A set theoretical approach for ABox reasoning services

Domenico Cantone, Marianna Nicolosi-Asmundo, and
Daniele Francesco Santamaria

University of Catania, Dept. of Mathematics and Computer Science
email: {cantone,nicolosi,santamaria}@dmi.unict.it

Abstract. In this paper we consider the most common ABox reasoning services for the description logic $\mathcal{DL}\langle \mathsf{4LQS}^{\mathsf{R},\times}\rangle(\mathbf{D})$ ($\mathcal{DL}_{\mathbf{D}}^{4,\times}$, for short) and prove their decidability via a reduction to the satisfiability problem for the set-theoretic fragment $\mathsf{4LQS}^{\mathsf{R}}$. $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ is a very expressive description logic admitting various concept and role constructs, datatypes, and it allows one to represent rule based languages such as SWRL.
Decidability results are achieved by defining a generalized version of the conjunctive query answering problem, called HOCQA (Higher Order Conjunctive Query Answering), that can be instantiated to the most widespread ABox reasoning tasks. Then, a KE-tableau based procedure is defined to calculate the answer set from a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ knowledge base and from a higher order $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ conjunctive query. The system is an extension of a KE-tableau based decision procedure for the CQA problem introduced in a previous work and allows one to reason on several well known ABox reasoning tasks.

1 Introduction

Recently, some results from Computable Set Theory have been applied in the ambit of knowledge representation for the semantic web to define and reason about description logics and rule languages.

In particular, the decidable four-level stratified fragment of set theory $\mathsf{4LQS}^{\mathsf{R}}$ involving variables of four sorts, pair terms, and a restricted form of quantification over variables of the first three sorts (cf. [7]) is used in [4] to represent the description logic $\mathcal{DL}\langle \mathsf{4LQS}^{\mathsf{R}}\rangle(\mathbf{D})$ (more simply referred to as $\mathcal{DL}_{\mathbf{D}}^{4}$). The logic $\mathcal{DL}_{\mathbf{D}}^{4}$ admits concept constructs such as full negation, union and intersection of concepts, concept domain and range, existential quantification and min cardinality on the left-hand side of inclusion axioms. It also supports role constructs such as role chains on the left hand side of inclusion axioms, union, intersection, and complement of abstract roles, and properties on roles such as transitivity, symmetry, reflexivity, and irreflexivity. As briefly shown in [4] $\mathcal{DL}_{\mathbf{D}}^{4}$ is particularly suitable to express a rule language such as the Semantic Web Rule Language (SWRL) [25], an extension of the Ontology Web Language (OWL). It admits datatypes, a simple form of concrete domains that are relevant in real world applications. In [4] the consistency problem for $\mathcal{DL}_{\mathbf{D}}^{4}$-knowledge bases has been proved decidable by means of a reduction to the satisfiability problem for $\mathsf{4LQS^R}$, proved decidable in [7]. It has also been shown, under not very restrictive constraints, that the consistency problem for $\mathcal{DL}^4_{\mathbf{D}}$-knowledge bases is **NP**-complete. The latter result has practical interest since such a restricted version of $\mathcal{DL}^4_{\mathbf{D}}$ allows on to express several ontologies, such as Ontoceramic [11].

In [9] the description logic $\mathcal{DL}\langle\mathsf{4LQS^{R,\times}}\rangle(\mathbf{D})$ ($\mathcal{DL}^{4,\times}_{\mathbf{D}}$, for short), extending $\mathcal{DL}^4_{\mathbf{D}}$ with Boolean operations on concrete roles and with the product of concepts, is introduced and the *Conjunctive Query Answering* (CQA) problem for $\mathcal{DL}^{4,\times}_{\mathbf{D}}$ is defined and proved decidable via a reduction to the CQA problem for $\mathsf{4LQS^R}$, whose decidability follows from that of the satisfiability problem for $\mathsf{4LQS^R}$ (proved in [7]). CQA is a a powerful way to query ABoxes relevant in the context of description logics and, in particular, for real world applications based on semantic web technologies, since it provides a mechanism allowing users and applications to interact with ontologies and data. The task of CQA has been studied for several well-known description logics (cf. [1–3,13–18,20–23]). Finally, an always terminating KE-tableau based procedure is designed that, given a $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-query Q and a $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-knowledge base $\mathcal{KB}$ represented in set-theoretic terms, determines the answer set of Q with respect to $\mathcal{KB}$, providing also some complexity results. The KE-tableau system [12] has been chosen because this variant of the tableau method permits the construction of trees whose distinct branches define mutually exclusive situations thus preventing the proliferation of redundant branches, typical of semantic tableaux.

In this paper we extend the results presented in [9] considering the main ABox reasoning tasks for $\mathcal{DL}^{4,\times}_{\mathbf{D}}$ such as instance checking and concept retrieval and study their decidability via a reduction to the satisfiability problem for the set-theoretic fragment $\mathsf{4LQS^R}$. Specifically, we define Higher-Order $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-conjunctive queries (HO $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-conjunctive queries) admitting variables of three sorts: individual and datatype values variables, concept variables, and role variables. HO $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-conjunctive queries can be instantiated to any of the ABox reasoning tasks we are considering in the paper. Then, we define the Higher Order Conjunctive Query Answering (HOCQA) problem for $\mathcal{DL}^{4,\times}_{\mathbf{D}}$ and prove its decidability by reducing it to the HOCQA problem for $\mathsf{4LQS^R}$. Decidability of the latter problem follows from that of the satisfiability problem for $\mathsf{4LQS^R}$. $\mathsf{4LQS^R}$ representation of $\mathcal{DL}^{4,\times}_{\mathbf{D}}$ knowledge bases is defined according to [9]. HO $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-conjunctive queries are easily translated into $\mathsf{4LQS^R}$-formulae. In particular individual and datatype value variables are mapped into $\mathsf{4LQS^R}$ variables of sort 0, concept variables into $\mathsf{4LQS^R}$ variables of sort 1, and role variables into $\mathsf{4LQS^R}$ variables of sort 3. Finally we define a KE-tableau based decision procedure for the HOCQA task for $\mathcal{DL}^{4,\times}_{\mathbf{D}}$ that extends the system presented in [9] since it is provided with a specific procedure to deal with literals of the forms $x = y$ and $\neg(x = y)$ and it is able to deal with HO $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-conjunctive queries.

2 Preliminaries

2.1 The set-theoretic fragment $\mathsf{4LQS}^{\mathsf{R}}$

It is convenient to first introduce the syntax and semantics of a more general four-level quantified language, denoted 4LQS. Then we provide some restrictions on quantified formulae of 4LQS that characterize $\mathsf{4LQS}^{\mathsf{R}}$. The interested reader can find all the details in [7] together with the decision procedure for the satisfiability problem for $\mathsf{4LQS}^{\mathsf{R}}$.

4LQS involves four collections, $\mathcal{V}_i$, of variables of sort i, for $i = 0, 1, 2, 3$. Variables of sort i, for $i = 0, 1, 2, 3$, will be denoted by $X^i, Y^i, Z^i, \ldots$ (in particular, variables of sort 0 will also be denoted by $x, y, z, \ldots$). In addition to variables, 4LQS involves also *pair terms* of the form $\langle x, y\rangle$, with $x, y \in \mathcal{V}_0$.

4LQS-*quantifier-free atomic formulae* are classified as:

- level 0: $x = y$, $x \in X^1$, $\langle x, y\rangle = X^2$, $\langle x, y\rangle \in X^3$;
- level 1: $X^1 = Y^1$, $X^1 \in X^2$;
- level 2: $X^2 = Y^2$, $X^2 \in X^3$.

4LQS-*purely universal formulae* are classified as:

- level 1: $(\forall z_1) \ldots (\forall z_n)\varphi_0$, where $z_1, \ldots, z_n \in \mathcal{V}_0$ and φ_0 is any propositional combination of quantifier-free atomic formulae of level 0;
- level 2: $(\forall Z^1_1) \ldots (\forall Z^1_m)\varphi_1$, where $Z^1_1, \ldots, Z^1_m \in \mathcal{V}_1$ and φ_1 is any propositional combination of quantifier-free atomic formulae of levels 0 and 1, and of purely universal formulae of level 1;
- level 3: $(\forall Z^2_1) \ldots (\forall Z^2_p)\varphi_2$, where $Z^2_1, \ldots, Z^2_p \in \mathcal{V}_2$ and φ_2 is any propositional combination of quantifier-free atomic formulae and of purely universal formulae of levels 1 and 2.

4LQS-formulae are all the propositional combinations of quantifier-free atomic formulae of levels 0, 1, 2, and of purely universal formulae of levels 1, 2, 3.

The variables $z_1, \ldots, z_n$ are said to occur *quantified* in $(\forall z_1) \ldots (\forall z_n)\varphi_0$. Likewise, $Z^1_1, \ldots, Z^1_m$ and $Z^2_1, \ldots, Z^2_p$ occur quantified in $(\forall Z^1_1) \ldots (\forall Z^1_m)\varphi_1$ and in $(\forall Z^2_1) \ldots (\forall Z^2_p)\varphi_2$, respectively. A variable occurs *free* in a 4LQS-formula φ if it does not occur quantified in any subformula of φ. For $i = 0, 1, 2, 3$, we denote with $\mathtt{Var}_i(\varphi)$ the collections of variables of level i occurring free in φ.

A substitution $\sigma \coloneqq \{\boldsymbol{x}/\boldsymbol{y}, \boldsymbol{X}^1/\boldsymbol{Y}^1, \boldsymbol{X}^2/\boldsymbol{Y}^2, \boldsymbol{X}^3/\boldsymbol{Y}^3\}$ is the mapping $\varphi \mapsto \varphi\sigma$ such that, for any given 4LQS-formula φ, $\varphi\sigma$ is the 4LQS-formula obtained from φ by replacing the free occurrences of the variables x_i (for $i = 1, \ldots, n$) with y_i, of X^1_j (for $j = 1, \ldots, m$) with Y^1_j, of X^2_k (for $k = 1, \ldots, p$) with Y^2_k, and of X^3_h (for $h = 1, \ldots, q$) with Y^3_h, respectively. A substitution σ is *free* for φ if the formulae φ and $\varphi\sigma$ have exactly the same occurrences of quantified variables.

A 4LQS-*interpretation* is a pair $\boldsymbol{\mathcal{M}} = (D, M)$, where D is a non-empty collection of objects (called *domain* or *universe* of $\boldsymbol{\mathcal{M}}$) and M is an assignment over the variables in $\mathcal{V}_i$, for $i = 0, 1, 2, 3$, such that:

$$MX^0 \in D, \quad MX^1 \in \mathcal{P}(D), \quad MX^2 \in \mathcal{P}(\mathcal{P}(D)), \quad MX^3 \in \mathcal{P}(\mathcal{P}(\mathcal{P}(D))),$$

where $X^i \in \mathcal{V}_i$, for $i = 0, 1, 2, 3$, and $\mathcal{P}(s)$ denotes the powerset of s.
Pair terms are interpreted *à la* Kuratowski, and therefore we put

$$M\langle x, y\rangle := \{\{Mx\}, \{Mx, My\}\}.$$

Quantifier-free atomic formulae and purely universal formulae are evaluated in a standard way according to the usual meaning of the predicates '$\in$' and '$=$'. The interpretation of quantifier-free atomic formulae and of purely universal formulae is given in [7].

Finally, compound formulae are interpreted according to the standard rules of propositional logic. If $\boldsymbol{\mathcal{M}} \models \varphi$, then $\boldsymbol{\mathcal{M}}$ is said to be a 4LQS-model for φ. A 4LQS-formula is said to be *satisfiable* if it has a 4LQS-model. A 4LQS-formula is *valid* if it is satisfied by all 4LQS-interpretations.

We are now ready to present the fragment $\mathsf{4LQS^R}$ of 4LQS of our interest. This is the collection of the formulae ψ of 4LQS fulfilling the restrictions:

1. for every purely universal formula $(\forall Z_1^1)\ldots(\forall Z_m^1)\varphi_1$ of level 2 occurring in ψ and every purely universal formula $(\forall z_1)\ldots(\forall z_n)\varphi_0$ of level 1 occurring negatively in φ_1, φ_0 is a propositional combination of quantifier-free atomic formulae of level 0 and the condition
$$\neg\varphi_0 \rightarrow \bigwedge_{i=1}^{n}\bigwedge_{j=1}^{m} z_i \in Z_j^1$$
is a valid 4LQS-formula (in this case we say that $(\forall z_1)\ldots(\forall z_n)\varphi_0$ is *linked to the variables* $Z_1^1, \ldots, Z_m^1$);
2. for every purely universal formula $(\forall Z_1^2)\ldots(\forall Z_p^2)\varphi_2$ of level 3 in ψ:
   - every purely universal formula of level 1 occurring negatively in φ_2 and not occurring in a purely universal formula of level 2 is only allowed to be of the form
   $$(\forall z_1)\ldots(\forall z_n)\neg(\bigwedge_{i=1}^{n}\bigwedge_{j=1}^{n}\langle z_i, z_j\rangle = Y_{ij}^2),$$
   with $Y_{ij}^2 \in \mathcal{V}^2$, for $i, j = 1, \ldots, n$;
   - purely universal formulae $(\forall Z_1^1)\ldots(\forall Z_m^1)\varphi_1$ of level 2 may occur only positively in φ_2.[1]

Restriction 1 has been introduced for technical reasons concerning the decidability of the satisfiability problem for the fragment, while restriction 2 allows one to define binary relations and several operations on them.

The semantics of $\mathsf{4LQS^R}$ plainly coincides with that of 4LQS.

2.2 The logic $\mathcal{DL}\langle\mathsf{4LQS^{R,\times}}\rangle(\mathbf{D})$

The description logic $\mathcal{DL}\langle\mathsf{4LQS^{R,\times}}\rangle(\mathbf{D})$ (more simply referred to as $\mathcal{DL}_{\mathbf{D}}^{4,\times}$) is an extension of the logic $\mathcal{DL}\langle\mathsf{4LQS^R}\rangle(\mathbf{D})$, presented in [4] where Boolean operations on concrete roles and the product of concepts are defined. $\mathcal{DL}\langle\mathsf{4LQS^{R,\times}}\rangle(\mathbf{D})$ is

[1] Definitions of positive occurrence and of negative occurrence of a formula inside another formula can be found in [7].

more liberal than $\mathcal{SROIQ}(\mathbf{D})$, the logic underlying the most expressive Ontology Web Language 2 profile, OWL 2 DL [26], for what concerns the construction of role inclusion axioms since the roles involved are not required to be subject to any ordering relationship, and the notion of simple role is not needed. It also admits datatypes, a simple form of concrete domains that are relevant in real-world applications. In particular, it treats derived datatypes by admitting datatype terms constructed from data ranges by means of a finite number of applications of the Boolean operators. Basic and derived datatypes can be used inside inclusion axioms involving concrete roles.

Datatypes are defined according to [19] as follows. Let $\mathbf{D} = (N_D, N_C, N_F, \cdot^{\mathbf{D}})$ be a *datatype map*, where N_D is a finite set of datatypes, N_C is a function assigning a set of constants $N_C(d)$ to each datatype $d \in N_D$, N_F is a function assigning a set of facets $N_F(d)$ to each $d \in N_D$, and $\cdot^{\mathbf{D}}$ is a function assigning a datatype interpretation $d^{\mathbf{D}}$ to each datatype $d \in N_D$, a facet interpretation $f^{\mathbf{D}} \subseteq d^{\mathbf{D}}$ to each facet $f \in N_F(d)$, and a data value $e_d^{\mathbf{D}} \in d^{\mathbf{D}}$ to every constant $e_d \in N_C(d)$. We shall assume that the interpretations of the datatypes in N_D are nonempty pairwise disjoint sets.

Let $\mathbf{R_A}$, $\mathbf{R_D}$, $\mathbf{C}$, $\mathbf{I}$ be denumerable pairwise disjoint sets of abstract role names, concrete role names, concept names, and individual names, respectively. We assume that the set of abstract role names $\mathbf{R_A}$ contains a name U denoting the universal role.

