon as well, we focus on CQs with (1) at least one answer variable, and (2) that are *connected.* The combined combined approach that we now outline can be extended to all CQs, although the details of doing so lie outside the scope of this paper since they do not affect query rewritability. Finally, we assume that the KBs under consideration are consistent. Hence, it will be unnecessary to check for *at most number restrictions* (functionality) in our constructions.

The Combined Combined Approach. To study FO rewritability of conjunctive queries over Horn-$\mathcal{ALCHQI}$ knowledge bases, we begin with the following manifestation of a combined combined approach to OMQ originally developed for the feature logic Horn-$\mathcal{DLFD}$ [31,32,34].[3] Our objective is to modify the approach to suit Horn-$\mathcal{ALCHQI}$, and to show how such can be used to decide query rewritability with respect to knowledge bases expressed in terms of such dialects.

Proposition 9 (Combined Combined Approach for Horn-$\mathcal{ALCHQI}$) *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a consistent Horn-$\mathcal{ALCHQI}$ knowledge base and φ a conjunctive query. Then there is a UCQ query $\varphi_{\mathcal{T}}$ and a Datalog program $\Pi_{\mathcal{T}}$, both of which can be effectively constructed from $\mathcal{T}$, such that*

$$\mathcal{K} \models \varphi(\bar{a}) \iff \Pi_{\mathcal{T}}(\mathcal{A}) \models \varphi_{\mathcal{T}}(\bar{a})$$

for any tuple of constant symbols $\bar{a}$, and where $\Pi_{\mathcal{T}}(\mathcal{A})$ is the minimal model of $\Pi_{\mathcal{T}}$ when evaluated over $\mathcal{A}$.

In the rest of this section, we give the definition of $\Pi_{\mathcal{T}}(\mathcal{A})$ and $\varphi_{\mathcal{T}}$.

Datalog programs and clauses that follow use the standard syntax and semantics, and, in particular, predicates used in such programs are classified as either EDB (extensional predicates), those for which we have explicit data, and IDB (intensional predicates), predicates whose interpretation is defined by the *minimal model semantics* of Datalog [37,38,39].

[3] Note that the original *combined approach* [24,29] used the TBox subsumptions to complete the ABox but not to rewrite the CQ. The approach presented here *combines* this combined approach with a variation on *perfect rewriting* [11]; hence we call it the *combined combined approach.*

Definition 10 (Datalog Program $\mathit{\Pi}_{\mathcal{T}}$) The Datalog program $\mathit{\Pi}_{\mathcal{T}}$ used in Proposition 9 consists of completion rules obtained by translating subsumptions that are logical consequences of $\mathcal{T}$. The form of these subsumptions and their translation are given as follows:

(consequences of $\mathcal{T}$)	(completion rule in $\mathit{\Pi}_{\mathcal{T}}$)
$\mathrm{A}_1 \sqcap \mathrm{A}_2 \sqsubseteq \mathrm{B}$	$\mathsf{C}_B(x) \leftarrow \mathsf{C}_{A_1}(x), \mathsf{C}_{A_2}(x)$
$\mathrm{A} \sqsubseteq \forall R.\mathrm{B}$	$\mathsf{C}_B(x) \leftarrow \mathsf{C}_A(y), \mathsf{R}_R(y,x)$
$\exists R.\mathrm{A} \sqsubseteq \mathrm{B}$	$\mathsf{C}_B(x) \leftarrow \mathsf{C}_A(y), \mathsf{R}_R(x,y)$
$R \sqsubseteq S$	$\mathsf{R}_S(x,y) \leftarrow \mathsf{R}_R(x,y)$

For every primitive concept B and role R, we introduce an unary EDB predicates $\mathsf{P}_B(x)$ and $\mathsf{P}_R(x,y)$ together with additional clauses $\mathsf{C}_B(x) \leftarrow \mathsf{P}_B(x)$, $\mathsf{R}_R(x,y) \leftarrow \mathsf{P}_R(x,y)$, and $\mathsf{R}_{R^-}(x,y) \leftarrow \mathsf{P}_R(y,x)$ (accounting for *explicit* data of the form $A(a)$ and $R(a,b)$ in an ABox), and IDB predicates $\mathsf{C}_B(x)$ and $\mathsf{R}_R(x,y)$ corresponding to the *completion* of the ABox w.r.t. $\mathcal{T}$.

