The very expressive n-ary description logic $\mathcal{DLR}^{\pm}$

Alessandro Artale, Enrico Franconi, Rafael Peñaloza, Francesco Sportelli

KRDB Research Centre, Free University of Bozen-Bolzano, Italy
{artale,franconi,penaloza,sportelli}@inf.unibz.it

Abstract. We introduce an extension of the n-ary description logic $\mathcal{DLR}$ to deal with attribute-labelled tuples (generalising the positional notation), with arbitrary projections of relations (inclusion dependencies), generic functional dependencies and with global and local objectification (reifying relations or their projections). We show how a simple syntactic condition on the appearance of projections and functional dependencies in a knowledge base makes the language decidable without increasing the computational complexity of the basic $\mathcal{DLR}$ language.

1 Introduction

We introduce in this paper the language $\mathcal{DLR}^{+}$ which extends the n-ary description logics $\mathcal{DLR}$ [Calvanese *et al.*, 1998; Baader *et al.*, 2003] and $\mathcal{DLR}_{\mathit{ifd}}$ [Calvanese *et al.*, 2001] as follows:

– the semantics is based on attribute-labelled tuples: an element of a tuple is identified by an attribute and not by its position in the tuple, e.g., the relation `Person` has attributes `firstname`, `lastname`, `age`, `height` with instance: $\langle$ `firstname: Enrico, lastname: Franconi, age: 53, height: 1.90`$\rangle$;
– renaming of attributes is possible, e.g., to recover the positional semantics: `firstname,lastname,age,height` $\rightleftarrows$ `1,2,3,4`;
– it can express projections of relations, and therefore inclusion dependencies, e.g., $\exists[\texttt{firstname,lastname}]\texttt{Student} \sqsubseteq \exists[\texttt{firstname,lastname}]\texttt{Person}$;
– it can express multiple-attribute cardinalities, and therefore functional dependencies and multiple-attribute keys, e.g., the functional dependency from `firstname, lastname` to `age` in `Person` can be written as:
$\exists[\texttt{firstname,lastname}]\texttt{Person} \sqsubseteq$
$\quad\exists^{\leqslant 1}[\texttt{firstname,lastname}](\exists[\texttt{firstname,lastname,age}]\texttt{Person})$;
– it can express global and local objectification (also known as reification): a tuple may be identified by a unique global identifier, or by an identifier which is unique only within the interpretation of a relation, e.g., to identify the name of a person we can write $\texttt{Name} \sqsubseteq \odot\,\exists[\texttt{firstname,lastname}]\texttt{Person}$.

We show how a simple syntactic condition on the appearance of projections in the knowledge base makes the language decidable without increasing the computational complexity of the basic $\mathcal{DLR}$ language. We call $\mathcal{DLR}^{\pm}$ this fragment

$$
\begin{array}{lcl}
C & \to & \top \mid \bot \mid CN \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \exists^{\lessgtr q}[U_i]R \mid \circledcirc R \mid \odot RN \\
R & \to & RN \mid R_1 \backslash R_2 \mid R_1 \sqcap R_2 \mid R_1 \sqcup R_2 \mid \sigma_{U_i:C} R \mid \exists^{\lessgtr q}[U_1,\ldots,U_k]R \\
\varphi & \to & C_1 \sqsubseteq C_2 \mid R_1 \sqsubseteq R_2 \mid CN(O) \mid RN(U_1:O_1,\ldots,U_n:O_n) \mid O_1 = O_2 \mid O_1 \neq O_2 \\
\vartheta & \to & U_1 \rightleftarrows U_2
\end{array}
$$

Fig. 1. Syntax of $\mathcal{DLR}^+$.

$$
\begin{array}{rll}
\tau(R_1 \backslash R_2) = & \tau(R_1) & \text{if } \tau(R_1) = \tau(R_2) \\
\tau(R_1 \sqcap R_2) = & \tau(R_1) & \text{if } \tau(R_1) = \tau(R_2) \\
\tau(R_1 \sqcup R_2) = & \tau(R_1) & \text{if } \tau(R_1) = \tau(R_2) \\
\tau(\sigma_{U_i:C} R) = & \tau(R) & \text{if } U_i \in \tau(R) \\
\tau(\exists^{\lessgtr q}[U_1,\ldots,U_k]R) = & \{U_1,\ldots,U_k\} & \text{if } \{U_1,\ldots,U_k\} \subset \tau(R) \\
\tau(R) = & \varnothing & \text{otherwise}
\end{array}
$$

Fig. 2. The signature of $\mathcal{DLR}^+$ relations.

of $\mathcal{DLR}^+$. $\mathcal{DLR}^\pm$ is able to correctly express the UML fragment as introduced in [Berardi *et al.*, 2005; Artale *et al.*, 2007] and the ORM fragment as introduced in [Franconi and Mosca, 2013].