(a) $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-datatype, (b) $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concept, (c) $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract role, and (d) $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concrete role terms are constructed according to the following syntax rules:

(a) $t_1, t_2 \longrightarrow dr \mid \neg t_1 \mid t_1 \sqcap t_2 \mid t_1 \sqcup t_2 \mid \{e_d\}$,

(b) $C_1, C_2 \longrightarrow A \mid \top \mid \bot \mid \neg C_1 \mid C_1 \sqcup C_2 \mid C_1 \sqcap C_2 \mid \{a\} \mid \exists R.Self \mid \exists R.\{a\} \mid \exists P.\{e_d\}$,

(c) $R_1, R_2 \longrightarrow S \mid U \mid R_1^- \mid \neg R_1 \mid R_1 \sqcup R_2 \mid R_1 \sqcap R_2 \mid R_{C_1|} \mid R_{|C_1} \mid R_{C_1 \mid C_2} \mid id(C) \mid C_1 \times C_2$,

(d) $P_1, P_2 \longrightarrow T \mid \neg P_1 \mid P_1 \sqcup P_2 \mid P_1 \sqcap P_2 \mid P_{C_1|} \mid P_{|t_1} \mid P_{C_1|t_1}$,

where dr is a data range for $\mathbf{D}$, t_1, t_2 are data-type terms, e_d is a constant in $N_C(d)$, a is an individual name, A is a concept name, C_1, C_2 are $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concept terms, S is an abstract role name, R, R_1, R_2 are $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract role terms, T is a concrete role name, and P, P_1, P_2 are $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concrete role terms. We remark that data-type terms are introduced in order to represent derived data-types.

A $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge base is a triple $\mathcal{K} = (\mathcal{R}, \mathcal{T}, \mathcal{A})$ such that $\mathcal{R}$ is a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*RBox*, $\mathcal{T}$ is a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*TBox*, and $\mathcal{A}$ a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*ABox*.

A $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*RBox* is a collection of statements of the following forms: $R_1 \equiv R_2$, $R_1 \sqsubseteq R_2$, $R_1 \ldots R_n \sqsubseteq R_{n+1}$, $\mathsf{Sym}(R_1)$, $\mathsf{Asym}(R_1)$, $\mathsf{Ref}(R_1)$, $\mathsf{Irref}(R_1)$, $\mathsf{Dis}(R_1, R_2)$, $\mathsf{Tra}(R_1)$, $\mathsf{Fun}(R_1)$, $R_1 \equiv C_1 \times C_2$, $P_1 \equiv P_2$, $P_1 \sqsubseteq P_2$, $\mathsf{Dis}(P_1, P_2)$, $\mathsf{Fun}(P_1)$, where R_1, R_2 are $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract role terms, C_1, C_2 are $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract concept terms, and P_1, P_2 are $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concrete role terms. Any expression of the type $w \sqsubseteq R$, where w is a finite string of $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract role terms and R is an $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract role term is called a *role inclusion axiom (RIA)*.

A $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*TBox* is a set of statements of the types:

- $C_1 \equiv C_2$, $C_1 \sqsubseteq C_2$, $C_1 \sqsubseteq \forall R_1.C_2$, $\exists R_1.C_1 \sqsubseteq C_2$, $\geq_n R_1.C_1 \sqsubseteq C_2$, $C_1 \sqsubseteq \leq_n R_1.C_2$,
- $t_1 \equiv t_2$, $t_1 \sqsubseteq t_2$, $C_1 \sqsubseteq \forall P_1.t_1$, $\exists P_1.t_1 \sqsubseteq C_1$, $\geq_n P_1.t_1 \sqsubseteq C_1$, $C_1 \sqsubseteq \leq_n P_1.t_1$,

where C_1, C_2 are $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concept terms, t_1, t_2 datatype terms, R_1 a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract role term, P_1 a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concrete role term. Any statement of the form $C \sqsubseteq D$, with C, D $\mathcal{DL}_{\mathbf{D}}^{4}$-concept terms, is a *general concept inclusion axiom (GCI)*.

A $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*ABox* is a set of *individual assertions* of the forms: $a : C_1$, $(a, b) : R_1$, $a = b$, $a \neq b$, $e_d : t_1$, $(a, e_d) : P_1$, with C_1 a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concept term, d a datatype, t_1 a datatype term, R_1 a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract role term, P_1 a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concrete role term, a, b individual names, and e_d a constant in $N_C(d)$.

The semantics of $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ is given by means of an interpretation $\mathbf{I} = (\Delta^{\mathbf{I}}, \Delta_{\mathbf{D}}, \cdot^{\mathbf{I}})$, where $\Delta^{\mathbf{I}}$ and $\Delta_{\mathbf{D}}$ are non-empty disjoint domains such that $d^{\mathbf{D}} \subseteq \Delta_{\mathbf{D}}$, for every $d \in N_D$, and $\cdot^{\mathbf{I}}$ is an interpretation function. The definition of the interpretation of concepts and roles, axioms and assertions is illustrated in Table 1.

Name	Syntax	Semantics
concept	A	$A^{\mathbf{I}} \subseteq \Delta^{\mathbf{I}}$
ab. (resp., cn.) rl.	R (resp., P)	$R^{\mathbf{I}} \subseteq \Delta^{\mathbf{I}} \times \Delta^{\mathbf{I}}$ (resp., $P^{\mathbf{I}} \subseteq \Delta^{\mathbf{I}} \times \Delta_{\mathbf{D}}$)
individual	a	$a^{\mathbf{I}} \in \Delta^{\mathbf{I}}$
nominal	$\{a\}$	$\{a\}^{\mathbf{I}} = \{a^{\mathbf{I}}\}$
dtype (resp., ng.)	d (resp., $\neg d$)	$d^{\mathbf{D}} \subseteq \Delta_{\mathbf{D}}$ (resp., $\Delta_{\mathbf{D}} \setminus d^{\mathbf{D}}$)
negative datatype term	$\neg t_1$	$(\neg t_1)^{\mathbf{D}} = \Delta_{\mathbf{D}} \setminus t_1^{\mathbf{D}}$
datatype terms intersection	$t_1 \sqcap t_2$	$(t_1 \sqcap t_2)^{\mathbf{D}} = t_1^{\mathbf{D}} \cap t_2^{\mathbf{D}}$
datatype terms union	$t_1 \sqcup t_2$	$(t_1 \sqcup t_2)^{\mathbf{D}} = t_1^{\mathbf{D}} \cup t_2^{\mathbf{D}}$
constant in $N_C(d)$	e_d	$e_d^{\mathbf{D}} \in d^{\mathbf{D}}$
data range	$\{e_{d_1}, \ldots, e_{d_n}\}$	$\{e_{d_1}, \ldots, e_{d_n}\}^{\mathbf{D}} = \{e_{d_1}^{\mathbf{D}}\} \cup \ldots \cup \{e_{d_n}^{\mathbf{D}}\}$
data range	ψ_d	$\psi_d^{\mathbf{D}}$
data range	$\neg dr$	$\Delta_{\mathbf{D}} \setminus dr^{\mathbf{D}}$
top (resp., bot.)	$\top$ (resp., $\bot$)	$\Delta^{\mathbf{I}}$ (resp., $\emptyset$)
negation	$\neg C$	$(\neg C)^{\mathbf{I}} = \Delta^{\mathbf{I}} \setminus C$
conj. (resp., disj.)	$C \sqcap D$ (resp., $C \sqcup D$)	$(C \sqcap D)^{\mathbf{I}} = C^{\mathbf{I}} \cap D^{\mathbf{I}}$ (resp., $(C \sqcup D)^{\mathbf{I}} = C^{\mathbf{I}} \cup D^{\mathbf{I}}$)
valued exist. quantification	$\exists R.a$	$(\exists R.a)^{\mathbf{I}} = \{x \in \Delta^{\mathbf{I}} : \langle x, a^{\mathbf{I}} \rangle \in R^{\mathbf{I}}\}$
datatyped exist. quantif.	$\exists P.e_d$	$(\exists P.e_d)^{\mathbf{I}} = \{x \in \Delta^{\mathbf{I}} : \langle x, e_d^{\mathbf{D}} \rangle \in P^{\mathbf{I}}\}$
self concept	$\exists R.\mathit{Self}$	$(\exists R.\mathit{Self})^{\mathbf{I}} = \{x \in \Delta^{\mathbf{I}} : \langle x, x \rangle \in R^{\mathbf{I}}\}$
nominals	$\{a_1, \ldots, a_n\}$	$\{a_1, \ldots, a_n\}^{\mathbf{I}} = \{a_1^{\mathbf{I}}\} \cup \ldots \cup \{a_n^{\mathbf{I}}\}$
universal role	U	$(U)^{\mathbf{I}} = \Delta^{\mathbf{I}} \times \Delta^{\mathbf{I}}$
inverse role	R^-	$(R^-)^{\mathbf{I}} = \{\langle y, x \rangle \mid \langle x, y \rangle \in R^{\mathbf{I}}\}$
concept cart. prod.	$C_1 \times C_2$	$(C_1 \times C_2)^I = C_1^I \times C_2^I$
abstract role complement	$\neg R$	$(\neg R)^{\mathbf{I}} = (\Delta^{\mathbf{I}} \times \Delta^{\mathbf{I}}) \setminus R^{\mathbf{I}}$

abstract role union	$R_1 \sqcup R_2$	$(R_1 \sqcup R_2)^{\mathbf{I}} = R_1^{\mathbf{I}} \cup R_2^{\mathbf{I}}$
abstract role intersection	$R_1 \sqcap R_2$	$(R_1 \sqcap R_2)^{\mathbf{I}} = R_1^{\mathbf{I}} \cap R_2^{\mathbf{I}}$
abstract role domain restr.	$R_{C\vert}$	$(R_{C\vert})^{\mathbf{I}} = \{\langle x, y\rangle \in R^{\mathbf{I}} : x \in C^{\mathbf{I}}\}$
concrete role complement	$\neg P$	$(\neg P)^{\mathbf{I}} = (\Delta^{\mathbf{I}} \times \Delta^{\mathbf{D}}) \setminus P^{\mathbf{I}}$
concrete role union	$P_1 \sqcup P_2$	$(P_1 \sqcup P_2)^{\mathbf{I}} = P_1^{\mathbf{I}} \cup P_2^{\mathbf{I}}$
concrete role intersection	$P_1 \sqcap P_2$	$(P_1 \sqcap P_2)^{\mathbf{I}} = P_1^{\mathbf{I}} \cap P_2^{\mathbf{I}}$
concrete role domain restr.	$P_{C\vert}$	$(P_{C\vert})^{\mathbf{I}} = \{\langle x, y\rangle \in P^{\mathbf{I}} : x \in C^{\mathbf{I}}\}$
concrete role range restr.	$P_{\vert t}$	$(P_{\vert t})^{\mathbf{I}} = \{\langle x, y\rangle \in P^{\mathbf{I}} : y \in t^{\mathbf{D}}\}$
concrete role restriction	$P_{C_1\vert t}$	$(P_{C_1\vert t})^{\mathbf{I}} = \{\langle x, y\rangle \in P^{\mathbf{I}} : x \in C_1^{\mathbf{I}} \wedge y \in t^{\mathbf{D}}\}$
concept subsum.	$C_1 \sqsubseteq C_2$	$\mathbf{I} \models_{\mathbf{D}} C_1 \sqsubseteq C_2 \iff C_1^{\mathbf{I}} \subseteq C_2^{\mathbf{I}}$
ab. role subsum.	$R_1 \sqsubseteq R_2$	$\mathbf{I} \models_{\mathbf{D}} R_1 \sqsubseteq R_2 \iff R_1^{\mathbf{I}} \subseteq R_2^{\mathbf{I}}$
role incl. axiom	$R_1 \ldots R_n \sqsubseteq R$	$\mathbf{I} \models_{\mathbf{D}} R_1 \ldots R_n \sqsubseteq R \iff R_1^{\mathbf{I}} \circ \ldots \circ R_n^{\mathbf{I}} \subseteq R^{\mathbf{I}}$
cn. role subsum.	$P_1 \sqsubseteq P_2$	$\mathbf{I} \models_{\mathbf{D}} P_1 \sqsubseteq P_2 \iff P_1^{\mathbf{I}} \subseteq P_2^{\mathbf{I}}$
symmetric role	$\mathsf{Sym}(R)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Sym}(R) \iff (R^-)^{\mathbf{I}} \subseteq R^{\mathbf{I}}$
asymmetric role	$\mathsf{Asym}(R)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Asym}(R) \iff R^{\mathbf{I}} \cap (R^-)^{\mathbf{I}} = \emptyset$
transitive role	$\mathsf{Tra}(R)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Tra}(R) \iff R^{\mathbf{I}} \circ R^{\mathbf{I}} \subseteq R^{\mathbf{I}}$
disj. ab. role	$\mathsf{Dis}(R_1, R_2)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Dis}(R_1, R_2) \iff R_1^{\mathbf{I}} \cap R_2^{\mathbf{I}} = \emptyset$
reflexive role	$\mathsf{Ref}(R)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Ref}(R) \iff \{\langle x, x\rangle \mid x \in \Delta^{\mathbf{I}}\} \subseteq R^{\mathbf{I}}$
irreflexive role	$\mathsf{Irref}(R)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Irref}(R) \iff R^{\mathbf{I}} \cap \{\langle x, x\rangle \mid x \in \Delta^{\mathbf{I}}\} = \emptyset$
func. ab. role	$\mathsf{Fun}(R)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Fun}(R) \iff (R^-)^{\mathbf{I}} \circ R^{\mathbf{I}} \subseteq \{\langle x, x\rangle \mid x \in \Delta^{\mathbf{I}}\}$
disj. cn. role	$\mathsf{Dis}(P_1, P_2)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Dis}(P_1, P_2) \iff P_1^{\mathbf{I}} \cap P_2^{\mathbf{I}} = \emptyset$
func. cn. role	$\mathsf{Fun}(P)$	$\mathbf{I} \models_{\mathbf{D}} \mathsf{Fun}(p) \iff \langle x, y\rangle \in P^{\mathbf{I}}$ and $\langle x, z\rangle \in P^{\mathbf{I}}$ imply $y = z$
datatype terms equivalence	$t_1 \equiv t_2$	$\mathbf{I} \models_{\mathbf{D}} t_1 \equiv t_2 \iff t_1^{\mathbf{D}} = t_2^{\mathbf{D}}$
datatype terms diseq.	$t_1 \not\equiv t_2$	$\mathbf{I} \models_{\mathbf{D}} t_1 \not\equiv t_2 \iff t_1^{\mathbf{D}} \neq t_2^{\mathbf{D}}$
datatype terms subsum.	$t_1 \sqsubseteq t_2$	$\mathbf{I} \models_{\mathbf{D}} (t_1 \sqsubseteq t_2) \iff t_1^{\mathbf{D}} \subseteq t_2^{\mathbf{D}}$
concept assertion	$a : C_1$	$\mathbf{I} \models_{\mathbf{D}} a : C_1 \iff (a^{\mathbf{I}} \in C_1^{\mathbf{I}})$
agreement	$a = b$	$\mathbf{I} \models_{\mathbf{D}} a = b \iff a^{\mathbf{I}} = b^{\mathbf{I}}$
disagreement	$a \neq b$	$\mathbf{I} \models_{\mathbf{D}} a \neq b \iff \neg(a^{\mathbf{I}} = b^{\mathbf{I}})$
ab. role asser.	$(a, b) : R$	$\mathbf{I} \models_{\mathbf{D}} (a, b) : R \iff \langle a^{\mathbf{I}}, b^{\mathbf{I}}\rangle \in R^{\mathbf{I}}$
cn. role asser.	$(a, e_d) : P$	$\mathbf{I} \models_{\mathbf{D}} (a, e_d) : P \iff \langle a^{\mathbf{I}}, e_d^{\mathbf{D}}\rangle \in P^{\mathbf{I}}$

Table 1: Semantics of $\mathcal{DL}_{\mathbf{D}}^{4,\times}$.

Legenda. ab: abstract, cn.: concrete, rl.: role, ind.: individual, d. cs.: datatype constant, dtype: datatype, ng.: negated, bot.: bottom, incl.: inclusion, asser.: assertion.

Let $\mathcal{R}$, $\mathcal{T}$, and $\mathcal{A}$ be as above. An interpretation $\mathbf{I} = (\Delta^{\mathbf{I}}, \Delta_{\mathbf{D}}, \cdot^{\mathbf{I}})$ is a **D**-model of $\mathcal{R}$ (resp., $\mathcal{T}$), and we write $\mathbf{I} \models_{\mathbf{D}} \mathcal{R}$ (resp., $\mathbf{I} \models_{\mathbf{D}} \mathcal{T}$), if $\mathbf{I}$ satisfies each axiom in $\mathcal{R}$ (resp., $\mathcal{T}$) according to the semantic rules in Table 1. Analogously, $\mathbf{I} = (\Delta^{\mathbf{I}}, \Delta_{\mathbf{D}}, \cdot^{\mathbf{I}})$ is a **D**-model of $\mathcal{A}$, and we write $\mathbf{I} \models_{\mathbf{D}} \mathcal{A}$, if $\mathbf{I}$ satisfies each assertion in $\mathcal{A}$, according to the semantic rules in Table 1.

A $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge base $\mathcal{K} = (\mathcal{A}, \mathcal{T}, \mathcal{R})$ is consistent if there is an interpretation $\mathbf{I} = (\Delta^{\mathbf{I}}, \Delta_{\mathbf{D}}, \cdot^{\mathbf{I}})$ that is a **D**-model of $\mathcal{A}$, $\mathcal{T}$, and $\mathcal{R}$.