Note that the subsumptions for existential restrictions do not contribute to the definition of an ABox completion since they do not generate additional ABox assertions for ABox individuals. Similarly, number restrictions (at most restrictions) do not play any role here as we assume that ABoxes we try to *complete* are always consistent with the TBox $\mathcal{T}$ (as we already mentioned above). Also note that, unlike in the classical combined approach [24,28], our combined combined approach does *not* introduce any new constants in the ABox, similarly to [32]. The following definition is an adaptation of the approach to query reformulation to Horn-$\mathcal{ALCHQI}$:

Definition 11 (Query Reformulation) *Let $\varphi = \{\bar{x} \mid \psi\}$ be a CQ. We write $\mathsf{Fold}_{\mathcal{T}}(\varphi)$ to denote the set of CQs obtained by applying the following when initialized with the singleton set*

$$\{\{\bar{x} \mid \psi\}\}.$$

(computing $\varphi_{\mathcal{T}}$) Update $\mathsf{Fold}_{\mathcal{T}}(\varphi)$ for a CQ $\{\bar{y} \mid \psi\} \in \mathsf{Fold}_{\mathcal{T}}(\varphi)$ according to the following rewrite rules (top-down) until no such rewrite is possible:

1. *If $\{\mathrm{A}(x), \mathrm{B}(x)\} \subseteq \psi$ and $\mathcal{T} \models \mathrm{A} \sqcap B \sqsubseteq \bot$ then*
$$\mathsf{Fold}_{\mathcal{T}}(\varphi) := \mathsf{Fold}_{\mathcal{T}}(\varphi) - \{\{\bar{y} \mid \psi\}\}.$$
2. *If $\{\mathrm{A}(x), \mathrm{B}(x)\} \subseteq \psi$ and $\mathcal{T} \models \mathrm{A} \sqsubseteq \mathrm{B}$ then*
$$\mathsf{Fold}_{\mathcal{T}}(\varphi) := \mathsf{Fold}_{\mathcal{T}}(\varphi) - \{\{\bar{y} \mid \psi\}\} \cup \{\{\bar{y} \mid \psi - \{\mathrm{B}(x)\}\}\}.$$
3. *If $\{\mathrm{R}(x,y), \mathrm{R}'(x,y)\} \subseteq \psi$ and $\mathcal{T} \models \mathrm{R} \sqsubseteq \mathrm{R}'$ then*
$$\mathsf{Fold}_{\mathcal{T}}(\varphi) := \mathsf{Fold}_{\mathcal{T}}(\varphi) - \{\{\bar{y} \mid \psi\}\} \cup \{\{\bar{y} \mid \psi - \{\mathrm{R}'(x,y)\}\}\}.$$
4. *If x and y are variables in ψ then*
$$\mathsf{Fold}_{\mathcal{T}}(\varphi) := \mathsf{Fold}_{\mathcal{T}}(\varphi) \cup \{\{\bar{y} \mid \psi\}[x/y]\}.$$

5. *If* $\{R_1(x,y),\ldots,R_n(x,y),R'_1(y,x),\ldots,R'_m(y,x),\mathrm{A}_1(y),\ldots,\mathrm{A}_k(y)\}\subseteq\psi$ *and* y *does not appear elsewhere in* ψ *nor in* $\bar{y}$ *then*

$$\mathsf{Fold}_{\mathcal{T}}(\varphi) := \mathsf{Fold}_{\mathcal{T}}(\varphi)\cup\{\{\bar{y}\mid\psi'\}\}$$

for all ψ' *of the form*

$$\psi-\{R_1(x,y),\ldots,R_n(x,y),R'_1(y,x),\ldots,R'_m(y,x),\mathrm{A}_1(y),\ldots,\mathrm{A}_k(y)\}\\ \cup\{\mathrm{B}_{1,i_1}(x),\ldots,\mathrm{B}_{k,i_k}(x)\}$$

that are generated by sets $\{\mathrm{B}_{1,i_1}(x),\ldots,\mathrm{B}_{k,i_k}(x)\}$ *such that (i) for some* B_{i,i_j} *and* $\mathrm{B}\in\mathsf{PC}\cup\{\top\}$ *we have* $\mathcal{T}\models\mathrm{B}_{i,i_j}\sqsubseteq\exists R_i.B$ *or* $\mathcal{T}\models\mathrm{B}_{i,i_j}\sqsubseteq\exists R_i'^{-}.B$ *and (ii) such that each of the* A_i *concepts for which* $\mathcal{T}\not\models\mathrm{B}\sqsubseteq\mathrm{A}_i$ *there is a concept* B_{i,i_j} *for which* $\mathcal{T}\models\mathrm{B}_{i,i_j}\sqsubseteq\forall R_i.\mathrm{A}_i$ *or* $\mathcal{T}\models\mathrm{B}_{i,i_j}\sqsubseteq\forall R_j'^{-}.\mathrm{A}_i$*, where* B_{i,i_j} *is maximal w.r.t.* $\sqsubseteq$*.*[4]