2 The Description Logic $\mathcal{DLR}^+$

We first define the syntax of the language $\mathcal{DLR}^+$. A $\mathcal{DLR}^+$ *signature* is a tuple $\mathcal{L} = (\mathcal{C}, \mathcal{R}, \mathcal{O}, \mathcal{U}, \tau)$ where $\mathcal{C}$, $\mathcal{R}$, $\mathcal{O}$ and $\mathcal{U}$ are finite, mutually disjoint sets of *concept names*, *relation names*, *individual names*, and *attributes*, respectively, and τ is a *relation signature* function, associating a set of attributes to each relation name $\tau(RN) = \{U_1,\ldots,U_n\} \subseteq \mathcal{U}$ with $n \geqslant 2$. The *arity* of a relation R is the number of the attributes in its signature; i.e., $\textsc{arity}(R) = |\tau(R)|$. The syntax of concepts C, relations R, formulas φ, and attribute renaming axioms ϑ is given in Figure 1, where $CN \in \mathcal{C}$, $RN \in \mathcal{R}$, $U \in \mathcal{U}$, q is a positive integer and $2 \leqslant k < \textsc{arity}(R)$. We extend the signature function τ to arbitrary relations as specified in Figure 2.

A $\mathcal{DLR}^+$ *TBox* $\mathcal{T}$ is a finite set of *concept inclusion* axioms of the form $C_1 \sqsubseteq C_2$ and *relation inclusion* axioms of the form $R_1 \sqsubseteq R_2$. We will often use $X_1 \equiv X_2$ as a shortcut for the two axioms $X_1 \sqsubseteq X_2$ and $X_2 \sqsubseteq X_1$. A $\mathcal{DLR}^+$ *ABox* $\mathcal{A}$ is a finite set of *concept instance* axioms of the form $CN(O)$, *relation instance* axioms of the form $RN(U_1 : O_1,\ldots,U_n : O_n)$, and *same/distinct individual* axioms of the form $O_1 = O_2$ and $O_1 \neq O_2$, with $O_i \in \mathcal{O}$. It is easy to see that restricting ABox axioms to concept and relation names only does not affect the expressivity of $\mathcal{DLR}^+$ due to the availability of TBox axioms. A set of renaming axioms forms a *renaming schema*, which induces an equivalence relation $(\rightleftarrows, \mathcal{U})$ over the attributes $\mathcal{U}$, providing a partition of $\mathcal{U}$ into equivalence classes each one representing the alternative ways to name attributes. We write $[U]_{\Re}$ to denote the equivalence class of the attribute U w.r.t. the equivalence relation $(\rightleftarrows, \mathcal{U})$. We allow only *well founded* renaming schemas, namely schemas such that each equivalence class $[U]_{\Re}$ in the induced equivalence relation never contains two attributes from the same relation signature. We use the shortcut $U_1 \ldots U_n \rightleftarrows U'_1 \ldots U'_n$ to group many renaming axioms with the obvious meaning that $U_i \rightleftarrows U'_i$, for all $i = 1, \ldots, n$. A $\mathcal{DLR}^+$ knowledge base (KB) $\mathcal{KB} = (\mathcal{T}, \mathcal{A}, \Re)$ is composed by a TBox $\mathcal{T}$, an ABox $\mathcal{A}$, and a renaming schema $\Re$.

The renaming schema reconciles the attribute and the positional perspectives on relations (see also the similar perspectives in relational databases [Abiteboul *et al.*, 1995]). They are crucial when expressing both inclusion axioms and operators ($\sqcap$, $\sqcup$, $\setminus$) between relations, which make sense only over *union compatible* relations. Two relations R_1, R_2 are union compatible if their signatures are equal up to the attribute renaming induced by the renaming schema $\Re$, namely, $\tau(R_1) = \{U_1, \ldots, U_n\}$ and $\tau(R_2) = \{V_1, \ldots, V_n\}$ have the same arity n and $[U_i]_{\Re} = [V_i]_{\Re}$ for each $1 \leqslant i \leqslant n$. Notice that through the renaming schema, relations can use just local attribute names that can then be renamed when composing relations. Also note that it is obviously possible for the same attribute to appear in the signature of different relations.

Example 1. Consider the relation names R_1, R_2 s.t. $\tau(R_1) = \{W_1, W_2, W_3, W_4\}$, $\tau(R_2) = \{V_1, V_2, V_3, V_4, V_5\}$. The axiom:

rpn: I simplified the example a bit, but can roll back to previous example if needed

$$\exists[W_1, W_2]R_1 \sqsubseteq \exists^{\leqslant 1}[W_1, W_2]R_1$$

states that $\{W_1, W_2\}$ is the *multi-attribute key* of R_1.[1] It is also possible to express that there is a *functional dependency* from the attributes $\{V_3, V_4\}$ to $\{V_5\}$ of R_2 through the axiom:

$$\exists[V_3, V_4]R_2 \sqsubseteq \exists^{\leqslant 1}[V_3, V_4](\exists[V_3, V_4, V_5]R_2). \qquad (1)$$