Decidability of the consistency problem for $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge bases was proved in [4] via a reduction to the satisfiability problem for formulae of a four level quantified syllogistic called $\mathsf{4LQS^R}$. The latter problem was proved decidable in [7].

3 ABox Reasoning services for $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ knowledge base

The most important feature of a knowledge representation system is the capability of providing reasoning services. Depending on the type of the application domains, there are many different kinds of implicit knowledge that is desirable to infer from what is explicitly mentioned in the knowledge base. In particular, reasoning problems regarding ABox consist in querying a knowledge base in order to retrieve information concerning data stored in it. In this section we study the decidability for the most widespread ABox reasoning tasks for the logic $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ resorting to a general problem, called Higher Order Conjuctive Query Answering (HOCQA), that can be instantiated to each of them.

Let $\mathsf{V_i} = \{\mathsf{v}_1, \mathsf{v}_2, \ldots\}$, $\mathsf{V_c} = \{\mathsf{c}_1, \mathsf{c}_2, \ldots\}$, $\mathsf{V_{ar}} = \{\mathsf{r}_1, \mathsf{r}_2, \ldots\}$, and $\mathsf{V_{cr}} = \{\mathsf{p}_1, \mathsf{p}_2, \ldots\}$ be denumerable and infinite sets of variables pairwise disjoint and disjoint from **Ind**, from $\bigcup\{N_C(d) : d \in N_{\mathbf{D}}\}$, from **C**, from $\mathbf{R_A}$, and from $\mathbf{R_D}$. A HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*atomic formula* is an expression of one of the following types: $R(w_1, w_2)$, $P(w_1, u_1)$, $C(w_1)$, $\mathsf{r}(w_1, w_2)$, $\mathsf{p}(w_1, u_1)$, $\mathsf{c}(w_1)$, $w_1 = w_2$, $u_1 = u_2$, where $w_1, w_2 \in \mathsf{V_i} \cup \mathbf{Ind}$, $u_1, u_2 \in \mathsf{V_i} \cup \bigcup\{N_C(d) : d \in N_{\mathbf{D}}\}$, R is a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-abstract role term, P is a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concrete role term, C is a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-concept term, $\mathsf{r} \in \mathsf{V_{ar}}$, $\mathsf{p} \in \mathsf{V_{cr}}$, and $\mathsf{c} \in \mathsf{V_c}$. A HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-atomic formula containing no variables is said to be *ground*. A HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*literal* is a HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-atomic formula or its negation. A HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-*conjunctive query* is a conjunction of HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-literals. We denote with λ the *empty* HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive query.

Let $\mathsf{v}_1, \ldots, \mathsf{v}_n \in \mathsf{V_i}$, $\mathsf{c}_1, \ldots, \mathsf{c}_m \in \mathsf{V_c}$, $\mathsf{r}_1, \ldots, \mathsf{r}_k \in \mathsf{V_{ar}}$, $\mathsf{p}_1, \ldots, \mathsf{p}_h \in \mathsf{V_{cr}}$, $o_1, \ldots, o_n \in \mathbf{Ind} \cup \bigcup\{N_C(d) : d \in N_{\mathbf{D}}\}$, $C_1, \ldots, C_m \in \mathbf{C}$, $R_1, \ldots, R_k \in \mathbf{R_A}$, and $P_1, \ldots, P_h \in \mathbf{R_D}$. A substitution

$$\sigma \coloneqq \{\mathsf{v}_1/o_1, \ldots, \mathsf{v}_n/o_n, \mathsf{c}_1/C_1, \ldots, \mathsf{c}_m/C_m, \mathsf{r}_1/R_1, \ldots, \mathsf{r}_k/R_k, \mathsf{p}_1/P_1, \ldots, \mathsf{p}_h/P_h\}$$

is a map such that, for every HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-literal L, $L\sigma$ is obtained from L by replacing

– the occurrences of $\mathsf{v}_1, \ldots, \mathsf{v}_n$ in L with $o_1, \ldots, o_n$, respectively;
– the occurrences of $\mathsf{c}_1, \ldots, \mathsf{c}_m$ in L with $C_1, \ldots, C_m$, respectively;
– the occurrences of $\mathsf{r}_1, \ldots, \mathsf{r}_k$ in L with $R_1, \ldots, R_k$, respectively;
– the occurrences of $\mathsf{p}_1, \ldots, \mathsf{p}_h$ in L with $P_1, \ldots, P_h$, respectively.

Substitutions can be extended to HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive queries in the usual way. Let $Q \coloneqq (L_1 \wedge \ldots \wedge L_m)$ be a HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive query, and $\mathcal{KB}$ a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge base. A substitution σ involving *exactly* the variables occurring in Q is a *solution for Q w.r.t. $\mathcal{KB}$* if there exists a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-interpretation $\mathbf{I}$ such that $\mathbf{I} \models_{\mathbf{D}} \mathcal{KB}$ and $\mathbf{I} \models_{\mathbf{D}} Q\sigma$. The collection Σ of the solutions for Q w.r.t. $\mathcal{KB}$ is the *higher order (HO) answer set of Q w.r.t. $\mathcal{KB}$*. Then the *HO conjunctive query answering* (HOCQA) problem for Q w.r.t. $\mathcal{KB}$ consists in finding the HO answer set Σ of Q w.r.t. $\mathcal{KB}$. We shall solve the HOCQA problem just stated by reducing it to the analogous problem formulated in the context of the fragment $\mathsf{4LQS^R}$ (and in turn to the decision procedure for $\mathsf{4LQS^R}$ presented in [7]). The HOCQA problem for $\mathsf{4LQS^R}$-formulae can be stated as follows. Let ϕ be a $\mathsf{4LQS^R}$-formula and let ψ be a conjunction of $\mathsf{4LQS^R}$-quantifier-free atomic formulae of level 0 of the types $x = y$, $x \in X^1$, $\langle x, y\rangle \in X^3$, or their negations.

The *HOCQA problem for ψ w.r.t. ϕ* consists in computing the *HO answer set of ψ w.r.t. ϕ*, namely the collection Σ' of all the substitutions σ' such that $\boldsymbol{\mathcal{M}} \models \phi \wedge \psi\sigma'$, for some $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$.

In view of the decidability of the satisfiability problem for $\mathsf{4LQS^R}$-formulae, the HOCQA problem for $\mathsf{4LQS^R}$-formulae is decidable as well.

Indeed, given two $\mathsf{4LQS^R}$-formulae ϕ and ψ satisfying the above requirements, to compute the HO answer set of ψ w.r.t. ϕ, for each candidate substitution $\sigma' \coloneqq \{\boldsymbol{x}/\boldsymbol{z}, \boldsymbol{X^1}/\boldsymbol{Y^1}, \boldsymbol{X^2}/\boldsymbol{Y^2}, \boldsymbol{X^3}/\boldsymbol{Y^3}\}$ one has just to test for satisfiability of the $\mathsf{4LQS^R}$-formula $\phi \wedge \psi\sigma'$. Since the number of possible candidate substitutions is $\left|\bigcup_{i=0}^{3} \mathtt{Var}_i(\phi)\right|^{\left|\bigcup_{i=0}^{3} \mathtt{Var}_i(\psi)\right|}$ and the satisfiability problem for $\mathsf{4LQS^R}$-formulae is decidable, the HO answer set of ψ w.r.t. ϕ can be computed effectively. Summarizing,

Lemma 1. *The HOCQA problem for $\mathsf{4LQS^R}$-formulae is decidable.* □

The following theorem states that also the HOCQA problem for $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ is decidable.

Theorem 1. *Given a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge base $\mathcal{KB}$ and a HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$- conjunctive query Q, the HOCQA problem for Q w.r.t. $\mathcal{KB}$ is decidable.*

Proof. We first outline the main ideas and then we provide a formal proof of the theorem.

In order to define a $\mathsf{4LQS^R}$*formula* $\phi_{\mathcal{KB}}$, we recall the definition a function θ that maps the $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge base $\mathcal{KB}$ in the $\mathsf{4LQS^R}$-formula in Conjunctive Normal Form (CNF) $\phi_{\mathcal{KB}}$, introduced in [10]. The definition of the mapping θ is inspired to the definition of the mapping τ introduced in the proof of Theorem 1 in [4]. Specifically, θ differs from τ because it allows quantification only on variables of level 0, it treats Boolean operations on concrete roles and the product of concepts, and it constructs $4LQS^R$-formulae in CNF. To prepare for the definition of θ, we map injectively individuals $a \in \mathbf{Ind}$ and constants $e_d \in N_C(d)$ into level 0 variables x_a, x_{e_d}, the constant concepts $\top$ and $\bot$, datatype terms t, and concept terms C into level 1 variables $X^1_\top$, $X^1_\bot$, X^1_t, X^1_C, respectively, and the universal relation on individuals U, abstract role terms R, and concrete role terms P into level 3 variables X^3_U, X^3_R, and X^3_P, respectively.[2]

Then the mapping θ is defined as follows:

$\theta(C_1 \equiv \top) \coloneqq (\forall z)((\neg(z \in X^1_{C_1}) \vee z \in X^1_\top) \wedge (\neg(z \in X^1_\top) \vee z \in X^1_{C_1})),$
$\theta(C_1 \equiv \neg C_2) \coloneqq (\forall z)((\neg(z \in X^1_{C_1}) \vee \neg(z \in X^1_{C_2})) \wedge (z \in X^1_{C_2} \vee z \in X^1_{C_1})),$
$\theta(C_1 \equiv C_2 \sqcup C_3) \coloneqq (\forall z)((\neg(z \in X^1_{C_1}) \vee (z \in X^1_{C_2} \vee z \in X^1_{C_3})) \wedge ((\neg(z \in X^1_{C_2}) \vee z \in X^1_{C_1}) \wedge (\neg(z \in X^1_{C_3}) \vee z \in X^1_{C_1})),$
$\theta(C_1 \equiv \{a\}) \coloneqq (\forall z)(\neg(z \in X^1_{C_1}) \vee z = x_a) \wedge (\neg(z = x_a) \vee z \in X^1_{C_1}),$
$\theta(C_1 \sqsubseteq \forall R_1.C_2) \coloneqq (\forall z_1)(\forall z_2)(\neg(z_1 \in X^1_{C_1}) \vee (\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee z_2 \in X^1_{C_2})),$
$\theta(\exists R_1.C_1 \sqsubseteq C_2) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee \neg(z_2 \in X^1_{C_1})) \vee z_1 \in X^1_{C_2}),$
$\theta(C_1 \equiv \exists R_1.\{a\}) \coloneqq (\forall z)((\neg(z \in X^1_{C_1}) \vee \langle z, x_a\rangle \in X^3_{R_1}) \wedge (\neg(\langle z, x_a\rangle \in X^3_{R_1}) \vee z \in X^1_{C_1})),$
$\theta(C_1 \sqsubseteq \leq_n R_1.C_2) \coloneqq (\forall z)(\forall z_1)\ldots(\forall z_{n+1})(\neg(z \in X^1_{C_1}) \vee (\bigwedge_{i=1}^{n+1}(\neg(z_i \in X_{C_2}) \vee \neg(\langle z, z_i\rangle \in X^3_{R_1}) \vee \bigvee_{i<j} z_i = z_j)),$
$\theta(\geq_n R_1.C_1 \sqsubseteq C_2) \coloneqq (\forall z)(\forall z_1)\ldots(\forall z_n)(\bigwedge_{i=1}^{n}((\neg(z_i \in X^1_{C_1}) \vee \neg(\langle z, z_i\rangle \in X^3_{R_1})) \vee \bigvee_{i<j} z_i = z_j) \vee z \in X^1_{C_2}),$
$\theta(C_1 \sqsubseteq \forall P_1.t_1) \coloneqq (\forall z_1)(\forall z_2)(\neg(z_1 \in X^1_{C_1}) \vee (\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee z_2 \in X^1_{t_1})),$
$\theta(\exists P_1.t_1 \sqsubseteq C_1) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee \neg(z_2 \in X^1_{t_1})) \vee z_1 \in X^1_{C_1}),$
$\theta(C_1 \equiv \exists P_1.\{e_d\}) \coloneqq (\forall z)((\neg(z \in X^1_{C_1}) \vee \langle z, x_{e_d}\rangle \in X^3_{P_1}) \wedge (\neg(\langle z, x_{e_d}\rangle \in X^3_{P_1}) \vee z \in X^1_{C_1})),$
$\theta(C_1 \sqsubseteq \leq_n P_1.t_1) \coloneqq (\forall z)(\forall z_1)\ldots(\forall z_{n+1})(\neg(z \in X^1_{C_1}) \vee (\bigwedge_{i=1}^{n+1}(\neg(z_i \in X_{t_1}) \vee \neg(\langle z, z_i\rangle \in X^3_{P_1}) \vee \bigvee_{i<j} z_i = z_j)),$
$\theta(\geq_n P_1.t_1 \sqsubseteq C_1) \coloneqq (\forall z)(\forall z_1)\ldots(\forall z_n)(\bigwedge_{i=1}^{n}((\neg(z_i \in X^1_{t_1}) \vee \neg(\langle z, z_i\rangle \in X^3_{P_1})) \vee \bigvee_{i<j} z_i = z_j) \vee z \in X^1_{C_1}),$
$\theta(R_1 \equiv U) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee \langle z_1, z_2\rangle \in X^3_U) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_U) \vee \langle z_1, z_2\rangle \in X^3_{R_1})),$
$\theta(R_1 \equiv \neg R_2) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee \neg(\langle z_1, z_2\rangle \in X^3_{R_2})) \wedge (\langle z_1, z_2\rangle \in X^3_{R_2} \vee \neg(\langle z_1, z_2\rangle \in X^3_{R_1}))),$
$\theta(R \equiv C_1 \times C_2) \coloneqq (\forall z_1)(\forall z_2)(\neg(\langle z_1, z_2\rangle \in X^3_R) \vee z_1 \in X^1_{C_1}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_R) \vee z_2 \in X^1_{C_2}) \wedge ((\neg(z_1 \in X^1_{C_1}) \vee \neg(z_2 \in X^1_{C_2})) \vee \langle z_1, z_2\rangle \in X^3_R))$

[2] The use of level 3 variables to model abstract and concrete role terms is motivated by the fact that their elements, that is ordered pairs $\langle x, y\rangle$, are encoded in Kuratowski's style as $\{\{x\}, \{x, y\}\}$, namely as collections of sets of objects.