The reformulation of φ *w.r.t.* $\mathcal{T}$*,* $\varphi_{\mathcal{T}}$*, is then given by the UCQ* $\bigvee_{\psi\in\mathsf{Fold}(\varphi)}\psi$*.*

Note that, for our purposes, the *existence* of $\varphi_{\mathcal{T}}$ is sufficient since we are concerned with rewritability and not query answering. The existence of $\varphi_{\mathcal{T}}$ also indicates that the non-rewritability of CQs is confined to the interaction of the TBox with explicit data given by an ABox.

3 Classification of TBoxes

To test for *FO* definability of the completion (i.e., all predicates that stand for the completed ABox instance), we use the following construction:

Definition 12 (Clark's Completion $\Sigma_{\mathcal{T}}$) *The* Clark's Completion [13] $\Sigma_{\mathcal{T}}$ *of* $\Pi_{\mathcal{T}}$ *is given by a set of formulas*

$$\mathsf{C}_B(x)\leftrightarrow\mathsf{P}_B(x)\vee(\exists y.\alpha_1)\vee\ldots\vee(\exists y.\alpha_n)\\ \mathsf{R}_R(x,y)\leftrightarrow\mathsf{P}_R(x,y)\vee\beta_1\vee\ldots\vee\beta_m$$

corresponding to all clauses $\mathsf{C}_B(x)\leftarrow\alpha_i$ *and* $\mathsf{R}_R(x,y)\leftarrow\beta_j$ *in* $\Pi_{\mathcal{T}}$ *(grouped by their heads).*

The *bodies* α_i (β_i) are introduced in Definition 10. Note also that the *Clark's Completion* is no longer a Datalog program. This completion, however, closes the original Datalog program in the following sense:

Proposition 13 ([13], simplified for this paper)

- $\Pi_{\mathcal{T}}\cup\mathcal{A}_{db}\models\mathsf{C}_B(a)$ *implies* $\Sigma_{\mathcal{T}}\cup\mathcal{A}_{db}\models\mathsf{C}_B(a)$*, and*
- $\Pi_{\mathcal{T}}\cup\mathcal{A}_{db}\not\models\mathsf{C}_B(a)$ *implies* $\Sigma_{\mathcal{T}}\cup\mathcal{A}_{db}\models\neg\mathsf{C}_B(a)$*.*

[4] This rule allows one to *remove atoms over quantified variables* (y in our case) in a query by adding atoms over the remaining variables (x in our case) that imply the existence of the original atoms with the help of the TBox.

(and similarly for $\mathsf{R}_R(a,b)$ consequences) for every ABox $\mathcal{A}$ and constant a (and b), where $\mathcal{A}_{db}$ is the closed world variant of $\mathcal{A}$, a set of ground facts such that all facts not in *$\mathcal{A}_{db}$ are false.*

Note that Clark's result works in the much more general setting of logic programs with function symbols and possibly infinite resolution proofs and under the *Negation As Failure* semantics. Since Clark's completion makes all IDB predicates closed, we can now use standard tools for testing for explicit definability.

Also note that, had we used $\Pi_\mathcal{T}$ instead, none of the definability results could possibly hold, and that, in the absence of role/feature subsumptions (such as role hierarchies), there is no need to apply the completion to the R_R atoms.

Proposition 14 (Projective Beth Definability [4]) *Let Σ be an FO theory over symbols in L and $L' \subseteq L$. Then the following are equivalent:*

1. *For M_1 and M_2 models of Σ such that $M_{1|L'} = M_{2|L'}$, it holds that $M_1 \models \varphi[\boldsymbol{a}]$ iff $M_2 \models \varphi[\boldsymbol{a}]$ for all M_1, M_2, and $\boldsymbol{a}$ tuples of constants, and*
2. *φ is equivalent under Σ to a formula ψ in L' (we say φ* is Beth definable over Σ and L'*).*

This gives us a complete characterization of FO rewritability of the ABox closure of individual primitive concept names with respect to Horn-$\mathcal{ALCHQI}$ TBoxes as follows:

Theorem 15 *Let $\mathcal{T}$ be a Horn-$\mathcal{ALCHQI}$ TBox over the primitive concept names $\{A_1,\ldots,A_k\}$ and role names $\{R_1,\ldots,R_n\}$. Then the completion of the primitive concept* $\mathrm{A_i}$ *(role R_i) w.r.t. $\mathcal{T}$ is* FO definable *if and only if $\mathsf{C}_{A_i}(x)$ ($\mathsf{R}_{R_i}(x,y)$) is Beth definable over $\Sigma_\mathcal{T}$ and $L' = \{\mathsf{P}_{A_1},\ldots,\mathsf{P}_{A_k},\mathsf{P}_{R_1},\ldots,\mathsf{P}_{R_n}\}$.*

Proof (sketch): Follows immediately from the properties of Beth definability (Proposition 14) and the definition and properties of the Clark's completion (Proposition 13).

Observe that one can restrict the alphabet of the ABox (L') to target only ABoxes over restricted signature(s).

Given $\Sigma_\mathcal{T}$, one can now reformulate (1) in Proposition 14 as a logical implication problem by making a copy of all formulas of $\Sigma_\mathcal{T}$ in which all non-logical symbols *not in* $\{\mathsf{P}_{A_1},\ldots,\mathsf{P}_{A_k},\mathsf{P}_{R_1},\ldots,\mathsf{P}_{R_n}\}$ are starred. Hence, the definability question for $\mathsf{C}_A(x)$ and $\mathsf{R}_R(x,y)$ can be expressed as a logical implication question of the form:

$$\begin{array}{l} \Sigma_\mathcal{T} \cup \Sigma^*_\mathcal{T} \models \forall x.\mathsf{C}_A(x) \to \mathsf{C}^*_A(x) \\ \Sigma_\mathcal{T} \cup \Sigma^*_\mathcal{T} \models \forall x,y.\mathsf{R}_R(x,y) \to \mathsf{R}^*_R(x,y) \end{array} \qquad (1)$$

Note that, without role constructors, there is no need to check for the definability of $\mathsf{R}_R(x,y)$ atoms since they are always definable (we elaborate on the role of role constructors in Section 5). Note also that, on closer inspection, all formulas in $\Sigma_\mathcal{T}$ can be written as $\mathcal{ALCI}$ subsumptions. Hence:

Theorem 16 *Let $\mathcal{T}$ be a Horn-$\mathcal{ALCHQI}$ TBox. Then the existence of*

1. *the FO rewritability of the $\mathcal{A}$ completion with respect to $\mathcal{T}$, and*
2. *the uniform query rewritability over $\mathcal{T}$*

are decidable and in EXPTIME.

Proof (sketch): The first claim follows immediately from Theorem 15 applied to all atoms of the form C_B and the decidability and complexity of reasoning in $\mathcal{ALCI}$. The second claim follows by observing that (i) definability of atomic queries implies definability of arbitrary UCQs using the combined combined approach, and that (ii) non-definability of a single atomic query exhibits the need for a non FO ABox completion for queries containing/consisting of this atom.

In the second case, one can restrict the definability conditions to atoms that can appear in the query $\varphi_{\mathcal{T}}$. A matching lower bound can be obtained for expressive fragments of Horn-$\mathcal{ALC}$ (for which the reasoning complexity is EXPTIME-complete).[5] However, since the size (and the construction) of rewritings will dominate this cost (even for the simplest ontology languages [23]), exact complexity bounds are mostly of academic interest.

Construction of rewritings. To obtain an algorithm that constructs rewritings from our characterization of FO rewritability, we utilize Craig Interpolation:

Proposition 17 (Craig Interpolation [14]) *Let φ and ϕ be FO formulas such that $\models \varphi \rightarrow \phi$. Then there is an FO formula ψ, called* Craig interpolant*, containing only symbols common to φ and ϕ such that $\models \varphi \rightarrow \psi$ and $\models \psi \rightarrow \phi$.*

Moreover, the interpolant can be extracted, typically in linear time, from a proof of $\models \varphi \rightarrow \phi$, as long as a reasonably structural proof system, such as resolution, (cut-free) sequent calculus, and/or analytic tableau is used. Combining the above construction with the rewriting $\varphi_{\mathcal{T}}$ we get:

Theorem 18 *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a consistent Horn-$\mathcal{ALCHQI}$ knowledge base. Then the data complexity of uniform conjunctive query answering is in AC^0 whenever the $\mathcal{A}$ completion with respect to $\mathcal{T}$ is FO definable with respect to $\Sigma_{\mathcal{T}}$.*

Proof (sketch): Let $\psi_A(x)$ be an FO definition of C_A w.r.t. $\Sigma_{\mathcal{T}}$. Then $\mathcal{K} \models \varphi(\boldsymbol{a})$ iff $\mathcal{A}_{db} \models \varphi_{\mathcal{T}}[\psi_A[y/x]/A(y) \mid A \in \mathsf{PC}](\boldsymbol{a})$. The claim follows since $\varphi_{\mathcal{T}}[\psi_A[y/x]/A(y) \mid A \in \mathsf{PC}]$ is an FO formula, in particular a UCQ.