Relationships between projections of relations can be expressed as follows. Consider the attribute renaming axiom $W_1W_2W_3 \rightleftarrows V_3V_4V_5$. Then, one can express that a projection of R_1 is a sub-relation of a projection of R_2 by the axiom

$$\exists[W_1, W_2, W_3]R_1 \sqsubseteq \exists[V_3, V_4, V_5]R_2.$$

The semantics of $\mathcal{DLR}^+$ uses of the notion of *labelled tuples* over a domain Δ: a *$\mathcal{U}$-labelled tuple over Δ* (or *tuple* for short) is a function $t \colon \mathcal{U} \to \Delta$. For $U \in \mathcal{U}$, we write $t[U]$ to refer to the domain element $d \in \Delta$ labelled by U, if the function t is defined for U—that is, if the attribute U is a label of the tuple t. Given $d_1, \ldots, d_n \in \Delta$, the expression $\langle U_1 \colon d_1, \ldots, U_n \colon d_n \rangle$ stands for the tuple t such that $t[U_i] = d_i$, for $1 \leqslant 1 \leqslant n$. The *projection* of the tuple t over the attributes $U_1, \ldots, U_k$ (i.e., the function t restricted to be undefined for the labels not in $U_1, \ldots, U_k$) is denoted by $t[U_1, \ldots, U_k]$. $T_\Delta(\mathcal{U})$ denotes the set of all $\mathcal{U}$-labelled tuples over Δ.

[1] $\exists[U_1, \ldots, U_k]R$ stands for $\exists^{\geqslant 1}[U_1, \ldots, U_k]R$.

$$
\begin{aligned}
\top^{\mathcal{I}} &= \Delta \\
\bot^{\mathcal{I}} &= \emptyset \\
(\neg C)^{\mathcal{I}} &= \top^{\mathcal{I}} \backslash C^{\mathcal{I}} \\
(C_1 \sqcap C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}} \\
(C_1 \sqcup C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cup C_2^{\mathcal{I}} \\
(\exists^{\lesseqgtr q}[U_i]R)^{\mathcal{I}} &= \{d \in \Delta \mid \ \left|\{t \in R^{\mathcal{I}} \mid t[\rho(U_i)] = d\}\right| \lesseqgtr q\} \\
(\circledcirc\, R)^{\mathcal{I}} &= \{d \in \Delta \mid d = \imath(t) \wedge t \in R^{\mathcal{I}}\} \\
(\odot\, RN)^{\mathcal{I}} &= \{d \in \Delta \mid d = \ell_{RN}(t) \wedge t \in RN^{\mathcal{I}}\} \\
(R_1 \backslash R_2)^{\mathcal{I}} &= R_1^{\mathcal{I}} \backslash R_2^{\mathcal{I}} \\
(R_1 \sqcap R_2)^{\mathcal{I}} &= R_1^{\mathcal{I}} \cap R_2^{\mathcal{I}} \\
(R_1 \sqcup R_2)^{\mathcal{I}} &= \{t \in R_1^{\mathcal{I}} \cup R_2^{\mathcal{I}} \mid \rho(\tau(R_1)) = \rho(\tau(R_2))\} \\
(\sigma_{U_i:C} R)^{\mathcal{I}} &= \{t \in R^{\mathcal{I}} \mid t[\rho(U_i)] \in C^{\mathcal{I}}\} \\
(\exists^{\lesseqgtr q}[U_1, \ldots, U_k]R)^{\mathcal{I}} &= \{\langle \rho(U_1) : d_1, \ldots, \rho(U_k) : d_k \rangle \in T_{\Delta}(\{\rho(U_1), \ldots, \rho(U_k)\}) \mid \\
&\qquad \left|\{t \in R^{\mathcal{I}} \mid t[\rho(U_1)] = d_1, \ldots, t[\rho(U_k)] = d_k\}\right| \lesseqgtr q \Big\}
\end{aligned}
$$

Fig. 3. Semantics of $\mathcal{DLR}^+$ expressions.

A $\mathcal{DLR}^+$ *interpretation*, $\mathcal{I} = (\Delta, \cdot^{\mathcal{I}}, \rho, \imath, L)$ consisting of a nonempty *domain* Δ, an *interpretation function* $\cdot^{\mathcal{I}}$, a *renaming function* ρ, a *global objectification function* $\imath$, and a family L containing one *local objectification function* ℓ_{RN_i} for each named relation $RN_i \in \mathcal{R}$. The renaming function ρ is a total function $\rho : \mathcal{U} \to \mathcal{U}$ representing a canonical renaming for all attributes. For brevity, we denote $\rho(\{U_1, \ldots, U_k\}) = \{\rho(U_1), \ldots, \rho(U_k)\}$. The global objectification function is an injective function, $\imath : T_{\Delta}(\mathcal{U}) \to \Delta$, associating a *unique* global identifier to each possible tuple. The local objectification functions, $\ell_{RN_i} : T_{\Delta}(\mathcal{U}) \to \Delta$, are distinct for each relation name in the signature, and as the global objectification function they are injective: they associate an identifier—which is unique only within the interpretation of a relation name—to each possible tuple. The interpretation function $\cdot^{\mathcal{I}}$ assigns a domain element to each individual, $O^{\mathcal{I}} \in \Delta$, a set of domain elements to each concept name, $CN^{\mathcal{I}} \subseteq \Delta$, and a set of $\mathcal{U}$-labelled tuples over Δ to each relation name conforming with its signature and the renaming function $RN^{\mathcal{I}} \subseteq T_{\Delta}(\{\rho(U) \mid U \in \tau(RN)\})$. This function $\cdot^{\mathcal{I}}$ is unambiguously extended over concept and relation expressions as specified in Figure 3.