$\theta(R_1 \equiv R_2 \sqcup R_3) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee (\langle z_1, z_2\rangle \in X^3_{R_2} \vee \langle z_1, z_2\rangle \in X^3_{R_3})) \wedge ((\neg(\langle z_1, z_2\rangle \in X^3_{R_2}) \vee \langle z_1, z_2\rangle \in X^3_{R_1}) \wedge ((\neg(\langle z_1, z_2\rangle \in X^3_{R_3}) \vee \langle z_1, z_2\rangle \in X^3_{R_1})))),$
$\theta(R_1 \equiv R_2^-) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee \langle z_2, z_1\rangle \in X^3_{R_2}) \wedge (\neg(\langle z_2, z_1\rangle \in X^3_{R_2}) \vee \langle z_1, z_2\rangle \in X^3_{R_1})),$
$\theta(R_1 \equiv id(C_1)) \coloneqq (\forall z_1)(\forall z_2)(((\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee z_1 \in X^1_{C_1}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee z_2 \in X^1_{C_1}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee z_1 = z_2)) \wedge ((\neg(z_1 \in X^1_{C_1}) \vee \neg(z_2 \in X^1_{C_1}) \vee z_1 \neq z_2) \vee \langle z_1, z_2\rangle \in X^3_{R_1})),$
$\theta(R_1 \equiv R_{2_{C_1|}}) \coloneqq (\forall z_1)(\forall z_2)(((\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee \langle z_1, z_2\rangle \in X^3_{R_2}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee z_1 \in X^1_{C_1})) \wedge ((\neg(\langle z_1, z_2\rangle \in X^3_{R_2}) \vee \neg(z_1 \in X^1_{C_1})) \vee \langle z_1, z_2\rangle \in X^3_{R_1})),$
$\theta(R_1 \ldots R_n \sqsubseteq R_{n+1}) \coloneqq (\forall z)(\forall z_1)\ldots(\forall z_n)((\neg(\langle z, z_1\rangle \in X^3_{R_1}) \vee \ldots \vee \neg(\langle z_{n-1}, z_n\rangle \in X^3_{R_n})) \vee \langle z, z_n\rangle \in X^3_{R_{n+1}}),$
$\theta(\mathsf{Ref}(R_1)) \coloneqq (\forall z)(\langle z, z\rangle \in X^3_{R_1}),$
$\theta(\mathsf{Irref}(R_1)) \coloneqq (\forall z)(\neg(\langle z, z\rangle \in X^3_{R_1})),$
$\theta(\mathsf{Fun}(R_1)) \coloneqq (\forall z_1)(\forall z_2)(\forall z_3)((\neg(\langle z_1, z_2\rangle \in X^3_{R_1}) \vee \neg(\langle z_1, z_3\rangle \in X^3_{R_1})) \vee z_2 = z_3),$
$\theta(P_1 \equiv P_2) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee \langle z_1, z_2\rangle \in X^3_{P_2}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{P_2}) \vee \langle z_1, z_2\rangle \in X^3_{P_1})),$
$\theta(P_1 \equiv \neg P_2) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee \neg(\langle z_1, z_2\rangle \in X^3_{P_2})) \wedge (\langle z_1, z_2\rangle \in X^3_{P_2} \vee \langle z_1, z_2\rangle \in X^3_{P_1})),$
$\theta(P_1 \sqsubseteq P_2) \coloneqq (\forall z_1)(\forall z_2)(\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee \langle z_1, z_2\rangle \in X^3_{P_2}),$
$\theta(\mathsf{Fun}(P_1)) \coloneqq (\forall z_1)(\forall z_2)(\forall z_3)((\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee \neg(\langle z_1, z_3\rangle \in X^3_{P_1}) \vee z_2 = z_3),$
$\theta(P_1 \equiv P_{2_{C_1|}}) \coloneqq (\forall z_1)(\forall z_2)(((\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee \langle z_1, z_2\rangle \in X^3_{P_2}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee z_1 \in X^1_{C_1}) \wedge ((\neg\langle z_1, z_2\rangle \in X^3_{P_2}) \vee \neg(z_1 \in X^1_{C_1}) \vee \langle z_1, z_2\rangle \in X^3_{P_1})),$
$\theta(P_1 \equiv P_{2_{|t_1}}) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee \langle z_1, z_2\rangle \in X^3_{P_2}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee z_2 \in X^1_{t_1}) \wedge ((\neg(\langle z_1, z_2\rangle \in X^3_{P_2}) \vee \neg(z_2 \in X^1_{t_1})) \vee \langle z_1, z_2\rangle \in X^3_{P_1})),$
$\theta(P_1 \equiv P_{2_{C_1|t_1}}) \coloneqq (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee \langle z_1, z_2\rangle \in X^3_{P_2}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee z_1 \in X^1_{C_1}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{P_1}) \vee z_2 \in X^1_{t_1}) \wedge (\neg(\langle z_1, z_2\rangle \in X^3_{P_2}) \vee \neg(z_1 \in X^1_{C_1}) \vee \neg(z_2 \in X^1_{t_1}) \vee \langle z_1, z_2\rangle \in X^3_{P_1})),$
$\theta(t_1 \equiv t_2) \coloneqq (\forall z)((\neg(z \in X^1_{t_1}) \vee z \in X^1_{t_2}) \wedge (\neg(z \in X^1_{t_2}) \vee z \in X^1_{t_1}))$, $\theta(t_1 \equiv \neg t_2) \coloneqq (\forall z)((\neg(z \in X^1_{t_1}) \vee \neg(z \in X^1_{t_2})) \wedge (z \in X^1_{t_2} \vee z \in X^1_{t_1})),$
$\theta(t_1 \equiv t_2 \sqcup t_3) \coloneqq (\forall z)((\neg(z \in X^1_{t_1}) \vee (z \in X^1_{t_2} \vee z \in X^1_{t_3})) \wedge ((\neg(z \in X^1_{t_2}) \vee z \in X^1_{t_1}) \wedge (\neg(z \in X^1_{t_3}) \vee z \in X^1_{t_1})))),$
$\theta(t_1 \equiv t_2 \sqcap t_3) \coloneqq (\forall z)((\neg(z \in X^1_{t_1}) \vee (z \in X^1_{t_2} \wedge z \in X^1_{t_3})) \wedge (((\neg(z \in X^1_{t_2}) \vee \neg(z \in X^1_{t_3})) \vee z \in X^1_{t_1})),$
$\theta(t_1 \equiv \{e_d\}) \coloneqq (\forall z)((\neg(z \in X^1_{t_1}) \vee z = x_{e_d}) \wedge (\neg(z = x_{e_d}) \vee z \in X^1_{t_1})),$
$\theta(a : C_1) \coloneqq x_a \in X^1_{C_1},$
$\theta((a,b) : R_1) \coloneqq \langle x_a, x_b\rangle \in X^3_{R_1},$
$\theta((a,b) : \neg R_1) \coloneqq \neg(\langle x_a, x_b\rangle \in X^3_{R_1}),$
$\theta(a = b) \coloneqq x_a = x_b$, $\theta(a \neq b) \coloneqq \neg(x_a = x_b),$
$\theta(e_d : t_1) \coloneqq x_{e_d} \in X^1_{t_1},$
$\theta((a, e_d) : P_1) \coloneqq \langle x_a, x_{e_d}\rangle \in X^3_{P_1}$, $\theta((a, e_d) : \neg P_1) \coloneqq \neg(\langle x_a, x_{e_d}\rangle \in X^3_{P_1}),$
$\theta(\alpha \wedge \beta) \coloneqq \theta(\alpha) \wedge \theta(\beta).$

Let $\mathcal{KB}$ be our $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-knowledge base, and let $\mathsf{cpt}_{\mathcal{KB}}$, $\mathsf{arl}_{\mathcal{KB}}$, $\mathsf{crl}_{\mathcal{KB}}$, and $\mathsf{ind}_{\mathcal{KB}}$ be, respectively, the sets of concept, of abstract role, of concrete role, and of individual names in $\mathcal{KB}$. Moreover, let $N_D^{\mathcal{KB}} \subseteq N_D$ be the set of datatypes in

$\mathcal{KB}$, $N_F^{\mathcal{KB}}$ a restriction of N_F assigning to every $d \in N_{\mathbf{D}}^{\mathcal{KB}}$ the set $N_F^{\mathcal{KB}}(d)$ of facets in $N_F(d)$ and in $\mathcal{KB}$. Analogously, let $N_C^{\mathcal{KB}}$ be a restriction of the function N_C associating to every $d \in N_{\mathbf{D}}^{\mathcal{KB}}$ the set $N_C^{\mathcal{KB}}(d)$ of constants contained in $N_C(d)$ and in $\mathcal{KB}$. Finally, for every datatype $d \in N_D^{\mathcal{KB}}$, let $\mathsf{bf}_{\mathcal{KB}}^{\mathbf{D}}(d)$ be the set of facet expressions for d occurring in $\mathcal{KB}$ and not in $N_F(d) \cup \{\top^d, \bot_d\}$. We assume without loss of generality that the facet expressions in $\mathsf{bf}_{\mathcal{KB}}^{\mathbf{D}}(d)$ are in Conjunctive Normal Form. We define the $\mathsf{4LQS^R}$-formula $\phi_{\mathcal{KB}}$ expressing the consistency of $\mathcal{KB}$ as follows:

$$\phi_{\mathcal{KB}} \coloneqq \bigwedge_{H \in \mathcal{KB}} \theta(H) \wedge \bigwedge_{i=1}^{12} \xi_i \,,$$

where

$$\xi_1 \coloneqq (\forall z)((\neg(z \in X_{\mathbf{I}}^1) \vee \neg(z \in X_{\mathbf{D}}^1)) \wedge (z \in X_{\mathbf{D}}^1 \vee z \in X_{\mathbf{I}}^1)) \wedge (\forall z)(z \in X_{\mathbf{I}}^1 \vee z \in X_{\mathbf{D}}^1) \wedge \neg(\forall z)\neg(z \in X_{\mathbf{I}}^1) \wedge \neg(\forall z)\neg(z \in X_{\mathbf{D}}^1),$$

$$\xi_2 \coloneqq ((\forall z)((\neg(z \in X_{\mathbf{I}}^1) \vee z \in X_{\top}^1) \wedge (\neg(z \in X_{\top}^1) \vee z \in X_{\mathbf{I}}^1)) \wedge (\forall z)\neg(z \in X_{\bot}),$$

$$\xi_3 \coloneqq \bigwedge_{A \in \mathsf{cpt}_{\mathcal{KB}}} (\forall z)(\neg(z \in X_A^1) \vee z \in X_{\mathbf{I}}^1),$$

$$\xi_4 \coloneqq (\bigwedge_{d \in N_D^{\mathcal{KB}}} ((\forall z)(\neg(z \in X_d^1) \vee z \in X_{\mathbf{D}}^1) \wedge \neg(\forall z)\neg(z \in X_d^1)) \wedge (\forall z) (\bigwedge_{(d_i, d_j \in N_D^{\mathcal{KB}}, i<j)} ((\neg(z \in X_{d_i}^1) \vee \neg(z \in X_{d_j}^1)) \wedge (z \in X_{d_j}^1 \vee z \in X_{d_i}^1)))),$$

$$\xi_5 \coloneqq \bigwedge_{d \in N_D^{\mathcal{KB}}} ((\forall z)((\neg(z \in X_d^1) \vee z \in X_{\top_d}^1) \wedge (\neg(z \in X_{\top_d}^1) \vee z \in X_d^1) \wedge (\forall z)\neg(z \in X_{\bot_d}^1)),$$

$$\xi_6 \coloneqq \bigwedge_{\substack{f_d \in N_F^{\mathcal{KB}}(d),\\ d \in N_D^{\mathcal{KB}}}} (\forall z)(\neg(z \in X_{f_d}^1) \vee z \in X_d^1),$$

$$\xi_7 \coloneqq (\forall z_1)(\forall z_2)((\neg(z_1 \in X_{\mathbf{I}}^1) \vee \neg(z_2 \in X_{\mathbf{I}}^1) \vee \langle z_1, z_2 \rangle \in X_U^3) \wedge ((\neg(\langle z_1, z_2 \rangle \in X_U^3) \vee z_1 \in X_{\mathbf{I}}^1) \wedge (\neg(\langle z_1, z_2 \rangle \in X_U^3) \vee z_2 \in X_{\mathbf{I}}^1))),$$

$$\xi_8 \coloneqq \bigwedge_{R \in \mathsf{arl}_{\mathcal{KB}}} (\forall z_1)(\forall z_2)((\neg(\langle z_1, z_2 \rangle \in X_R^3) \vee z_1 \in X_{\mathbf{I}}^1) \wedge (\neg(\langle z_1, z_2 \rangle \in X_R^3) \vee z_2 \in X_{\mathbf{I}}^1))),$$

$$\xi_9 \coloneqq \bigwedge_{T \in \mathsf{crl}_{\mathcal{KB}}} (\forall z_1)(\forall z_2)(\neg(\langle z_1, z_2 \rangle \in X_T^3) \vee z_1 \in X_{\mathbf{I}}^1) \wedge (\neg(\langle z_1, z_2 \rangle \in X_T^3) \vee z_2 \in X_{\mathbf{D}}^1))),$$

$$\xi_{10} \coloneqq \bigwedge_{a \in \mathsf{ind}_{\mathcal{KB}}} (x_a \in X_{\mathbf{I}}^1) \wedge \bigwedge_{\substack{d \in N_D^{\mathcal{KB}},\\ e_d \in N_C^{\mathcal{KB}}(d)}} x_{e_d} \in X_d^1,$$

$$\xi_{11} \coloneqq \bigwedge_{\{e_{d_1},\ldots,e_{d_n}\} \text{ in } \mathcal{KB}} (\forall z)((\neg(z \in X^1_{\{e_{d_1},\ldots,e_{d_n}\}}) \vee \bigvee_{i=1}^{n} (z = x_{e_{d_i}})) \wedge (\bigwedge_{i=1}^{n} (z \neq x_{e_{d_i}} \vee z \in X^1_{\{e_{d_1},\ldots,e_{d_n}\}}))) \wedge \bigwedge_{\{a_1,\ldots,a_n\} \text{ in } \mathcal{KB}} (\forall z)((\neg(z \in X^1_{\{a_1,\ldots,a_n\}}) \vee \bigvee_{i=1}^{n} (z = x_{a_i})) \wedge (\bigwedge_{i=1}^{n} (z \neq x_{a_i} \vee z \in X^1_{\{a_1,\ldots,a_n\}}))),$$

$$\xi_{12} \coloneqq \bigwedge_{\substack{d \in N^{\mathcal{KB}}_{\mathbf{D}},\\ \psi_d \in \mathsf{bf}^{\mathbf{D}}_{\mathcal{KB}}(d)}} (\forall z)(\neg(z \in X^1_{\psi_d}) \vee z \in \zeta(X^1_{\psi_d})) \wedge (\neg(z \in \zeta(X^1_{\psi_d})) \vee z \in X^1_{\psi_d})$$

with ζ the transformation function from $\mathsf{4LQS^R}$-variables of level 1 to $\mathsf{4LQS^R}$-formulae recursively defined, for $d \in N^{\mathcal{KB}}_{\mathbf{D}}$, by

$$\zeta(X^1_{\psi_d}) \coloneqq \begin{cases} X^1_{\psi_d} & \text{if } \psi_d \in N^{\mathcal{KB}}_F(d) \cup \{\top^d, \bot_d\} \\ \neg\zeta(X^1_{\chi_d}) & \text{if } \psi_d = \neg\chi_d \\ \zeta(X^1_{\chi_d}) \wedge \zeta(X^1_{\varphi_d}) & \text{if } \psi_d = \chi_d \wedge \varphi_d \\ \zeta(X^1_{\chi_d}) \vee \zeta(X^1_{\varphi_d}) & \text{if } \psi_d = \chi_d \vee \varphi_d \,. \end{cases}$$

In the above formulae, the variable $X^1_{\mathbf{I}}$ denotes the set of individuals **Ind**, X^1_d a datatype $d \in N^{\mathcal{KB}}_D$, $X^1_{\mathbf{D}}$ a superset of the union of datatypes in $N^{\mathcal{KB}}_D$, $X^1_{\top_d}$ and $X^1_{\bot_d}$ the constants $\top_d$ and $\bot_d$, and $X^1_{f_d}$, $X^1_{\psi_d}$ a facet f_d and a facet expression ψ_d, for $d \in N^{\mathcal{KB}}_D$, respectively. In addition, X^1_A, X^3_R, X^3_T denote a concept name A, an abstract role name R, and a concrete role name T occurring in $\mathcal{KB}$, respectively. Finally, $X^1_{\{e_{d_1},\ldots,e_{d_n}\}}$ denotes a data range $\{e_{d_1},\ldots,e_{d_n}\}$ occurring in $\mathcal{KB}$, and $X^1_{\{a_1,\ldots,a_n\}}$ a finite set $\{a_1,\ldots,a_n\}$ of nominals in $\mathcal{KB}$.

The constraints $\xi_1 - \xi_{12}$, slightly different from the constraints $\psi_1 - \psi_{12}$ defined in the proof of Theorem 1 in [4], are introduced to guarantee that each model of $\phi_{\mathcal{KB}}$ can be easily transformed in a $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-interpretation.

The HOCQA problem for $\mathcal{DL}^{4,\times}_{\mathbf{D}}$ can be solved via an effective reduction to the HOCQA problem for $\mathsf{4LQS^R}$-formulae, and then exploiting Lemma 1. The reduction is accomplished through the function θ extended in order to map also $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-conjunctive queries into $\mathsf{4LQS^R}$-formulae in conjunctive normal form (CNF), which can be used to map effectively HOCQA problems from the $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-context into the $\mathsf{4LQS^R}$-context. More specifically, given a $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-knowledge base $\mathcal{KB}$ and a $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-HO conjunctive query Q, using the function θ we can effectively construct the following $\mathsf{4LQS^R}$-formulae in CNF:

$$\phi_{\mathcal{KB}} \coloneqq \bigwedge_{H \in \mathcal{KB}} \theta(H) \wedge \bigwedge_{i=1}^{12} \xi_i, \qquad \psi_Q \coloneqq \theta(Q)\,.$$

Then, if we denote by Σ the high order answer set of Q w.r.t. $\mathcal{KB}$ and by Σ' the high order answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$, we have that Σ consists of all substitutions σ (involving exactly the variables occurring in Q) such that $\theta(\sigma) \in \Sigma'$. Since, by Lemma 1, Σ' can be computed effectively, then Σ can be computed effectively too.

The mapping θ is extended for $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-conjuctive queries as follows.