Our approach also provides decidability for the non-uniform (query-specific) problems. Also, while one can explicitly construct $\varphi_{\mathcal{T}}$, to decide FO rewritability of a CQ, one only needs to determine the atomic formulas for which interpolants are needed in the reformulation. This yields our desired result:

[5] It was noted in [36] that the exact complexity is open for PTIME fragments of Horn-$\mathcal{DLFD}$, such as $\mathcal{CFD}_{nc}$ and $\mathcal{CFDI}_{kc}^{\forall-}$.

Theorem 19 *Let $\mathcal{T}$ be a Horn-$\mathcal{ALCHQI}$ TBox and φ a CQ. Then the following are equivalent:*

1. *φ is FO rewritable with respect to $\mathcal{T}$ and*
2. *$\Sigma_{\mathcal{T}} \cup \Sigma^*_{\mathcal{T}} \models \forall x.\mathsf{C}_A(x) \rightarrow \mathsf{C}^*_A(x)$ for all C_A appearing in $\varphi_{\mathcal{T}}$ and $\Sigma_{\mathcal{T}} \cup \Sigma^*_{\mathcal{T}} \models \forall x, y.\mathsf{R}_R(x, y) \rightarrow \mathsf{R}^*_R(x, y)$ for all R_R appearing in $\varphi_{\mathcal{T}}$.*

The exact complexity again depends on the complexity of (2) above. In the general case, an EXPTIME bound follows from [20], but again, a more refined analysis is in order for fragments of Horn-$\mathcal{ALCHQI}$.

4 A One-step Construction of Rewritings

We have been considering the test for existence of rewritings and the construction of such rewritings as a two part process: (i) the construction of an ABox completion, and (ii) subsequent query reformulation. We have already noted that non-rewritability can always be traced to part (i) of this process. An interesting question that emerges, however, is whether such a two-part process is needed. In this section we outline a *one-step* approach to the problem that continues to be based on Clark's completion and on Beth definability, optionally followed by Craig interpolation. Now, however, we apply both techniques to the full TBox, i.e., including *existential restrictions* that may generate anonymous objects, and to the user query, (to save space, in Horn-$\mathcal{ALC}$ only).

Definition 20 (Logic Program for Horn-$\mathcal{ALC}$ TBox)

(entailed by $\mathcal{T}$)	*(completion rule in $\Pi_{\mathcal{T}}$)*
$\mathrm{A} \sqsubseteq \bot$	$\mathsf{C}_\bot(x) \leftarrow \mathsf{C}_A(x)$
$\mathrm{A}_1 \sqcap \mathrm{A}_2 \sqsubseteq \mathrm{B}$	$\mathsf{C}_B(x) \leftarrow \mathsf{C}_{A_1}(x), \mathsf{C}_{A_2}(x)$
$\mathrm{A} \sqsubseteq \forall R.\mathrm{B}$	$\mathsf{C}_B(x) \leftarrow \mathsf{C}_A(y), \mathsf{R}_R(y, x)$
$\exists R.\mathrm{A} \sqsubseteq \mathrm{B}$	$\mathsf{C}_B(x) \leftarrow \mathsf{C}_A(y), \mathsf{R}_R(x, y)$
$\mathrm{A} \sqsubseteq \exists R.\mathrm{B}$	$\mathsf{R}_R(x, f_R(x)) \leftarrow \mathsf{C}_A(x)$, *and* $\mathsf{C}_B(f_R(x)) \leftarrow \mathsf{C}_A(x)$