The interpretation $\mathcal{I}$ satisfies the concept inclusion axiom $C_1 \sqsubseteq C_2$ if $C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$, it satisfies the relation inclusion axiom $R_1 \sqsubseteq R_2$ if $R_1^{\mathcal{I}} \subseteq R_2^{\mathcal{I}}$, it satisfies a concept instance axiom $CN(O)$ if $O^{\mathcal{I}} \in CN^{\mathcal{I}}$, it satisfies a relation instance axiom $RN(U_1 : O_1, \ldots, U_n : O_n)$ if $\langle \rho(U_1) \colon O_1^{\mathcal{I}}, \ldots, \rho(U_n) \colon O_n^{\mathcal{I}} \rangle \in RN^{\mathcal{I}}$, it satisfies a same individuals axiom $O_1 = O_2$ if $O_1^{\mathcal{I}} = O_2^{\mathcal{I}}$, and it satisfies a distinct individuals axiom $O_1 \neq O_2$ if $O_1^{\mathcal{I}} \neq O_2^{\mathcal{I}}$. $\mathcal{I}$ satisfies a renaming schema $\Re$ if for all $U, V \in \mathcal{U}$, (i) $\rho(U) \in [U]_{\Re}$, and (ii) if $V \in [U]_{\Re}$, then $\rho(U) = \rho(V)$. $\mathcal{I}$ is a *model* for a knowledge base $(\mathcal{T}, \mathcal{A}, \Re)$ if it satisfies all the axioms in the TBox $\mathcal{T}$ and in the ABox $\mathcal{A}$, and the renaming schema $\Re$.

KB satisfiability refers to the problem of deciding the existence of a model of a given knowledge base; *concept satisfiability* (resp. *relation satisfiability*) is the problem of deciding whether there is a model of the knowledge base that assigns a non-empty extension to a given concept (resp. relation); and *entailment* is to check whether a given knowledge base logically implies an axiom, that is, whenever all the models of the knowledge base are also models of the axiom. For instance, let $\mathcal{KB}$ be the KB containing all the axioms in Example 1. Then V_3, V_4 are a key for the relation R_2; that is,

$$\mathcal{KB} \models \exists[V_3, V_4]R_2 \sqsubseteq \exists^{\leqslant 1}[V_3, V_4]R_2.$$

3 Expressiveness of $\mathcal{DLR}^+$

$\mathcal{DLR}^+$ is a very expressive DL capable of asserting several kinds of constraints that are important in the context of relational databases, such as *equijoins*, *functional dependency* axioms, *key* axioms, *external uniqueness* axioms, and *identification* axioms.

An *equijoin* among two relations with disjoint signatures is the set of all combinations of tuples in the relations that are equal on their selected attribute names. Given two relations R_1 and R_2 with $\tau(R_1) = \{U^1, U_1^1, \ldots, U_{n_1}^1\}$ and $\tau(R_2) = \{U^2, U_1^2, \ldots, U_{n_2}^2\}$ their equijoin over U^1 and U^2 is the relation $R = R_1 \underset{U^1=U^2}{\bowtie} R_2$ with the signature $\tau(R) = \{U^1, U_1^1, \ldots, U_{n_1}^1, U_1^2, \ldots, U_{n_2}^2\}$, which can be expressed in $\mathcal{DLR}^+$ with the axioms:

$$\begin{array}{l} \exists[U^1, U_1^1, \ldots, U_{n_1}^1]R \equiv R_1 \sqcap (\sigma_{U^1:(\exists[U^1]R_1 \sqcap \exists[U^2]R_2)}R_1) \\ \exists[U^2, U_1^2, \ldots, U_{n_2}^2]R \equiv R_2 \sqcap (\sigma_{U^2:(\exists[U^1]R_1 \sqcap \exists[U^2]R_2)}R_2) \qquad U^2 \rightleftarrows U^1. \end{array}$$

A *functional dependency* axiom $(R : U_1 \ldots U_j \rightarrow U)$ states that the values of the attributes $U_1 \ldots U_j$ uniquely determine the value of the attribute U in the relation R. Formally, an interpretation $\mathcal{I}$ satisfies the functional dependency axiom $(R : U_1 \ldots U_j \rightarrow U)$ if, for all $s, t \in R^{\mathcal{I}}$, $s[U_1] = t[U_1], \ldots, s[U_j] = t[U_j]$ imply $s[U] = t[U]$. Functional dependency axioms are also called *internal uniqueness* axioms [Halpin and Morgan, 2008]. A functional dependency can be expressed in $\mathcal{DLR}^+$, assuming that $\{U_1, \ldots, U_j, U\} \subseteq \tau(R)$, with the axiom:

$$\exists[U_1, \ldots, U_j]R \sqsubseteq \exists^{\leqslant 1}[U_1, \ldots, U_j](\exists[U_1, \ldots, U_j, U]R).$$