$\theta(R_1(w_1, w_2)) \coloneqq \langle x_{w_1}, x_{w_2}\rangle \in X^3_{R_1}$,
$\theta(P_1(w_1, u_1)) \coloneqq \langle x_{w_1}, x_{u_1}\rangle \in X^3_{P_1}$,
$\theta(C_1(w_1) \coloneqq x_{w_1} \in X^1_{C_1}$,
$\theta(w_1 = w_2) \coloneqq x_{w_1} = x_{w_2}$,
$\theta(u_1 = u_2) \coloneqq x_{u_1} = x_{u_2}$.
$\theta(\mathsf{c}_1(w_1)) \coloneqq w_1 \in X^1_{\mathsf{c}_1}$.
$\theta(\mathsf{r}_1(w_1, w_2)) \coloneqq \langle w_1, w_2\rangle \in X^3_{\mathsf{r}_1}$.
$\theta(\mathsf{p}_1(w_1, u_1)) \coloneqq \langle w_1, u_1\rangle \in X^3_{\mathsf{p}_1}$.

To complete, we extend the mapping θ on substitutions

$$\sigma \coloneqq \{\mathsf{v}_1/o_1, \dots \mathsf{v}_n/o_n, \mathsf{c}_1/C_1, \dots, \mathsf{c}_m/C_m, \mathsf{r}_1/R_1, \dots, \mathsf{r}_k, /R_k, \mathsf{p}_1/P_1, \dots \mathsf{p}_h/P_h\}$$

with $\mathsf{v}_1, \dots, \mathsf{v}_n \in \mathsf{V_i}$, $\mathsf{c}_1, \dots, \mathsf{c}_m \in \mathsf{V_c}$, $\mathsf{r}_1, \dots, \mathsf{r}_k \in \mathsf{V_{ar}}$, $\mathsf{p}_1, \dots, \mathsf{p}_h \in \mathsf{V_{cr}}$, $o_1, \dots, o_n \in \mathbf{Ind} \cup \bigcup\{N_C(d) : d \in N_{\mathbf{D}}\}$, $C_1, \dots, C_m \in \mathbf{C}$, $R_1, \dots, R_k \in \mathbf{R_A}$, and $P_1, \dots, P_h \in \mathbf{R_D}$.

We put

$$\begin{aligned}\theta(\sigma) =& \theta(\{\mathsf{v}_1/o_1, \dots \mathsf{v}_n/o_n, \mathsf{c}_1/C_1, \dots, \mathsf{c}_m/C_m, \mathsf{r}_1/R_1, \dots, \mathsf{r}_k, /R_k,\\ &\mathsf{p}_1/P_1, \dots \mathsf{p}_h/P_h\})\\ =&\{x_{\mathsf{v}_1}/x_{o_1}, \dots, x_{\mathsf{v}_n}/x_{o_n}, X^1_{\mathsf{c}_1}/X^1_{C_1}, \dots, X^1_{\mathsf{c}_m}/X^1_{C_m}, X^3_{\mathsf{r}_1}/X^3_{R_1}, \dots, X^3_{\mathsf{r}_k}, /X^3_{R_k},\\ &X^3_{\mathsf{p}_1}/X^3_{P_1}, \dots, X^3_{\mathsf{p}_h}/X^3_{P_h}\}\\ =&\sigma'\end{aligned} \tag{1}$$

where $x_{\mathsf{v}_1}, \dots x_{\mathsf{v}_n}, x_{o_1}, \dots, x_{o_n}$ are variables of level 0, $X^1_{\mathsf{c}_1}, \dots, X^1_{\mathsf{c}_m}, X^1_{C_1}, \dots, X^1_{C_m}$ are variables of level 1, $X^3_{\mathsf{r}_1}, \dots, X^3_{\mathsf{r}_k}, X^3_{\mathsf{p}_1}, \dots, X^3_{\mathsf{p}_h}, X^3_{R_1}, \dots, X^3_{R_k}$, and $X^3_{P_1}, \dots, X^3_{P_h}$ are variables of level 3 in $\mathsf{4LQS^R}$.

To prove the theorem, we show that Σ is the high order answer set for Q w.r.t. $\mathcal{KB}$ iff Σ is equal to $\bigcup_{\boldsymbol{\mathcal{M}} \models \phi_{\mathcal{KB}}} \Sigma'_{\boldsymbol{\mathcal{M}}}$, where $\Sigma'_{\boldsymbol{\mathcal{M}}}$ is the collection of substitutions σ such that $\boldsymbol{\mathcal{M}} \models \psi_Q\sigma$. Let us assume that Σ is high order the answer set for Q w.r.t. $\mathcal{KB}$. We have to show that Σ is equal to $\Sigma' = \bigcup_{\boldsymbol{\mathcal{M}} \models \phi_{\mathcal{KB}}} \Sigma'_{\boldsymbol{\mathcal{M}}}$, where $\Sigma'_{\boldsymbol{\mathcal{M}}}$ is the collection of all the substitutions σ' such that $\boldsymbol{\mathcal{M}} \models \psi_Q\sigma'$.

By contradiction, let us assume that there exists a $\sigma \in \Sigma$ such that $\sigma \notin \Sigma'$, namely $\boldsymbol{\mathcal{M}} \not\models \psi_Q\sigma$, for every $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$ with $\boldsymbol{\mathcal{M}} \models \phi_{\mathcal{KB}}$. Since $\sigma \in \Sigma$ there is a $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-interpretation $\mathbf{I}$ such that $\mathbf{I} \models_{\mathbf{D}} \mathcal{KB}$ and $\mathbf{I} \models_{\mathbf{D}} Q\sigma$. Then, by the construction above, we can define a $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}_{\mathbf{I}}$ such that $\boldsymbol{\mathcal{M}}_{\mathbf{I}} \models \phi_{\mathcal{KB}}$ and $\boldsymbol{\mathcal{M}}_{\mathbf{I}} \models \psi_Q\theta\sigma$. Absurd.

Conversely, let $\sigma' \in \Sigma'$ and assume by contradiction that $\sigma' \notin \Sigma$. Then, for all $\mathcal{DL}^{4,\times}_{\mathbf{D}}$-interpretations such that $\mathbf{I} \models_D \mathcal{KB}$, it holds that $\mathbf{I} \not\models_D Q\sigma'$. Since $\sigma' \in \Sigma'$, there is a $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$ such that $\boldsymbol{\mathcal{M}} \models \phi_{\mathcal{KB}}$ and $\boldsymbol{\mathcal{M}} \models \psi\sigma'$. Then, by the construction above, we can define a $\mathcal{DL}^4_{\mathbf{D}}$-interpretation $\mathbf{I}_{\boldsymbol{\mathcal{M}}}$ such that $\mathbf{I}_{\boldsymbol{\mathcal{M}}} \models_D \mathcal{KB}$ and $\mathbf{I}_{\boldsymbol{\mathcal{M}}} \models_D Q\sigma'$. Absurd. $\square$

In what follows we list the most widespread reasoning services for $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-ABox and we show how to define them as particular cases of the *Higher Order Conjunctive Query Answering* task.

1. *Instance checking*: the problem of deciding whether or not an individual a is an instance of a concept C.
2. *Instance retrieval*: the problem of retrieving all the individuals that are instances of a given concept.
3. *Role filler retrieval*: the problem of retrieving all the fillers x such that the pair (a, x) is an instance of a role R.
4. *Concept retrieval*: the problem of retrieving all concept which an individual is an instance of.
5. *Role instance retrieval*: the problem of retrieving all roles which a pair of individuals (a, b) is an instance of.

The instance checking problem is a specialization of the HOCQA problem admitting HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive queries of the form $Q_{IC} = C(w_1)$, with $w_1 \in \mathbf{Ind}$. The instance retrieval problem is a particular case of the HOCQA problem in which HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive queries have the form $Q_{IR} = C(w_1)$, where w_1 is a variable in $\mathsf{V_i}$. The HOCQA problem can be instantiated to the role filler retrieval problem by admitting HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive queries $Q_{RF} = R(w_1, w_2)$, $w_1 \in \mathbf{Ind}$ and w_2 a variable in $\mathsf{V_i}$. The concept retrieval problem is a specialization of the HOCQA problem allowing HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive queries of the form $Q_{QR} = c(w_1)$, with $w_1 \in Ind$ and c a variable in $\mathsf{V_c}$. Finally, the role instance retrieval problem is a particularization of the HOCQA problem where HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive queries have the form $Q_{RI} = r(w_1, w_2)$, with $w_1, w_2 \in Ind$ and r a variable in $\mathsf{V_{cr}}$.

Notice that the CQA problem for $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ defined in [9] is an instance of the HOCQA problem admitting HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive queries of the form $Q_{CQA} = (L_1 \wedge L_2 \ldots \wedge L_n)$, with L_i an atomic formula of any of the types $R(w_1, w_2)$, $C(w_1)$, or $w_1 = w_2$ (or their negation), where $w_1, w_2 \in (\mathbf{Ind} \cup \mathsf{V_i})$. Notice also that problems 1, 2, and 3 are instances of the CQA problem for $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ while problems 4 and 5 fall outside the definition of CQA. As shown above they can be treated as specializations of HOCQA.

4 An algorithm for the $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ HOCQA problem

In this section we introduce an effective set-theoretic procedure to compute the answer set of an HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive query Q w.r.t. a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ knowledge base $\mathcal{KB}$. Such procedure consists of two separate subprocedures. The first one, introduced in Section 4.1 and called KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ is a KE-tableau based procedure taking as input $\phi_{\mathcal{KB}}$, the $\mathsf{4LQS^R}$-translation of $\mathcal{KB}$ and returning the KE-tableau representing the saturation of $\mathcal{KB}$. The second one, called *HOCQA-*$\mathcal{DL}_{\mathbf{D}}^{4,\times}$, is presented in Section 4.2. It takes as input the KE-tableau constructed by KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ and the $\mathsf{4LQS^R}$-formula ψ_Q, representing the HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive query Q and yields the answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$. More specifically, HOCQA-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ constructs for each open branch of the KE-tableau a decision tree whose leaves are labelled with an element of the answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$.

4.1 Saturation of a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ knowledge base

KE-tableau is a refutation system inspired to Smullyan's semantic tableaux [24]. The main characteristic distinguishing KE-tableau from the latter is the introduction of an analytic cut rule (PB-rule) that permits to reduce inefficiencies of semantic tableaux. Infact, firstly, the classic tableau system can not represent the use of auxiliary lemmas in proofs; secondly, it can not express the bivalence of classical logic. Thirdly, it is extremely inefficient, as it is witnessed by the fact that it can not polynomially simulate the truth-tables. None of these anomalies occurs if the cut rule is permitted. For these reasons, in this paper we present a KE-tableau based procedure that, given a $\mathsf{4LQS^R}$-formula $\phi_{\mathcal{KB}}$ representing a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge base, yields a complete KE-tableau $\mathcal{T}_{\mathcal{KB}}$ for $\phi_{\mathcal{KB}}$. Such procedure is an extension of the system introduced in [9] since it treats $\mathsf{4LQS^R}$-quantifier free atomic formulae of type $x = y$ and $\neg(x = y)$.

Assume without loss of generality that universal quantifiers in $\phi_{\mathcal{KB}}$ occur as inward as possible and that universally quantified variables are pairwise distinct. Let $S_1, \ldots, S_m$ be the conjuncts of $\phi_{\mathcal{KB}}$ that are $\mathsf{4LQS^R}$-purely universal formulae. For every $S_i = (\forall z_1^i) \ldots (\forall z_{n_i}^i)\chi_i$, $i = 1, \ldots, m$, we put

$$Exp(S_i) \coloneqq \bigwedge_{\{x_{a_1}, \ldots, x_{a_{n_i}}\} \subseteq \mathtt{Var}_0(\phi_{\mathcal{KB}})} S_i\{z_1^i / x_{a_1}, \ldots, z_{n_i}^i / x_{a_{n_i}}\}.$$

and $\Phi_{\mathcal{KB}} \coloneqq \{F_j : i = 1, \ldots, k\} \cup \bigcup_{i=1}^{m} Exp(S_i)$, where $F_1, \ldots, F_k$ are the conjuncts of $\phi_{\mathcal{KB}}$ that are $\mathsf{4LQS^R}$-quantifier free atomic formulae.

To prepare for the KE-tableau based procedure to be described next, we introduce some useful notions and notations (see [12] for a detailed overview of KE-tableau, an optimized variant of semantic tableaux).

Let $\Phi = \{C_1, \ldots, C_p\}$ be a collection of disjunctions of $\mathsf{4LQS^R}$-quantifier free atomic formulae of level 0 of the types: $x = y$, $x \in X^1$, $\langle x, y \rangle \in X^3$. $\mathcal{T}$ is a *KE-tableau* for Φ if there exists a finite sequence $\mathcal{T}_1, \ldots, \mathcal{T}_t$ such that (i) $\mathcal{T}_1$ is a one-branch tree consisting of the sequence $C_1, \ldots, C_p$, (ii) $\mathcal{T}_t = \mathcal{T}$, and (iii) for each $i < t$, $\mathcal{T}_{i+1}$ is obtained from $\mathcal{T}_i$ either by an application of one of the rules in Fig. 1 or by applying a substitution σ to a branch ϑ of $\mathcal{T}_i$. The set of formulae $\mathcal{S}_i^{\overline{\beta}} = \{\overline{\beta}_1, \ldots, \overline{\beta}_n\} \setminus \{\overline{\beta}_i\}$ occurring as premise in the E-rule contains the complements of all the components of the formula β with the exception of the component β_i. The substitution σ is applied to each formula X of ϑ, the resulting branch is denoted with $\vartheta\sigma$.

Let $\mathcal{T}$ be a KE-tableau. A branch ϑ of $\mathcal{T}$ is *closed* if it contains either both A and $\neg A$, for some formula A, or a literal of type $\neg(x = x)$. Otherwise, the branch is *open.* A KE-tableau is *closed* if all its branches are closed. A formula $\beta_1 \vee \ldots \vee \beta_n$ is *fulfilled* in a branch ϑ, if β_i is in ϑ, for some $i = 1, \ldots, n$. A branch

$$\frac{\beta_1 \vee \ldots \vee \beta_n \qquad \mathcal{S}_i^{\overline{\beta}}}{\beta_i} \ \textbf{E-Rule}$$

where $\mathcal{S}_i^{\overline{\beta}} := \{\overline{\beta}_1, ..., \overline{\beta}_n\} \setminus \{\overline{\beta}_i\}$, for $i = 1, ..., n$

$$\frac{}{A \ \mid \ \overline{A}} \ \textbf{PB-Rule}$$

with A a literal

Fig. 1. Expansion rules for the KE-tableau.

ϑ is *fulfilled* if every formula $\beta_1 \vee \ldots \vee \beta_n$ occurring in ϑ is fulfilled. A branch ϑ is *complete* if either it is closed or it is open, fulfilled, and it does not contain any literal of type $x = y$, where x and y are distinct variables. A KE-tableau is *complete* if all its branches are complete.

A $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$ *satisfies* a branch ϑ of a KE-tableau (ϑ *is satisfied* by $\boldsymbol{\mathcal{M}}$), and we write $\boldsymbol{\mathcal{M}} \models \vartheta$, if $\boldsymbol{\mathcal{M}} \models X$, for every formula X occurring in ϑ. A $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$ satisfies a KE-tableau $\mathcal{T}$ ($\mathcal{T}$ *is satisfied* by $\boldsymbol{\mathcal{M}}$), and we write $\boldsymbol{\mathcal{M}} \models \mathcal{T}$, if $\boldsymbol{\mathcal{M}}$ satisfies a branch ϑ of $\mathcal{T}$. A branch ϑ of a KE-tableau $\mathcal{T}$ is *satisfiable* if there exists a $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$ that satisfies ϑ. A KE-tableau is satisfiable if at least one of its branches is satisfiable.

Let ϑ be a branch of a KE-tableau. We denote with $<_\vartheta$ and arbitrary but fixed total order on variables in $\mathsf{Var}_0(\vartheta)$.

In what follows we introduce the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ taking as input the set $\Phi_{\mathcal{KB}}$ constructed from a $\mathsf{4LQS^R}$-formula $\phi_{\mathcal{KB}}$ representing a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge base $\mathcal{KB}$ as illustrated above, and returning a complete KE-tableau $\mathcal{T}_{\mathcal{KB}}$ for $\Phi_{\mathcal{KB}}$.