Note that the construction of a Datalog program in Section 3 omitted the last rule (for $\mathrm{A} \sqsubseteq \exists R.\mathrm{B}$) since the effect of that subsumption has been accommodated by query reformulation. The clauses stemming from the existential restrictions contain Skolem functions f_R and hence the resulting set of clauses is no longer a Datalog program. To define the Clark's completion, observe that the clauses for $\mathrm{A} \sqsubseteq \exists R.\mathrm{B}$ can be equivalently written as

$$\mathsf{R}_R(x, y) \leftarrow y = f_R(x), \mathsf{C}_A(x)$$
$$\mathsf{C}_B(x) \leftarrow x = f_R(y), \mathsf{C}_A(y)$$

as shown in [13]. Now, since the heads of the clauses in $\Pi_{\mathcal{T}}$ have a *uniform* format, we can create the completion $\Sigma_{\mathcal{T}}$ as in Definition 12. (This requires adding the standard equality axioms or assuming equality is an *interpreted* predicate.) For a given CQ φ of the form $\{\bar{x} \mid \psi\}$, extend the completion with

$$\Sigma_{\mathcal{T},\varphi} = \Sigma_{\mathcal{T}} \cup \{Q(\bar{x}) \leftrightarrow \psi\}$$

(with Q a new symbol). Then one can apply the definability test as in the previous case to obtain the rewritability test:

Theorem 21 *Let $\mathcal{T}$ be a Horn-$\mathcal{ALC}$ TBox and φ a CQ. Then the following are equivalent:*

1. $\Sigma_{\mathcal{T},\varphi} \cup \Sigma^*_{\mathcal{T},\varphi} \models \forall\bar{x}.Q(\bar{x}) \rightarrow Q^*(\bar{x})$*, and*
2. φ *is FO rewritable with respect to* $\mathcal{T}$*.*

The theorem provides a direct, sound and complete test for FO rewritability of CQs with respect to Horn-$\mathcal{ALC}$ TBoxes. However, unlike in the case of Datalog, we need to ensure that the test is still decidable and has a reasonable computational complexity. Note that the actual complexity in this case is tied to the proof system used to prove (1) in Theorem 21: as in DL reasoners, not every proof system achieves the optimal complexity bound. For Horn-$\mathcal{ALC}$, a reduction to the Ackermann prefix with equality [1] seems feasible, with the aim of obtaining the complexity bound using Fürer's result [18]. However, if one is interested in generating the rewriting in the form of an interpolant, a suitable proof system that supports interpolant generation, such as Analytic Tableau [17], (cut-free) Sequent Calculus, or Resolution, is needed. Alternative blocking-style techniques used in DL tableau reasoners are very likely to apply here as well. An intriguing possibility is to also just limit the depth of terms in a general high-performance theorem prover along the lines described by Chomicki for Datalog$_{nS}$ [12]. Both of these options are subjects of future research.

An interesting application of Theorem 21 emerges when the given TBox is formulated in DL-Lite variants, for which interpolants will then correspond to the result obtained by *perfect query reformulation* developed in [10,11]. It is relatively easy to observe that the *one-step* interpolation-based approach always succeeds and produces essentially the *perfect rewritings* of conjunctive queries.

5 Summary and Extensions

In this section, we briefly discuss several common extensions of Horn-$\mathcal{ALC}$ that we have omitted so far in our development to keep the presentation of the main ideas cleaner. In the light of Theorem 21, it is relatively immediate that any extension that leads to Horn $\Pi_{\mathcal{T}}$ can be accommodated. Note that, to extend the two-step combined combined approach, we would need to modify, often in non-trivial manner, the query reformulation algorithm. (For an example that accommodates inverse features and a variety of equality generating dependencies called *path-functional dependencies*, see [31].)

Additional concept and role constructors, and the induced subsumptions, can be classified in three groups:

1. Constructors that lead to *full* Horn rules, i.e., without existential quantifiers in their heads, that preserve the tree model property. Rules corresponding to these constructors can simply be added in Definition 10 and Definition 20 without any major impact on the query reformulation in Definition 11;
2. Constructors that lead to *embedded* Horn rules with existential quantifiers in their heads that continue to preserve the tree model property. Here, both Definition 10 and 11 need to be extended to account for the possibility of additional anonymous individuals. Alternatively, one can capture all of the effects by naturally extending Definition 20 and proceeding with a one-step definability test; and
3. Constructors that break the tree-model property. Examples relate to transitivity assertions and nominals; here, it is not always clear how to modify Definition 11 to make Proposition 9 hold. However, extending Definition 20 and subsequently using Theorem 21 will still work.

However, using Theorem 21, while sound and complete for determining rewritability, does not come for free. With each extension, one needs to revisit the decidability and complexity of the definability test which, ultimately, becomes undecidable. This happens even in cases when only unary function symbols are needed but where unrestricted use of binary predicates, such as roles, are allowed.