A special case of functional dependency axiom is the *key* axiom $(R : U_1 \ldots U_j \rightarrow R)$, which states that the values of the key attributes $U_1 \ldots U_j$ of a relation R uniquely identify tuples in the relation itself. A key axiom can be expressed in $\mathcal{DLR}^+$, assuming that $\{U_1 \ldots U_j\} \subseteq \tau(R)$, with the axiom:

$$\exists[U_1, \ldots, U_j]R \sqsubseteq \exists^{\leqslant 1}[U_1, \ldots, U_j]R.$$

The *external uniqueness* axiom $([U^1]R_1 \downarrow \ldots \downarrow [U^h]R_h)$ states that the join R of the relations $R_1, \ldots, R_h$ via the attributes $U^1, \ldots, U^h$ has the joined attribute functionally dependent on all the others [Halpin and Morgan, 2008]. This can be expressed in $\mathcal{DLR}^+$ with the axioms:

$$R \equiv R_1 \underset{U^1=U^2}{\bowtie} \cdots \underset{U^{h-1}=U^h}{\bowtie} R_h$$
$$R : U_1^1, \ldots, U_{n_1}^1, \ldots, U_1^h, \ldots, U_{n_h}^h \to U^1$$

where $\tau(R_i) = \{U^i, U_1^i, \ldots, U_{n_i}^i\}$ with $1 \leqslant i \leqslant h$, and R is a fresh new relation name with signature $\tau(R) = \{U^1, U_1^1, \ldots, U_{n_1}^1, \ldots, U_1^h, \ldots, U_{n_h}^h\}$.

Identification axioms as defined in $\mathcal{DLR}_{\mathit{ifd}}$ [Calvanese *et al.*, 2001] are a variant of external uniqueness axioms, constraining only the elements of a concept C; they can be expressed in $\mathcal{DLR}^+$ with the following axiom:

$$[U^1]\sigma_{U_1:C} R_1 \downarrow \ldots \downarrow [U^h]\sigma_{U_h:C} R_h.$$

The $\mathcal{DLR}_{\mathit{ifd}}$ description logic introduced by [Calvanese *et al.*, 2001] extends $\mathcal{DLR}$ with functional dependencies and identification axioms, and therefore it is included in $\mathcal{DLR}^+$.

The $\mathcal{CFD}$ family of feature-based description logics have been designed primarily to support efficient PTIME reasoning services about object relational data sources [Toman and Weddell, 2009]. $\mathcal{CFD}$ includes *path functional dependencies* which can be expressed in $\mathcal{DLR}^+$ as identification axioms involving joined sequences of functional binary relations.

Since a $\mathcal{DLR}^+$ TBox can express both inclusion axioms among concepts and among arbitrary projections of relations, and arbitrary functional dependency axioms, the entailment problem is undecidable [Chandra and Vardi, 1985].

4 The $\mathcal{DLR}^\pm$ fragment of $\mathcal{DLR}^+$

Given a $\mathcal{DLR}^+$ knowledge base $(\mathcal{T}, \Re)$, the *projection signature* is the set $\mathscr{T}$ containing the signatures $\tau(RN)$ of the relations $RN \in \mathcal{R}$, the singletons associated with each attribute name $U \in \mathcal{U}$, and the relation signatures that appear explicitly in projection constructs in some relation inclusion axiom from $\mathcal{T}$, together with their implicit occurrences due to the renaming schema. Formally, $\mathscr{T}$ is the smallest set such that

1. $\tau(RN) \in \mathscr{T}$ for all $RN \in \mathcal{R}$;
2. $\{U\} \in \mathscr{T}$ for all $U \in \mathcal{U}$; and
3. $\{U_1, \ldots, U_k\} \in \mathscr{T}$ for all $\exists^{\lesseqgtr q}[V_1, \ldots, V_k]R$ appearing in $\mathcal{T}$ and $\{U_i, V_i\} \subseteq [U_i]_\Re$ for $1 \leqslant i \leqslant k$.