Procedure 1 *KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ ($\Phi_{\mathcal{KB}}$)*

1. *$\mathcal{T}_{\mathcal{KB}} := \Phi_{\mathcal{KB}}$;*
2. *Select an open branch ϑ of $\mathcal{T}_{\mathcal{KB}}$ that is not fulfilled.*
   (a) *Select a formula $\beta_1 \vee \ldots \vee \beta_n$ in ϑ that is not fulfilled.*
      i. *If $\mathcal{S}_j^{\overline{\beta}}$ is in ϑ, for some $j \in \{1, \ldots, n\}$, apply the E-Rule to $\beta_1 \vee \ldots \vee \beta_n$ and $\mathcal{S}_j^{\overline{\beta}}$ in ϑ and go to step 2.*
      ii. *If $\mathcal{S}_j^{\overline{\beta}}$ is not in ϑ, for every $j = 1, \ldots, n$, let $B^{\overline{\beta}}$ be the collection of formulae $\overline{\beta}_1, \ldots, \overline{\beta}_n$ present in ϑ and let $\overline{\beta}_h$ be the lowest index formula such that $\overline{\beta}_h \in \{\{\overline{\beta}_1, \ldots, \overline{\beta}_n\} \setminus B^{\overline{\beta}}\}$, then apply the PB-rule to $\overline{\beta}_h$ on ϑ, and go to step 2.*
3. *Select an open branch ϑ of $\mathcal{T}_{\mathcal{KB}}$ containing literals of type $x = y$, with distinct x, y.*
   (a) *$\sigma_\vartheta := \epsilon$;*
   (b) *$\mathsf{Eq}_\vartheta := \{$literals of type $x = y$, occurring in $\vartheta\}$;*
   (c) *while Eq_ϑ contains $x = y$, with distinct x, y do*
      i. *Select $x = y$ in Eq_ϑ, with distinct x, y;*
      ii. *let $z := \min_{<_\vartheta}(x, y)$;*
      iii. *$\sigma_\vartheta := \sigma_\vartheta \cdot \{x/z, y/z\}$;*
      iv. *$\mathsf{Eq}_\vartheta := \mathsf{Eq}_\vartheta \sigma_\vartheta$;*
      end while
   (d) *$\vartheta := \vartheta\sigma_\vartheta$;*
4. *Return $\mathcal{T}_{\mathcal{KB}}$.*

4.1.1 Correctness, termination, and complexity of the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{\mathbf{4,\times}}$

In this section we show that the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ is correct, we briefly outline a proof of its termination, and we provide some complexity results.

Correctness of the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ is proved by showing that $\Phi_{\mathcal{KB}}$ defined above is satisfiable if and only if KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}(\Phi_{\mathcal{KB}})$ returns a not closed KE-tableau $\mathcal{T}_{\mathcal{KB}}$. Specifically, Theorem 2 below proves that if $\Phi_{\mathcal{KB}}$ is satisfiable then $\mathcal{T}_{\mathcal{KB}}$ is not closed, while Theorem 3 shows that if $\mathcal{T}_{\mathcal{KB}}$ is not closed then $\Phi_{\mathcal{KB}}$ is satisfiable. Technical Lemmas 2 and 3 supporting the proof of Theorem 2 are stated and shown in what follows.

Lemma 2. *Let ϑ be a branch of $\mathcal{T}_{\mathcal{KB}}$ selected at step* 3 *of the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}(\Phi_{\mathcal{KB}})$, let σ_ϑ be the associated substitution constructed during the execution of the while loop at step* 3(*c*)*, and let $\boldsymbol{\mathcal{M}} = (D, M)$ be a* $\mathsf{4LQS^R}$*-interpretation satisfying ϑ. Then*

$$Mx = Mx\sigma_\vartheta, \ \textit{for every } x \in \mathsf{Var}_0(\vartheta)\,, \tag{2}$$

is an invariant of the while loop at step 3(*c*)*.*

Proof. We prove the thesis by induction on the number i of iterations of the while loop at step 3(c) of the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ on input $\Phi_{\mathcal{KB}}$. For simplicity we indicate with $\sigma_\vartheta^{(i)}$ and with $Eq_{\sigma_\vartheta}^{(i)}$ the substitution σ_ϑ and the set Eq_{σ_ϑ} calculated at iteration $i \geq 0$, respectively.

If $i = 0$, $\sigma_\vartheta^{(0)}$ is the empty substitution ϵ and thus (2) trivially holds.

Assume by inductive hypothesis that (2) holds at iteration $i \geq 0$. We want to prove that (2) holds at iteration $i + 1$.

At iteration $i+1$, $\sigma_\vartheta^{(i+1)} = \sigma_\vartheta^{(i)} \cdot \{x/z, y/z\}$, where $z = \min_{<_\vartheta}\{x, y\}$ and $x = y$ is a literal in $Eq_{\sigma_\vartheta}^{(i)}$, with distinct x, y. We assume, without loss of generality, that z is the variable x (an analogous proof can be carried out assuming that z is the variable y). By inductive hypothesis $Mw = Mw\sigma_\vartheta^{(i)}$, for every $w \in \mathsf{Var}_0(\vartheta)$. If $w\sigma_\vartheta^{(i)} \in \mathsf{Var}_0(\vartheta) \setminus \{y\}$, plainly $w\sigma_\vartheta^{(i)}$ and $w\sigma_\vartheta^{(i+1)}$ coincide and thus $Mw\sigma_\vartheta^{(i)} = Mw\sigma_\vartheta^{(i+1)}$. Since $Mw = Mw\sigma_\vartheta^{(i)}$, it follows that $Mw = Mw\sigma_\vartheta^{(i+1)}$.

If $w\sigma_\vartheta^{(i)}$ coincides with y we reason as follows. At iteration $i+1$ variables x, y are considered because the literal $x = y$ is selected from $Eq_{\sigma_\vartheta}^{(i)}$. If $x = y$ is a literal belonging to ϑ, then $Mx = My$. Since $w\sigma_\vartheta^{(i)}$ coincides with y, $w\sigma_\vartheta^{(i+1)}$ coincides with x, $My = Mx$, and by inductive hypothesis $Mw = Mw\sigma_\vartheta^{(i)}$, it holds that $Mw = Mw\sigma_\vartheta^{(i+1)}$. If $x = y$ is not a literal occurring in ϑ, then ϑ must contain a literal $x' = y'$ such that, at iteration i, x coincides with $x'\sigma_\vartheta^{(i)}$ and y coincides with $y'\sigma_\vartheta^{(i)}$. Since $Mx' = My'$ and, by inductive hypothesis, $Mx' = Mx'\sigma_\vartheta^{(i)}$, and $My' = My'\sigma_\vartheta^{(i)}$, it holds that $Mx = My$, and thus, reasoning as above, $Mw = Mw\sigma_\vartheta^{(i+1)}$. Since (2) holds at each iteration of the while loop, it is an invariant of the loop as we wished to prove. □

Lemma 3. *Let $\mathcal{T}_0, \ldots, \mathcal{T}_h$ be a sequence of KE-tableaux such that $\mathcal{T}_0 = \Phi_{\mathcal{KB}}$, and $\mathcal{T}_{i+1}$ is obtained from $\mathcal{T}_i$ by applying either step $2(ai)$, or step $2(aii)$, or step $3(d)$ of the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$, for $i = 1, \ldots, h-1$. If $\mathcal{T}_i$ is satisfied by a $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$, then $\mathcal{T}_{i+1}$ is satisfied by $\boldsymbol{\mathcal{M}}$ as well, for $i = 1, \ldots, h-1$.*

Proof. Let $\boldsymbol{\mathcal{M}} = (D, M)$ be a $\mathsf{4LQS^R}$-interpretation satisfying $\mathcal{T}_i$. Then $\boldsymbol{\mathcal{M}}$ satisfies a branch $\bar{\vartheta}$ of $\mathcal{T}_i$. In case the branch $\bar{\vartheta}$ is different from the branch selected at step 2, if the E-rule (step $2(ai)$) or the PB-rule (step $2(aii)$) is applied, or at step 3, if a substitution for handling equalities (step $3(d)$) is applied, $\bar{\vartheta}$ belongs to $\mathcal{T}_{i+1}$ and therefore $\mathcal{T}_{i+1}$ is satisfied by $\boldsymbol{\mathcal{M}}$. In case $\bar{\vartheta}$ is the branch selected and modified to obtain $\mathcal{T}_{i+1}$, we have to consider the following distinct cases.

- $\bar{\vartheta}$ has been selected at step 2 and thus it is an open branch not yet fulfilled. Then, if step $2(ai)$ is executed, the E-rule is applied to a not fulfilled formula $\beta_1 \vee \ldots \vee \beta_n$ and to the set of formulae $\mathcal{S}_j^{\overline{\beta}}$ on the branch $\bar{\vartheta}$ generating the new branch $\bar{\vartheta}' := \bar{\vartheta}; \beta_i$. Plainly, if $\boldsymbol{\mathcal{M}} \models \bar{\vartheta}$, $\boldsymbol{\mathcal{M}} \models \beta_1 \vee \ldots \vee \beta_n$, $\boldsymbol{\mathcal{M}} \models \mathcal{S}_j^{\overline{\beta}}$ and, as a consequence, $\boldsymbol{\mathcal{M}} \models \beta_i$. Thus $\boldsymbol{\mathcal{M}} \models \bar{\vartheta}'$ and finally, $\boldsymbol{\mathcal{M}}$ satisfies $\mathcal{T}_{i+1}$. If step $2(aii)$ is performed, the PB-rule is applied on $\bar{\vartheta}$ originating the branches (belonging to $\mathcal{T}_{i+1}$) $\bar{\vartheta}' := \bar{\vartheta}; \overline{\beta}_h$ and $\bar{\vartheta}'' := \bar{\vartheta}; \beta_h$. Since either $\boldsymbol{\mathcal{M}} \models \beta_h$ or $\boldsymbol{\mathcal{M}} \models \overline{\beta}_h$, it holds that either $\boldsymbol{\mathcal{M}} \models \bar{\vartheta}'$ or $\boldsymbol{\mathcal{M}} \models \bar{\vartheta}''$. Thus $\boldsymbol{\mathcal{M}}$ satisfies $\mathcal{T}_{i+1}$, as we wished to prove.
- $\bar{\vartheta}$ has been selected at step 3 and thus it is an open and fulfilled branch not yet complete. Once step $3(d)$ is executed the new branch $\bar{\vartheta}\sigma_{\bar{\vartheta}}$ is generated. Since $\boldsymbol{\mathcal{M}} \models \bar{\vartheta}$ and, by Lemma 2, $Mx = Mx\sigma_{\bar{\vartheta}}$, for every $x \in \mathsf{Var}_0(\bar{\vartheta})$, it holds that $\boldsymbol{\mathcal{M}} \models \bar{\vartheta}\sigma_{\bar{\vartheta}}$ and that $\boldsymbol{\mathcal{M}}$ satisfies $\mathcal{T}_{i+1}$. Thus the thesis follows. □

We are ready to prove the following theorems.

Theorem 2. *If $\Phi_{\mathcal{KB}}$ is satisfiable, then $\mathcal{T}_{\mathcal{KB}}$ is not closed.*

Proof. Let us assume by contradiction that $\mathcal{T}_{\mathcal{KB}}$ is closed. Since $\Phi_{\mathcal{KB}}$ is satisfiable, there exists a $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$ satisfying every formula of $\Phi_{\mathcal{KB}}$. Thanks to Lemma 3, any KE-tableau for $\Phi_{\mathcal{KB}}$ obtained by applying either step $2(a.i)$, or step $2(a.ii)$, or step $3(d)$ of the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$, is satisfied by $\boldsymbol{\mathcal{M}}$. Thus $\mathcal{T}_{\mathcal{KB}}$ is satisfied by $\boldsymbol{\mathcal{M}}$ as well. In particular, there exists a branch ϑ_c of $\mathcal{T}_{\mathcal{KB}}$ satisfied by $\boldsymbol{\mathcal{M}}$. Since $\mathcal{T}_{\mathcal{KB}}$ is closed, by the absurd hypothesis, the branch ϑ_c is closed as well and thus, by definition, it contains either both A and $\neg A$, for some formula A, or a literal of type $\neg(x = x)$. ϑ is satisfied by $\boldsymbol{\mathcal{M}}$ and thus, either $\boldsymbol{\mathcal{M}} \models A$ and $\boldsymbol{\mathcal{M}} \models \neg A$ or $\boldsymbol{\mathcal{M}} \models \neg(x = x)$. Absurd. Thus, we have to admit that the KE-tableau $\mathcal{T}_{\mathcal{KB}}$ is not closed. □

Theorem 3. *If $\mathcal{T}_{\mathcal{KB}}$ is not closed, then $\Phi_{\mathcal{KB}}$ is satisfiable.*

Proof. Since $\mathcal{T}_{\mathcal{KB}}$ is not closed, there exists a branch ϑ' of $\mathcal{T}_{\mathcal{KB}}$ which is open and complete.

The branch ϑ' is obtained during the execution of the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ on input $\Phi_{\mathcal{KB}}$ from an open fulfilled branch ϑ by applying to ϑ the substitution

σ_ϑ constructed during the execution of step 3 of the procedure. Thus, $\vartheta' = \vartheta\sigma_\vartheta$. Since each formula of $\Phi_{\mathcal{KB}}$ occurs in ϑ, showing that ϑ is satisfiable is enough to prove that $\Phi_{\mathcal{KB}}$ is satisfiable.

Let us construct a $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}_\vartheta = (D_\vartheta, M_\vartheta)$ satisfying every formula X occurring in ϑ and thus $\Phi_{\mathcal{KB}}$.

$\boldsymbol{\mathcal{M}}_\vartheta = (D_\vartheta, M_\vartheta)$ is defined as follows.

- $D_\vartheta \coloneqq \{x\sigma_\vartheta : x \in \mathsf{Var}_0(\vartheta)\}$;
- $M_\vartheta x \coloneqq x\sigma_\vartheta$, $x \in \mathsf{Var}_0(\vartheta)$;
- $M_\vartheta X^1 \coloneqq \{x\sigma_\vartheta : x \in X^1 \text{ occurs in } \vartheta\}$, $X^1 \in \mathsf{Var}_1(\vartheta)$;
- $M_\vartheta X^3 \coloneqq \{\langle x\sigma_\vartheta, y\sigma_\vartheta\rangle : \langle x, y\rangle \in X^3 \text{ occurs in } \vartheta\}$, $X^3 \in \mathsf{Var}_3(\vartheta)$.

In what follows we show that $\boldsymbol{\mathcal{M}}_\vartheta$ satisfies each formula in ϑ. Our proof is carried out by induction on the structure of formulae and cases distinction. Let us consider, at first, a literal $x = y$ occurring in ϑ. By the construction of σ_ϑ described in the procedure KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$, $x\sigma_\vartheta$ and $y\sigma_\vartheta$ have to coincide. Thus $M_\vartheta x = x\sigma_\vartheta = y\sigma_\vartheta = M_\vartheta y$ and then $\boldsymbol{\mathcal{M}}_\vartheta \models x = y$.

Next we consider a literal $\neg(z = w)$ occurring in ϑ. If $z\sigma_\vartheta$ and $w\sigma_\vartheta$ coincide, namely they are the same variable, then the branch $\vartheta' = \vartheta\sigma_\vartheta$ must be a closed branch against our hypothesis. Thus $z\sigma_\vartheta$ and $w\sigma_\vartheta$ are distinct variables and therefore $M_\vartheta z = z\sigma_\vartheta \neq w\sigma_\vartheta = M_\vartheta w$, then $\boldsymbol{\mathcal{M}}_\vartheta \not\models z = w$ and finally $\boldsymbol{\mathcal{M}}_\vartheta \models \neg(z = w)$, as we wished to prove.

Let $x \in X^1$ be a literal occurring in ϑ. By the definition of M_ϑ, $x\sigma_\vartheta \in M_\vartheta X^1$, namely $M_\vartheta x \in M_\vartheta X^1$ and thus $\boldsymbol{\mathcal{M}}_\vartheta \models x \in X^1$ as desired. If $\neg(y \in X^1)$ occurs in ϑ, then $y\sigma_\vartheta \notin M_\vartheta X^1$. Assume, by contradiction that $y\sigma_\vartheta \in M_\vartheta X^1$. Then there is a literal $z \in X^1$ in ϑ such that $z\sigma_\vartheta$ and $y\sigma_\vartheta$ coincide. In this case the branch ϑ', obtained from ϑ applying the substitution σ_ϑ would be closed, contradicting the hypothesis. Thus $y\sigma_\vartheta \notin M_\vartheta X^1$ implies that $M_\vartheta y \notin M_\vartheta X^1$, that $\boldsymbol{\mathcal{M}}_\vartheta \not\models y \in X^1$, and finally that $\boldsymbol{\mathcal{M}}_\vartheta \models \neg(y \in X^1)$.

If $\langle x, y\rangle \in X^3$ is a literal in ϑ, then by definition of M_ϑ, $\langle x\sigma_\vartheta, y\sigma_\vartheta\rangle \in M_\vartheta X^3$, that is $\langle M_\vartheta x, M_\vartheta y\rangle \in M_\vartheta X^3$, and thus $\boldsymbol{\mathcal{M}}_\vartheta \models \langle x, y\rangle \in X^3$.