Extensions that are unlikely to be possible. There are limits to the definability-based approach:

(*beyond Horn logics*) The approach for Horn logics relies crucially on the existence of a unique minimal model (called the universal model in DL circles) that can be characterized using the Clark's completion. This insight then makes Beth definability and Craig interpolation work. It remains unclear how this idea could generalize to logics without the minimal model property (i.e., non-Horn). For these reasons PTIME-coNP boundaries [30] are unlikely to be resolved using these techniques.

(*beyond FO logics*) The synthesis of the rewritings is tied to Craig Interpolation. Hence synthesizing, e.g., linear Datalog or dealing with dichotomies on the NL-PTIME [26] boundary seems also to be beyond the capabilities of the techniques used in this paper. Applying results on interpolation in non-first order logics, such as the μ-calculus [15], will be the focus of future research. However, the combined combined approach already gives one a Datalog rewriting, so the space to be explored seems to be rather limited.

References

1. W. Ackermann. Uber die Erfullbarkeit gewisser Zahlausdrucke. *Mathematische Annalen*, 100:638–649, 1928.
2. Franz Baader, Diego Calvanese, Deborah L. McGuinness, Daniele Nardi, and Peter F. Patel-Schneider. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press, 2003.

3. Timea Bagosi, Diego Calvanese, Josef Hardi, Sarah Komla-Ebri, Davide Lanti, Martin Rezk, Mariano Rodriguez-Muro, Mindaugas Slusnys, and Guohui Xiao. The ontop framework for ontology based data access. In *The Semantic Web and Web Science - 8th Chinese Conference, CSWS 2014, Revised Selected Papers*, pages 67–77, 2014.
4. Evert Willem Beth. On Padoa's method in the theory of definition. *Indagationes Mathematicae*, 15:330–339, 1953.
5. Meghyn Bienvenu, Peter Hansen, Carsten Lutz, and Frank Wolter. First order-rewritability and containment of conjunctive queries in horn description logics. In *Proceedings of the 29th International Workshop on Description Logics, Cape Town, South Africa*, 2016.
6. Meghyn Bienvenu, Stanislav Kikot, Roman Kontchakov, Vladimir V. Podolskii, Vladislav Ryzhikov, and Michael Zakharyaschev. The complexity of ontology-based data access with OWL 2 QL and bounded treewidth queries. In *Proceedings of the 36th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, PODS 2017*, pages 201–216, 2017.
7. 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.
8. Alexander Borgida, Jos de Bruijn, Enrico Franconi, Inanç Seylan, Umberto Straccia, David Toman, and Grant E. Weddell. On finding query rewritings under expressive constraints. In *SEBD*, pages 426–437, 2010.
9. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, Antonella Poggi, Mariano Rodriguez-Muro, Riccardo Rosati, Marco Ruzzi, and Domenico Fabio Savo. The MASTRO system for ontology-based data access. *Semantic Web*, 2(1):43–53, 2011.
10. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. *DL-Lite*: Tractable description logics for ontologies. In *Proc. of the 20th Nat. Conf. on Artificial Intelligence (AAAI 2005)*, pages 602–607, 2005.
11. Diego Calvanese, Giuseppe de Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Tractable Reasoning and Efficient Query Answering in Description Logics: The DL-Lite Family. *J. of Automated Reasoning*, 39(3):385–429, 2007.
12. Jan Chomicki. Depth-bounded bottom-up evaluation of logic program. *J. Log. Program.*, 25(1):1–31, 1995.
13. Keith L. Clark. Negation as failure. In *Logic and Data Bases, Symposium on Logic and Data Bases, Centre d'études et de recherches de Toulouse, France, 1977*, pages 293–322, 1977.
14. William Craig. Three uses of the Herbrand-Genzen theorem in relating model theory and proof theory. *Journal of Symbolic Logic*, 22:269–285, 1957.
15. Giovanna D'Agostino and Marco Hollenberg. Logical questions concerning the mu-calculus: Interpolation, lyndon and los-tarski. *J. Symb. Log.*, 65(1):310–332, 2000.
16. Thomas Eiter, Magdalena Ortiz, Mantas Simkus, Trung-Kien Tran, and Guohui Xiao. Query rewriting for horn-shiq plus rules. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada.*, 2012.
17. Melvin Fitting. *First-Order Logic and Automated Theorem Proving, Second Edition.* Graduate Texts in Computer Science. Springer Publishers, 1996.
18. Martin Fürer. Alternation and the Ackermann Case of the Decision Problem. *L'Enseignement Math.*, 27:137–162, 1981.