The *projection signature graph* is the directed acyclic graph $(\supset, \mathscr{T})$ whose sinks are the attribute singletons $\{U\}$. Given a set of attributes $\tau = \{U_1, \ldots, U_k\} \subseteq \mathcal{U}$, the the *projection signature graph dominated by* τ, denoted as $\mathscr{T}_\tau$, is the subgraph of $(\supset, \mathscr{T})$ containing all the nodes reachable from τ. Given two sets of attributes

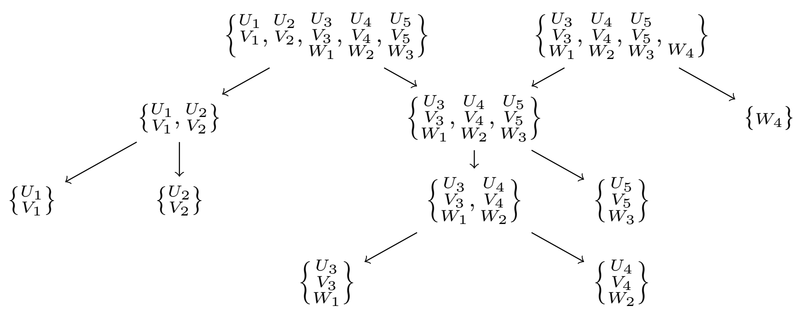

Fig. 4. The projection signature graph of the example.

$\tau_1, \tau_2 \subseteq \mathcal{U}$, $\textsc{path}_{\mathscr{T}}(\tau_1, \tau_2)$ denotes the set of paths in $(\supset, \mathscr{T})$ between τ_1 and τ_2. Note that $\textsc{path}_{\mathscr{T}}(\tau_1, \tau_2) = \varnothing$ both, when a path does not exist and when $\tau_1 \subseteq \tau_2$. The notation $\textsc{child}_{\mathscr{T}}(\tau_1, \tau_2)$ means that τ_2 is a child of τ_1 in $(\supset, \mathscr{T})$.

Definition 2. *A $\mathcal{DLR}^{\pm}$* knowledge base *is a $\mathcal{DLR}^{+}$ KB that satisfies the following two conditions:*

1. *the projection signature graph $(\supset, \mathscr{T})$ forms a multitree; i.e., for every node $\tau \in \mathscr{T}$, the graph $\mathscr{T}_\tau$ is a tree; and*
2. *for every projection construct $\exists^{\lesseqqgtr q}[U_1, \ldots, U_k]R$ appearing in $\mathcal{T}$, if $q > 1$ then the length of the path $\textsc{path}_{\mathscr{T}}(\tau(R), \{U_1, \ldots, U_k\})$ is 1.*

Essentially, the conditions in $\mathcal{DLR}^{\pm}$ restrict $\mathcal{DLR}^{+}$ in the way that multiple projections of relations may appear in the knowledge base. In particular, observe that in $\mathcal{DLR}^{\pm}$ $\textsc{path}_{\mathscr{T}}$ is necessarily functional, due to the multitree restriction. Figure 4 shows that the projection signature graph of the knowledge base from Example 1 is indeed a multitree. Note that in the figure we have collapsed equivalent attributes in a unique equivalence class, according to the renaming schema. Since all its projection constructs $q = 1$, this knowledge base belongs to $\mathcal{DLR}^{\pm}$.

It is easy to see that $\mathcal{DLR}$ is included in $\mathcal{DLR}^{\pm}$, since the projection signature graph of any $\mathcal{DLR}$ knowledge base is always a degenerate multitree with maximum depth equal to 1.

$\mathcal{DLR}_{ifd}$ [Calvanese *et al.*, 2001] extended with unary functional dependencies is also included in $\mathcal{DLR}^{\pm}$, with the proviso that projections of relations in the knowledge base form a multitree projection signature graph. Since (unary) functional dependencies are expressed via the inclusions of projections of relations (see, e.g., the functional dependency (1) in the previous example), by constraining the projection signature graph to be a multitree, the possibility to build combinations of functional dependencies as the ones in [Calvanese *et al.*, 2001] leading to undecidability is ruled out.

$$
\begin{aligned}
(\neg C)^\dagger &= \neg C^\dagger \\
(C_1 \sqcap C_2)^\dagger &= C_1^\dagger \sqcap C_2^\dagger \\
(C_1 \sqcup C_2)^\dagger &= C_1^\dagger \sqcup C_2^\dagger \\
(\exists^{\lesseqqgtr q}[U_i]R)^\dagger &= \exists^{\lesseqqgtr q} \left(\textsc{path}_{\mathscr{T}}(\tau(R),\{U_i\})^\dagger\right)^- . R^\dagger \\
(\circledcirc R)^\dagger &= R^\dagger \\
(\odot RN)^\dagger &= A^l_{RN} \\
\\
(R_1 \backslash R_2)^\dagger &= R_1^\dagger \sqcap \neg R_2^\dagger \\
(R_1 \sqcap R_2)^\dagger &= R_1^\dagger \sqcap R_2^\dagger \\
(R_1 \sqcup R_2)^\dagger &= R_1^\dagger \sqcup R_2^\dagger \\
(\sigma_{U_i:C} R)^\dagger &= R^\dagger \sqcap \forall \textsc{path}_{\mathscr{T}}(\tau(R),\{U_i\})^\dagger . C^\dagger \\
(\exists^{\lesseqqgtr q}[U_1,\ldots,U_k]R)^\dagger &= \exists^{\lesseqqgtr q} \left(\textsc{path}_{\mathscr{T}}(\tau(R),\{U_1,\ldots,U_k\})^\dagger\right)^- . R^\dagger
\end{aligned}
$$

Fig. 5. The mapping for concept and relation expressions.