Let $\neg(\langle z, w\rangle \in X^3)$ be a literal occurring in ϑ. Assume that $\langle z\sigma_\vartheta, w\sigma_\vartheta\rangle \in M_\vartheta X^3$. Then a literal $\langle z', w'\rangle \in X^3$ occurs in ϑ such that $z\sigma_\vartheta$ coincides with $z'\sigma_\vartheta$ and that $w\sigma_\vartheta$ coincides with $w'\sigma_\vartheta$. But then the branch $\vartheta' = \vartheta\sigma_\vartheta$ would be closed contradicting the hypothesis. Thus we have to admit that $\langle z\sigma_\vartheta, w\sigma_\vartheta\rangle \notin M_\vartheta X^3$, that is $\langle M_\vartheta z, M_\vartheta w\rangle \notin M_\vartheta X^3$. Thus $\boldsymbol{\mathcal{M}}_\vartheta \not\models \langle x, y\rangle \in X^3$ and finally $\boldsymbol{\mathcal{M}}_\vartheta \models \neg(\langle x, y\rangle \in X^3)$.

Let $\beta = \beta_1 \vee \ldots \vee \beta_k$ be a disjunction of literals in ϑ. Since ϑ is fulfilled, β is fulfilled too and, therefore, ϑ contains a disjunct β_i, for some $i \in \{1, \ldots, k\}$ of β. By inductive hypothesis $\boldsymbol{\mathcal{M}}_\vartheta \models \beta_i$ and thus $\boldsymbol{\mathcal{M}}_\vartheta \models \beta$.

We have shown that $\boldsymbol{\mathcal{M}}_\vartheta$ satisfies each formula in ϑ and, in particular the formulae in $\Phi_{\mathcal{KB}}$. It turns out that $\Phi_{\mathcal{KB}}$ is satisfiable as we wished to prove. □

It is easy to check that the $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}_\vartheta$ defined in Theorem 3 satisfies $\phi_{\mathcal{KB}}$, a collection of $\mathsf{4LQS^R}$-purely universal formulae and of $\mathsf{4LQS^R}$-quantifier free atomic formulae corresponding to a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-knowledge base $\mathcal{KB}$ and, therefore, that the following corollary holds.

Corollary 1. *If $\mathcal{T}_{\mathcal{KB}}$ is not closed, then $\phi_{\mathcal{KB}}$ is satisfiable.*

Termination of Procedure 1 is based on the fact that step (2) and step (3) terminate. It is easy to check that step (3) can be executed only after step (2) terminates and thus the termination proofs of step (2) and of step (3) are carried out indipendently. Concerning termination of step (2), our proof is based on the following two facts. The E-Rule and PB-Rule are applied only to non-fulfilled formulae on open branches and tend to reduce the number of non-fulfilled formulae occurring on the considered branch. In particular, when the E-Rule is applied on a branch ϑ, the number of non-fulfilled formulae in ϑ decreases. In case of application of the PB-Rule on a formula $\beta = \beta_1 \vee \ldots \vee \beta_n$ on a branch, the rule generates two branches. In one of them the number of non-fulfilled formulae decreases (because β becomes fulfilled). In the other one the number of non-fulfilled formulae stays constant but the subset $B^{\overline{\beta}}$ of $\{\overline{\beta}_1, \ldots, \overline{\beta}_n\}$ occurring on the branch gains a new element. Once $|B^{\overline{\beta}}|$ gets equal to $n-1$, namely after at most $n-1$ applications of the PB-rule, the E-rule is applied and the formula $\beta = \beta_1 \vee \ldots \vee \beta_n$ becomes fulfilled, thus decrementing the number of non-fulfilled formulae on the branch. Since the number of non-fulfilled formulae on each open branch gets equal to zero after a finite number of steps and the E-rule and PB-rule can be applied only to non-fulfilled formulae on open branches, the step (2) terminates. Termination of step (3) is proved considering that (a) the number of branches of the KE-tableau resulting from the execution of step (2) is finite and (b) the while loop of step (3) always terminates in a finite number of steps. We prove (b) reasoning as follows. Each branch ϑ of the KE-tableau resulting from step (2) is finite and thus the number of literals of type $x = y$ is finite as well.

Initially, the set Eq_ϑ contains a finite number of literals of the type $x = y$, and σ_ϑ is the empty substitution. The while-loop of step (3) terminates in a finite number of steps, since the number of literals of type $x = y$ with distinct x and y in Eq_ϑ can only decrease. At each iteration of the while-loop, the procedure constructs $\sigma_\vartheta := \sigma_\vartheta \cdot \{x/z, y/z\}$ choosing z among x and y according to a fixed total order over the variables of $\mathsf{Var}_0(\vartheta)$. Since the application of σ_ϑ to Eq_ϑ at each iteration replaces a literal of the type $x = y$ with distinct x and y, with a literal of the type $x = x$, the number of literals of type $x = y$ with distinct x, y of the set Eq_ϑ decreases. Thus step (3) terminates in a finite number of steps. Since step (2) and (3) terminate in a finite number of steps, thus Procedure 1 terminates, as we wished to prove.

Next, we provide some complexity results. Let r be the maximum number of universal quantifiers in S_i, and $k := |\mathtt{Var}_0(\phi_{\mathcal{KB}})|$. Then, each S_i generates k^r expansions. Since the knowledge base contains m such formulae, the number of disjunctions in the initial branch of the KE-tableau is $m \cdot k^r$. Next, let ℓ be the maximum number of literals in S_i, for $i = 1, \ldots, m$. Then, the maximum depth of the KE-tableau, namely the maximum size of the models of $\Phi_{\mathcal{KB}}$ constructed as illustrated above, is $\mathcal{O}(\ell m k^r)$ and the number of leaves of the tableau, that is the number of such models of $\Phi_{\mathcal{KB}}$, is $O(2^{\ell m k^r})$. Notice that the construction of Eq_ϑ and of σ_ϑ at step 3 of Procedure 1 is $\mathcal{O}(\ell m k^r)$, for each branch ϑ.

4.2 Computing the answer set

We now introduce a procedure that, given a KE-tableau constructed by Procedure 1 and a $\mathsf{4LQS^R}$-formula ψ_Q representing a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-HO conjunctive query Q, yields all the substitutions σ' of the HO answer set Σ' of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$, namely all the substitutions σ' such that $\boldsymbol{\mathcal{M}} \models \phi_{\mathcal{KB}} \wedge \psi_Q\sigma'$, for some $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$. By correctness of Procedure 1, we can limit ourselves to consider only the models $\boldsymbol{\mathcal{M}}_\vartheta$ of $\phi_{\mathcal{KB}}$ defined in the proof of Theorem 3 for open and fulfilled branches ϑ determined by the execution of step 2.

For every open and complete branch $\vartheta' = \vartheta\sigma_\vartheta$ of $\mathcal{T}_{\mathcal{KB}}$, we construct a decision tree $\mathcal{D}_{\vartheta'}$ such that every maximal branch of $\mathcal{D}_{\vartheta'}$ induces a substitution σ' such that $\boldsymbol{\mathcal{M}}_\vartheta \models \psi_Q\sigma_\vartheta\sigma'$. $\mathcal{D}_{\vartheta'}$ is defined in what follows.

Let d be the number of literals in ψ_Q. $\mathcal{D}'_\vartheta$ is a finite labelled tree of depth $d+1$ whose labelling satisfies the following conditions, for $i = 0, \ldots, d$:

(i) every node of $\mathcal{D}_{\vartheta'}$ at level i is labelled with $(\sigma'_i, \psi_Q\sigma_\vartheta\sigma'_i)$, and, in particular, the root is labelled with $(\sigma'_0, \psi_Q\sigma_\vartheta\sigma'_0)$, where σ'_0 is the empty substitution;

(ii) if a node at level i is labelled with $(\sigma'_i, \psi_Q\sigma_\vartheta\sigma'_i)$, then its s-successors, with $s > 0$, are labelled with $\big(\sigma'_i\varrho_1^{q_{i+1}}, \psi_Q\sigma_\vartheta(\sigma'_i\varrho_1^{q_{i+1}})\big), \ldots, \big(\sigma'_i\varrho_s^{q_{i+1}}, \psi_Q\sigma_\vartheta(\sigma'_i\varrho_s^{q_{i+1}})\big)$, where q_{i+1} is the $(i+1)$-st conjunct of $\psi_Q\sigma_\vartheta\sigma'_i$ and $\mathcal{S}_{q_{i+1}} = \{\varrho_1^{q_{i+1}}, \ldots, \varrho_s^{q_{i+1}}\}$ is the collection of the substitutions $\varrho = \{v_1/o_1, \ldots, v_n/o_n, c_1/C_1, \ldots, c_m/C_m, r_1/R_1, \ldots, r_k/R_k, p_1/P_1, \ldots, p_h/P_h\}$, with $\{v_1, \ldots, v_n\} = \mathtt{Var}_0(q_{i+1})$, $\{c_1, \ldots, c_m\} = \mathtt{Var}_1(q_{i+1})$, $\{p_1, \ldots, p_h, r_1, \ldots, r_k\} = \mathtt{Var}_3(q_{i+1})$, such that $t = q_{i+1}\varrho$, for some literal t in ϑ'. If $s = 0$, the node labelled with $(\sigma'_i, \psi_Q\sigma_\vartheta\sigma'_i)$ is a leaf node and, if $i = d$, $\sigma_\vartheta\sigma'_i$ is added to Σ'.

The answer set of ψ_Q with respect to $\phi_{\mathcal{KB}}$ is computed by Procedure 2.

Procedure 2 *HOCQA-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ ($\mathcal{T}_{\mathcal{KB}}$,$\psi_Q$)*

1. $\Sigma' := \emptyset$
2. *foreach open branch* ϑ' *in* $\mathcal{T}_{\mathcal{KB}}$ *do*
   (a) $\mathcal{S} := \emptyset$
   (b) *Let* σ_ϑ *be such that* $\vartheta' = \vartheta\sigma_\vartheta$
   (c) *push* $(\epsilon, \psi_Q\sigma_\vartheta)$ *in* $\mathcal{S}$
   (d) *while* $\mathcal{S}$ *is not empty do*
      i. $(\sigma', \psi_Q\sigma_\vartheta\sigma') :=$ *pop from* $\mathcal{S}$
      ii. *if* $\psi_Q\sigma_\vartheta\sigma' \neq \lambda$ *do*
         A. *Let* q *be the leftmost conjunct of* $\psi_Q\sigma_\vartheta\sigma'$
         B. $\psi_Q\sigma_\vartheta\sigma' := \psi_Q\sigma_\vartheta\sigma'$ *deprived of* q
         C. $Lit_Q^M := \{t \in \vartheta' : t = q\rho$, *for some substitution* $\rho\}$
         D. *while* Lit_Q^M *is not empty do*
            – *Let* $t \in Lit_Q^M$, $t = q\rho$
            – $Lit_Q^M := Lit_Q^M \setminus \{t\}$
            – *push* $(\sigma'\rho, \psi_Q\sigma_\vartheta\sigma'\rho)$ *in* $\mathcal{S}$
         E. *endwhile*
      iii. *else*

A. $\Sigma' = \Sigma' \cup \{\sigma_\vartheta \sigma'\}$

iv. *endif*

(e) *endwhile*

3. *endfor*
4. *return* Σ'

For each open branch ϑ' of $\mathcal{T}_{\mathcal{KB}}$, Procedure 2 computes the corresponding $\mathcal{D}'_\vartheta$ as follows. The procedure constructs a stack of the nodes of $\mathcal{D}'_\vartheta$. Initially the stack contains the root node $(\epsilon, \psi_Q \sigma_\vartheta)$ of $\mathcal{D}'_\vartheta$ as defined in condition (i). Then the procedure computes iteratively the following steps. It pops an element of the stack. If the last literal of the query ψ_Q has not been reached, the s-successors of the current node are computed as in condition (ii) and inserted in the stack. Otherwise the current node has the form (σ', λ) and the substitution $\sigma_\vartheta \sigma'$ is inserted in Σ'.

4.2.1 Correctness, termination, and complexity of Procedure 2

We show that Procedure 2 is correct by proving that the set Σ' coincides with the HO answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$.

It is convenient now to introduce the notions of ground and non ground literal, defined according to the mapping θ constructed in the proof of Theorem 1. A literal is said to be *ground* if it contains variables of sort 0 of the form x_o, where $o \in \mathbf{Ind} \cup \bigcup\{N_C(d) : d \in N_{\mathbf{D}}\}$, variables of sort 1 of the form X^1_C, where $C \in \mathbf{C}$, and variables of sort 3 of the form X^3_R, where $R \in \mathbf{R_A} \cup \mathbf{R_D}$. A literal is said to be *non ground* if it contains variable of sort 0 of the form x_v, where $v \in \mathsf{V_i}$, variables of sort 1 of the form X^1_c, where $c \in \mathsf{V_c}$, and variables of sort 3, of the form X^3_r, where $r \in \mathsf{V_{ar}} \cup \mathsf{V_{cr}}$. Notice that branches of $\mathcal{T}_{\mathcal{KB}}$ only contains ground literals while ψ_Q may also contain non ground literals. Thus, the substitutions constructed from σ' are of the type $\{x_1/y_1, \ldots, x_m/y_m\}$, where x_i is non ground and y_i is ground, for $i = 1, \ldots, m$.

We are ready to prove the following lemma.

Lemma 4. *Let $\psi_Q = q_1 \wedge \ldots \wedge q_d$ be an HO $\mathsf{4LQS^R}$-conjunctive query, let $\vartheta' = \vartheta\sigma_\vartheta$ be an open and complete branch of the KE-tableau $\mathcal{T}_{\mathcal{KB}}$, and let Σ' be the output of HOCQA-$\mathcal{DL}^{4,\times}_{\mathbf{D}}$($\mathcal{T}_{\mathcal{KB}}$,$\psi_Q$). Then, given a substitution σ,*

$$\sigma \in \Sigma' \iff \{q_1\sigma, \ldots, q_d\sigma\} \subseteq \vartheta' .$$

Proof. If $\sigma' \in \Sigma'$, then $\sigma' = \sigma_\vartheta \sigma'_1$ and the decision tree $\mathcal{D}_{\vartheta'}$ contains a branch η of length $d+1$ having as leaf (σ'_1, λ). Specifically, the branch η is constituted by the nodes

$(\epsilon, q_1\sigma_\vartheta \wedge \ldots \wedge q_d\sigma_\vartheta)$, $(\rho^{(1)}, q_2\sigma_\vartheta\rho^{(1)} \wedge \ldots \wedge q_d\sigma_\vartheta\rho^{(1)})$, ..., $(\rho^{(1)} \ldots \rho^{(d)}, \lambda)$,
and hence $\sigma' = \sigma_\vartheta\rho^{(1)} \ldots \rho^{(d)}$.
Consider the node

$(\rho^{(1)} \ldots \rho^{(i+1)}, q_{i+2}\sigma_\vartheta\rho^{(1)} \ldots \rho^{(i+1)} \wedge \ldots \wedge q_d\sigma_\vartheta\rho^{(1)} \ldots \rho^{(i+1)})$
constructed from the father node

$(\rho^{(1)} \ldots \rho^{(i+1)}, q_{i+1}\sigma_\vartheta\rho^{(1)} \ldots \rho^{(i)} \wedge \ldots \wedge q_d\sigma_\vartheta\rho^{(1)} \ldots \rho^{(i)})$

putting $q_{i+1}\sigma_\vartheta\rho^{(1)}\ldots\rho^{(i+1)} = t$, for some $t \in \vartheta'$. Since $q_{i+1}\sigma_\vartheta\rho^{(1)}\ldots\rho^{(i+1)}$ is a ground literal, $q_{i+1}\sigma_\vartheta\rho^{(1)}\ldots\rho^{(i+1)}$ coincides with $q_{i+1}\sigma'$, then $q_{i+1}\sigma' = t$, and hence $q_{i+1}\sigma' \in \vartheta'$. Given the generality of $i = 0,\ldots,d-1$, $\{q_1\sigma',\ldots,q_d\sigma'\} \subseteq \vartheta'$ as we wished to prove.

We now prove the second part of the lemma. We show that the decision tree $\mathcal{D}_{\vartheta'}$ constructed by Procedure 2 has a branch η of length $d+1$ having as leaf the node (σ_1', λ), with $\sigma_\vartheta\sigma_1' = \sigma' \in \Sigma'$. Since by hypothesis $\vartheta' = \vartheta\sigma_\vartheta$, the root of the decision tree $\mathcal{D}_{\vartheta'}$ is the node $(\epsilon, q_1\sigma_\vartheta \wedge \ldots \wedge q_d\sigma_\vartheta)$. At step i, the procedure selects a literal $q^(i)$, namely $q_i\sigma_\vartheta\rho^{(1)}\ldots\rho^{(i-1)}$, and finds a substitution $\rho^{(i)}$ such that $q_i\sigma_\vartheta\rho^{(1)}\ldots\rho^{(i)}$ coincides with $q_i\sigma'$. Then, the procedure constructs the node

$(\rho^{(1)}\ldots\rho^{(i)}, q_{i+1}\sigma_\vartheta\rho^{(1)}\ldots\rho^{(i)} \wedge \ldots \wedge q_d\sigma_\vartheta\rho^{(1)}\ldots\rho^{(i)})$

At step $d-1$, the procedure constructs the leaf node $(\rho^{(1)}\ldots\rho^{(d)}, \lambda)$, that is (σ_1', λ), as we wished to prove. □

We are ready to prove Theorem 4, stating correctness of Procedure 2.