19. Peter Hansen, Carsten Lutz, Inanç Seylan, and Frank Wolter. Efficient query rewriting in the description logic EL and beyond. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015*, pages 3034–3040, 2015.
20. Ian Horrocks, Ulrike Sattler, Sergio Tessaris, and Stephan Tobies. How to decide query containment under constraints using a description logic. In *Logic for Programming and Automated Reasoning, 7th International Conference, LPAR 2000, Reunion Island, France, November 11-12, 2000, Proceedings*, pages 326–343, 2000.
21. Alexander K. Hudek, David Toman, and Grant E. Weddell. On enumerating query plans using analytic tableau. In *Automated Reasoning with Analytic Tableaux and Related Methods - 24th International Conference, TABLEAUX 2015, Wrocław, Poland, September 21-24, 2015. Proceedings*, pages 339–354, 2015.
22. Ullrich Hustadt, Boris Motik, and Ulrike Sattler. Data complexity of reasoning in very expressive description logics. In *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 466–471. Professional Book Center, 2005.
23. Stanislav Kikot, Roman Kontchakov, Vladimir V. Podolskii, and Michael Zakharyaschev. Long rewritings, short rewritings. In *Proceedings of the 2012 International Workshop on Description Logics, DL-2012, Rome, Italy, June 7-10, 2012*, 2012.
24. Roman Kontchakov, Carsten Lutz, David Toman, Frank Wolter, and Michael Zakharyaschev. The combined approach to query answering in DL-Lite. In *Proc. KR*, pages 247–257, 2010.
25. Roman Kontchakov, Carsten Lutz, David Toman, Frank Wolter, and Michael Zakharyaschev. The combined approach to ontology-based data access. In *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 2656–2661, 2011.
26. Carsten Lutz and Leif Sabellek. Ontology-mediated querying with the description logic EL: trichotomy and linear datalog rewritability. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017*, pages 1181–1187, 2017.
27. Carsten Lutz, Inanç Seylan, David Toman, and Frank Wolter. The combined approach to OBDA: Taming role hierarchies using filters. In *ISWC (1)*, pages 314–330, 2013.
28. Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic $\mathcal{EL}$ using a relational database system. In *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, pages 2070–2075, 2009.
29. Carsten Lutz, David Toman, and Frank Wolter. Conjunctive query answering in the description logic EL using a relational database system. In *Proc. IJCAI*, pages 2070–2075, 2009.
30. Carsten Lutz and Frank Wolter. Non-uniform data complexity of query answering in description logics. In *Principles of Knowledge Representation and Reasoning: Proceedings of the Thirteenth International Conference, KR 2012*, 2012.
31. Stephanie McIntyre, Alexander Borgida, David Toman, and Grant E. Weddell. On limited conjunctions and partial features in parameter-tractable feature logics. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019*, pages 2995–3002, 2019.
32. Jason St. Jacques, David Toman, and Grant E. Weddell. Object-relational queries over $\mathcal{CFDI}_{nc}$ knowledge bases: OBDA for the SQL-Literate. In *Proc. IJCAI*, pages 1258–1264, 2016.
33. David Toman and Grant E. Weddell. *Fundamentals of Physical Design and Query Compilation.* Synthesis Lectures on Data Management. Morgan & Claypool Publishers, 2011.

34. David Toman and Grant E. Weddell. Conjunctive Query Answering in $\mathcal{CFD}_{nc}$: A PTIME Description Logic with Functional Constraints and Disjointness. In *Australasian Conference on Artificial Intelligence*, pages 350–361, 2013.
35. David Toman and Grant E. Weddell. An interpolation-based compiler and optimizer for relational queries (system design report). In *IWIL@LPAR 2017 Workshop and LPAR-21 Short Presentations, Maun, Botswana, May 7-12, 2017*, 2017.
36. David Toman and Grant E. Weddell. First order rewritability for ontology mediated querying in horn-dlfd. In *Proceedings of the 33rd International Workshop on Description Logics (DL 2020)*, volume 2663 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2020.
37. J.D. Ullman. *Principles of Database Systems*. Computer Science Press, 1982.
38. J.D. Ullman. *Principles of Database and Knowledge-Base Systems*, volume 1. Computer Science Press, 1988.
39. J.D. Ullman. *Principles of Database and Knowledge-Base Systems*, volume 2. Computer Science Press, 1989.