Also note that $\mathcal{DLR}^\pm$ is able to correctly express the UML fragment as introduced in [Berardi *et al.*, 2005; Artale *et al.*, 2007] and the ORM fragment as introduced in [Franconi and Mosca, 2013].

5 Mapping $\mathcal{DLR}^\pm$ to $\mathcal{ALCQI}$

We show that reasoning in $\mathcal{DLR}^\pm$ is ExpTime-complete by providing a mapping from $\mathcal{DLR}^\pm$ knowledge bases to $\mathcal{ALCQI}$ knowledge bases; the reverse mapping from $\mathcal{ALCQI}$ knowledge bases to $\mathcal{DLR}$ knowledge bases is well known. The proof is based on the fact that reasoning with $\mathcal{ALCQI}$ knowledge bases is ExpTime-complete [Baader *et al.*, 2003]. We adapt and extend the mapping presented for $\mathcal{DLR}$ in [Calvanese *et al.*, 1998] and† then adapted by [Horrocks *et al.*, 2000] to deal with ABoxes possibly without the UNA.

A: added

In the following we use the shortcut $(S_1 \circ \ldots \circ S_n)^-$ for $S_n^- \circ \ldots \circ S_1^-$, the shortcut $\exists^{\lesseqqgtr 1} S_1 \circ \ldots \circ S_n . C$ for $\exists^{\lesseqqgtr 1} S_1 . \ldots . \exists^{\lesseqqgtr 1} S_n . C$, the shortcut $\forall S_1 \circ \ldots \circ S_n . C$ for $\forall S_1 . \ldots . \forall S_n . C$, while† the concept expressions $\exists^{\lesseqqgtr 1} \bot . C$ and $\forall \bot . C$ have to be considered as the $\bot$ concept. Note† that the shortcut for qualified number restrictions is limited to the case $q = 1$. For a relation instance axiom in the ABox of the form $RN(U_1 : O_1, \ldots, U_n : O_n)$ we use the shortcut $RN(t)$, with $t = \langle U_1 : O_1, \ldots, U_n : O_n \rangle$ a *relation instance*, namely a $\mathcal{U}$-labelled tuple over the set of individuals in $\mathcal{O}$.

A: changed

A: sentence changed, the original is commented

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A}, \Re)$ be a $\mathcal{DLR}^\pm$ knowledge base. We first preprocess the $\mathcal{DLR}^\pm$ knowledge base by transforming it into a logically equivalent one as follows: for each equivalence class $[U]_\Re$ a single *canonical* representative of the class is chosen, and the $\mathcal{KB}$ is consistently rewritten by substituting each attribute with its canonical representative. After this rewriting, the renaming schema does not play any role in the mapping.

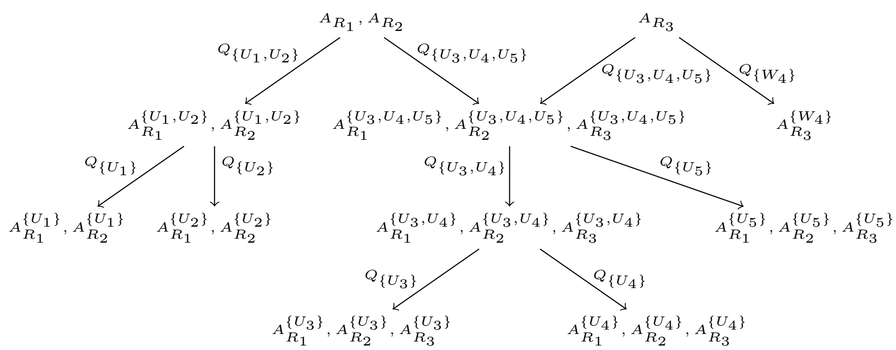

Fig. 6. The $\mathcal{ALCQI}$ signature generated by the example.

Let's first introduce a mapping function $\cdot^\dagger$ from $\mathcal{DLR}^\pm$ concept and relation expressions to $\mathcal{ALCQI}$ concepts. Starting with atomic expressions, the mapping function $\cdot^\dagger$ maps each concept name *CN* in the $\mathcal{DLR}^\pm$ knowledge base to an $\mathcal{ALCQI}$ concept name *CN* and each relation name *RN* in the $\mathcal{DLR}^\pm$ knowledge base to an $\mathcal{ALCQI}$ concept name A_{RN} (its global reification). For each relation name *RN*, the $\mathcal{ALCQI}$ signature also includes a concept name A^l_{RN} and a role name Q_{RN}, in order to capture the local objectification. The mapping $\cdot^\dagger$ is extended to concept and relation expressions as in Figure 5.