Theorem 4. *Let Σ' be the set of substitutions returned by Procedure 2 on input ψ_Q. Then Σ' is the HO answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$.*

Proof. To prove the theorem we show that the following two assertions hold.

1. If $\sigma' \in \Sigma'$, then σ' is an element of the HO answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$.
2. If σ' is a substitution of the HO answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$, then $\sigma' \in \Sigma'$.

We prove assertion (1) as follows. Let $\sigma' \in \Sigma'$ and $\vartheta' = \vartheta\sigma_\vartheta$ an open and complete branch of $\mathcal{T}_{\mathcal{KB}}$ such that $\mathcal{D}_{\vartheta'}$ contains a branch η of $d+1$ nodes whose leaf is labelled $\langle\sigma_1', \lambda\rangle$, where σ_1' is a substitution such that $\sigma' = \sigma_\vartheta\sigma_1'$. By Lemma 4, $\{q_1\sigma',\ldots,q_d\sigma'\} \subseteq \vartheta'$. Let $\boldsymbol{\mathcal{M}}_\vartheta$ be a $\mathsf{4LQS^R}$-interpretation constructed as shown in Theorem 3. We have that $\boldsymbol{\mathcal{M}}_\vartheta \models q_i\sigma'$, for $i = 1,\ldots,d$ because $\{q_1\sigma',\ldots,q_d\sigma'\} \subseteq \vartheta'$ holds. Thus $\boldsymbol{\mathcal{M}}_\vartheta \models \psi_Q\sigma'$, and since $\boldsymbol{\mathcal{M}}_\vartheta \models \phi_{\mathcal{KB}}$, $\boldsymbol{\mathcal{M}}_\vartheta \models \phi_{\mathcal{KB}} \wedge \psi_Q\sigma'$ holds. Hence σ' is a substitution of the answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$. To show that assertion (2) holds, let us consider a substitution σ' belonging to the answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$. Then there exists a $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}} \models \phi_{\mathcal{KB}} \wedge \psi_Q\sigma'$. Assume by contradiction that $\sigma' \notin \Sigma'$. Then, by Lemma 4 $\{q_1\sigma,\ldots,q_d\sigma'\} \not\subseteq \vartheta'$, for every open and complete branch ϑ' of $\mathcal{T}_{\mathcal{KB}}$. In particular, given any open complete branch ϑ' of $\mathcal{T}_{\mathcal{KB}}$, there is an $i \in \{1,\ldots,d\}$ such that $q_i\sigma' \notin \vartheta' = \vartheta\sigma_\vartheta$ and thus $\boldsymbol{\mathcal{M}}_\vartheta \not\models q_i\sigma'$.

By the generality of $\vartheta' = \vartheta\sigma_\vartheta$, it holds that every $\boldsymbol{\mathcal{M}}_\vartheta$ satisfying $\mathcal{T}_{\mathcal{KB}}$, and thus $\phi_{\mathcal{KB}}$, does not satisfy $\psi_Q\sigma'$. We recall that by correctness of Procedure 1, we can prove that $\boldsymbol{\mathcal{M}} \models \phi_{\mathcal{KB}} \wedge \psi_Q\sigma'$, for some $\mathsf{4LQS^R}$-interpretation $\boldsymbol{\mathcal{M}}$, by restricting our interest to the interpretations $\boldsymbol{\mathcal{M}}_\vartheta$ of $\phi_{\mathcal{KB}}$ defined in the proof of Theorem 3. It turns out that σ' is not a substitution belonging to the answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$, and this leads to a contradiction. Thus we have to admit that assertion (2) holds. Finally, since assertions (1) and (2) hold, Σ' and the answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$ coincide and the thesis holds. □

Termination of Procedure 2 is based on the following facts. The KE-tableau $\mathcal{T}_{\mathcal{KB}}$ generated by Procedure 1 consists of a finite number of open and complete

branches and thus the "foreach" loop at step 2 is executed a finite number of times. Each iteration of the "foreach" loop at step 2 terminates when the stack S of the nodes of the decision tree gets empty. Since the query ψ_Q contains a finite number of conjuncts and the number of literals on each open and complete branch of $\mathcal{T}_{\mathcal{KB}}$ is finite, the number of possible matches (the set Lit_Q^M) computed at step (C) is finite as well. Once the procedure has processed the last conjunct of the query, the set of possible matches, Lit_Q^M, is empty and thus no element gets pushed in the stack S anymore. Since the first instruction of the while-loop at step (i) removes an element from S, the stack gets empty after a finite number of "pops". Hence Procedure 2 terminates as we wished to prove.
We now provide some complexity results. Let $\delta(\mathcal{T}_{\mathcal{KB}})$ and $\lambda(\mathcal{T}_{\mathcal{KB}})$ be, respectively, the maximum depth of $\mathcal{T}_{\mathcal{KB}}$ and the number of leaves of $\mathcal{T}_{\mathcal{KB}}$ computed above. Let $k := \left|\bigcup_{i=0}^{3} \mathtt{Var}_i(\psi)\right|$. Plainly, $\delta(\mathcal{T}_{\mathcal{KB}}) = \mathcal{O}(\ell m k^r)$ and $\lambda(\mathcal{T}_{\mathcal{KB}}) = \mathcal{O}(2^{\ell m k^r})$. It is easy to verify that $s = 2^k$ is the maximum branching of $\mathcal{D}_\vartheta$. Since $\mathcal{D}_\vartheta$ is a s-ary tree of depth $d+1$, where d is the number of literals in ψ_Q, and the s-successors of a node are computed in $\mathcal{O}(\delta(\mathcal{T}_{\mathcal{KB}}))$ time, the number of leaves in $\mathcal{D}_\vartheta$ is $\mathcal{O}(s^{(d+1)}) = \mathcal{O}(2^{k(d+1)})$ and they are computed in $\mathcal{O}(2^{k(d+1)}\delta(\mathcal{T}_{\mathcal{KB}}))$ time. Finally, since we have $\lambda(\mathcal{T}_{\mathcal{KB}})$ of such decision trees, the answer set of ψ_Q w.r.t. $\phi_{\mathcal{KB}}$ is computed in time $\mathcal{O}(2^{k(d+1)}\delta(\mathcal{T}_{\mathcal{KB}})\lambda(\mathcal{T}_{\mathcal{KB}})) = \mathcal{O}(2^{k(d+1)} \cdot \ell m k^r \cdot 2^{\ell m k^r}) = \mathcal{O}(\ell m k^r 2^{k(d+1)+\ell m k^r})$. Since the size of $\phi_{\mathcal{KB}}$ and of ψ_Q are related to those of $\mathcal{KB}$ and of Q, respectively (see the proof of Theorem 1 for details on the reduction), the construction of the HO answer set of Q with respect to $\mathcal{KB}$ can be done in double-exponential time (see [9] for details). In case $\mathcal{KB}$ contains no role chain axioms and qualified cardinality restrictions, the complexity of our HOCQA problem is in EXPTIME, since the maximum number of universal quantifiers in $\phi_{\mathcal{KB}}$, namely r, is a constant (in particular $r = 3$). We remark that such result can be compared to the complexity of the CQA problem for a wide collection of description logics such as $\mathcal{SHIQ}$ [22]. In particular, the CQA problem for the very expressive description logic $\mathcal{SROIQ}$ turns out to be 2-NEXPTIME-complete.

5 Conclusions and future work

In this contribution we have considered an extension of the problem of the Conjunctive Query Answering (CQA) for the description logic $\mathcal{DL}\langle\mathsf{4LQS}^{\mathsf{R},\times}\rangle(\mathbf{D})$ ($\mathcal{DL}_{\mathbf{D}}^{4,\times}$, for short) to more general queries on roles and concepts. The resulting problem, called Higher Order Conjunctive Query Answering (HOCQA), can be instantiated to the most widespread ABox reasoning services such as instance retrieval, role filler retrieval, and instance checking. We have proved decidability of the problem of HOCQA by resorting to the satisfiability problem for the set-theoretic fragment $\mathsf{4LQS}^{\mathsf{R}}$.

We have introduced a procedure to compute the HO answer set of a $\mathsf{4LQS}^{\mathsf{R}}$-formula ψ_Q representing a HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive query Q w.r.t. a $\mathsf{4LQS}^{\mathsf{R}}$-formula $\phi_{\mathcal{KB}}$ representing a $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ knowledge base consisting of two subprocedures. The first one, called KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$, is based on the KE-tableau system and yields a KE-tableau $\mathcal{T}_{\mathcal{KB}}$ representing the saturation of $\phi_{\mathcal{KB}}$. The second one, called HOCQA-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ and based on decision trees, takes as input $\mathcal{T}_{\mathcal{KB}}$ and ψ_Q and returns the requested HO answer set. Subprocedures KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ and HOCQA-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ are proved correct and complete, and some complexity results are provided.

Such procedure extends the one introduced in [9] because it allows one to treat literals of type $x = y$ and $\neg(x = y)$ and it is able to handle HO $\mathcal{DL}_{\mathbf{D}}^{4,\times}$-conjunctive queries.

We are currently working at the implementation of the procedures KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ and HOCQA-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$. We plan to increase efficiency of the expansion rule of KE-$\mathcal{DL}_{\mathbf{D}}^{4,\times}$ and to generalize our procedure with a data-type checker in order to extend reasoning with data-types. Lastly, we intend to provide a parallel model of the procedures that we are implementing.

We also plan to increase the expressive power of the set theoretic fragments we are working with. In particular, we intend to define a decidable n-level stratified syllogistic allowing to represent an extension of $\mathcal{DL}_{\mathbf{D}}^{4,\times}$ admitting data-type groups. We also aim to extend the set theoretic fragments presented in [7] and in [8] with the construct of generalized union and with a restricted form of binary relational composition operator. The latter operator, in particular, turns out to be useful for the set theoretic representation of various logics. The KE-tableau based procedure will be adapted to the new set theoretic fragments also resorting to techniques introduced in [6] and in [5] in the ambit of relational dual tableaux. On the other hand we deem that KE-tableaux can be used in the ambit of relational dual tableaux to improve the performances of relational dual tableau-based decision procedures.

References

1. D. Calvanese, T. Eiter, and M. Ortiz. Answering regular path queries in expressive description logics: An automata-theoretic approach. In *AAAI*, volume 7, pages 391–396, 2007.
2. D. Calvanese, G. De Giacomo, D. Lembo, M. Lenzerini, and R. Rosati. Data complexity of query answering in description logics. *Artificial Intelligence*, 195:335 – 360, 2013.
3. D. Calvanese, G. De Giacomo, and M. Lenzerini. On the decidability of query containment under constraints. In *Proc. of the* 17^{th} *ACM SIGACT-SIGMOD-SIGART Symp. on Princ. of Database Systems*, PODS '98, pages 149–158, New York, NY, USA, 1998. ACM.
4. D. Cantone, C. Longo, M. Nicolosi-Asmundo, and D. F. Santamaria. Web ontology representation and reasoning via fragments of set theory. In *Web Reasoning and Rule Systems - 9th Int. Conf., RR 2015, Berlin, Germany, August 4-5, 2015, Proc.*, pages 61–76, 2015.
5. D. Cantone, M.Nicolosi-Asmundo, and E. Orlowska. Dual tableau-based decision procedures for some relational logics. In *Proceedings of the 25th Italian Conference on Computational Logic, CEUR-WS Vol. 598, Rende, Italy, July 7-9, 2010*, 2010.

6. D. Cantone, M.Nicolosi-Asmundo, and E. Orlowska. Dual tableau-based decision procedures for relational logics with restricted composition operator. *Journal of Applied Non-Classical Logics*, 21(2):177–200, 2011.
7. D. Cantone and M. Nicolosi-Asmundo. On the satisfiability problem for a 4-level quantified syllogistic and some applications to modal logic. *Fundamenta Informaticae*, 124(4):427–448, 2013.
8. D. Cantone and M. Nicolosi-Asmundo. The decision problem for a three-sorted fragment of set theory with restricted quantification and finite enumerations. *Electr. Notes Theor. Comput. Sci.*, 322:69–86, 2016.
9. D. Cantone, M. Nicolosi-Asmundo, and D. F. Santamaria. Conjunctive Query Answering via a Fragment of Set Theory. In *Proc. of the 17th Int.Conf. on Theoretical Computer Science, ICTCS 2016, CEUR-WS Vol. 1720, pp. 23–35*, Lecce, September 7-9 2016.
10. D. Cantone, M. Nicolosi-Asmundo, and D. F. Santamaria. Conjunctive Query Answering via a Fragment of Set Theory (Ext. Ver). *CoRR*, abs/1606.07337, 2016.
11. D. Cantone, M. Nicolosi-Asmundo, D. F. Santamaria, and F. Trapani. Ontoceramic: an owl ontology for ceramics classification. In *Proc. of the 30th It. Conf. on Computational Logic, CILC 2015*, volume 1459, pages 122–127, Genova, July 1-3 2015. CEUR Electronic Workshop Proceedings.
12. M. D'Agostino. Tableau methods for classical propositional logic. In Marcello D'Agostino, Dov M. Gabbay, Reiner Hähnle, and Joachim Posegga, editors, *Handbook of Tableau Methods*, pages 45–123. Springer, 1999.
13. B. Glimm, I. Horrocks, and U. Sattler. Conjunctive query entailment for shoq. In *Proc. of the 2007 Description Logic Workshop*, volume 250, pages 1–11, 2007.
14. B. Glimm, C. Lutz, I. Horrocks, and U. Sattler. Answering conjunctive queries in the $\mathcal{SHIQ}$ description logic. *Journal of Artif. Intell. Research*, 31:150–197, 2008.
15. I. Horrocks, U. Sattler, and S. Tobies. Reasoning with individuals for the description logic shiq. In David MacAllester, editor, *Proceedings of the 17th International Conference on Automated Deduction (CADE-17)*, number 1831 in Lecture Notes in Computer Science, pages 482–496, Germany, 2000. Springer Verlag.
16. I. Horrocks and S. Tessaris. A conjunctive query language for description logic aboxes. In *In Proc. of the 17th Nat. Conf. on Artificial Intelligence (AAAI-2000)*, pages 399–404, 2000.
17. U. Hustadt, B. Motik, and U. Sattler. Data complexity of reasoning in very expressive description logics. In *IJCAI-05, Proc. of the* 19^{th} *Inter. Joint Conf. on Art. Intell., Edinburgh, Scotland, UK, July 30-August 5, 2005*, pages 466–471, 2005.
18. C. Lutz. The complexity of conjunctive query answering in expressive description logics. In *Automated Reasoning, 4th International Joint Conference, IJCAR 2008, Sydney, Australia, August 12-15, 2008, Proceedings*, pages 179–193, 2008.
19. B. Motik and I. Horrocks. OWL datatypes: Design and implementation. In *Proc. of the 7th Int. Semantic Web Conference (ISWC 2008)*, volume 5318 of *LNCS*, pages 307–322. Springer, October 26–30 2008.
20. B. Motik, U. Sattler, and R. Studer. Query answering for owl-dl with rules. *Web Semantics: Science, Services and Agents on the World Wide Web*, 3(1):41–60, 2005.
21. M. Ortiz, D. Calvanese, and T. Eiter. Characterizing data complexity for conjunctive query answering in expressive description logics. In *Proc. of the 21st Nat. Conf. on Artificial Intelligence (AAAI 2006)*, pages 1–6, 2006.
22. M. Ortiz, R. Sebastian, and M. Šimkus. Query answering in the horn fragments of the description logics shoiq and sroiq. In *Proc. of the 22th Int. Joint Conf. on Artificial Intell. - Vol.Two*, IJCAI'11, pages 1039–1044. AAAI Press, 2011.

23. R. Rosati. On conjunctive query answering in EL. In *Proc. of the 2007 International Workshop on Description Logic (DL 2007)*, pages 1–8. CEUR Electronic Workshop Proceedings, 2007.
24. R.M. Smullyan. *First-order Logic*. Dover books on advanced Math. Dover, 1995.
25. SWRL:. `http://www.w3.org/Submission/SWRL/`.
26. World Wide Web Consortium. OWL 2 Web ontology language structural specification and functional-style syntax (second edition).