The mapping crucially uses the projection signature graph structure to map projections and selections, by accessing paths in the projection signature $\mathscr{T}$ associated to the $\mathcal{DLR}^\pm$ knowledge base. If there is a path from τ to τ' in $\mathscr{T}$— i.e., $\textsc{path}_{\mathscr{T}}(\tau,\tau') = \tau,\tau_1,\ldots,\tau_n,\tau'$—then its mapping is an $\mathcal{ALCQI}$ role chain expression using role names Q_{τ_i} as follows:

$$\textsc{path}_{\mathscr{T}}(\tau,\tau')^\dagger = Q_{\tau_1} \circ \ldots \circ Q_{\tau_n} \circ Q_{\tau'}.$$

with $\textsc{path}_{\mathscr{T}}(\tau,\tau')^\dagger = \bot$ if $\textsc{path}_{\mathscr{T}}(\tau,\tau') = \varnothing$.

The $\mathcal{ALCQI}$ signature also includes a concept name $A^{\tau_i}_{RN}$ for each projected signature τ_i in the projection signature graph dominated by $\tau(RN)$, $\tau_i \in \mathscr{T}_{\tau(RN)}$ (to capture global reifications of the projections of *RN*). Note that $A^{\tau(RN)}_{RN}$ coincides with A_{RN}.

Intuitively, the mapping reifies each node in the projection signature graph: the target $\mathcal{ALCQI}$ signature of the example of the previous section is partially presented in Fig. 6, together with the projection signature graph. Each node is labelled with the corresponding (global) reification concept ($A^{\tau_j}_{R_i}$), for each relation name R_i and each projected signature τ_j in the projection signature graph dominated by $\tau(R_i)$, while the edges are labelled by the roles (Q_{τ_i}) needed for the reification.

Note that in $\mathcal{DLR}^\pm$ the cardinalities on a path are restricted to the case $q = 1$ whenever a path is of length greater than 1, so we still remain within the $\mathcal{ALCQI}$ description logic when the mapping applies to cardinalities. So, if we need to express a cardinality constraint $\exists^{\lessgtr q}[U_i]R$, with $q > 1$, then U_i should not be mentioned in any other projection of the relation R in such a way that $|\textsc{path}_{\mathscr{T}}(\tau(R), \{U_i\})| = 1$.

In order to explain the need for the path function in the mapping, notice that a relation is reified according to the decomposition dictated by the projection signature graph it dominates. Thus, to access an attribute U_j of a relation R_i it is necessary to follow the path through the projections that use that attribute. This path is a role chain from the signature of the relation (the root) to the attribute as returned by the $\textsc{path}_{\mathscr{T}}(\tau(R_i), U_i)$ function. For example, considering Fig. 6, in order to access the attribute U_4 of the relation R_3 in the expression $(\sigma_{U_4:C} R_3)$, the path $\textsc{path}_{\mathscr{T}}(\tau(R_3), \{U_4\})^{\dagger}$ is equal to the role chain $Q_{\{U_3,U_4,U_5\}} \circ Q_{\{U_3,U_4\}} \circ Q_{\{U_4\}}$, so that $(\sigma_{U_4:C} R_3)^{\dagger} \;=\; A_{R_3} \sqcap \forall Q_{\{U_3,U_4,U_5\}} \circ Q_{\{U_3,U_4\}} \circ Q_{\{U_4\}}.C$. Similar considerations can be done when mapping cardinalities over relation projections.

Let $\mathcal{KB} = (\mathcal{T}, \mathcal{A})$ be a $\mathcal{DLR}^{\pm}$ knowledge base with a signature $(\mathcal{C}, \mathcal{R}, \mathcal{U}, \tau)$. The mapping $\gamma(\mathcal{KB}) = (\gamma(\mathcal{T}), \gamma(\mathcal{A}))$ is defined as the following $\mathcal{ALCQI}$ KB:

$$
\begin{aligned}
\gamma(\mathcal{KB}) \quad = \quad & \gamma_{dsj} \;\cup \bigcup_{RN \in \mathcal{R}} \gamma_{rel}(RN) \;\cup \bigcup_{RN \in \mathcal{R}} \gamma_{lobj}(RN) \;\cup \\
& \bigcup_{C_1 \sqsubseteq C_2 \in \mathcal{KB}} C_1^{\dagger} \sqsubseteq C_2^{\dagger} \;\cup \bigcup_{R_1 \sqsubseteq R_2 \in \mathcal{KB}} R_1^{\dagger} \sqsubseteq R_2^{\dagger},
\end{aligned}
$$

where

$$
\begin{aligned}
\gamma_{dsj} &= \left\{ A^{\tau_i}_{RN_1} \sqsubseteq \neg A^{\tau_j}_{RN_2} \mid RN_1, RN_2 \in \mathcal{R}, \tau_i, \tau_j \in \mathscr{T}, |\tau_i| \geqslant 2, |\tau_j| \geqslant 2, \tau_i \neq \tau_j \right\} \\
\gamma_{rel}(RN) &= \bigcup_{\tau_i \in \mathscr{T}_{\tau(RN)}} \; \bigcup_{\textsc{child}_{\mathscr{T}}(\tau_i, \tau_j)} \left\{ A^{\tau_i}_{RN} \sqsubseteq \exists Q_{\tau_j}.A^{\tau_j}_{RN},\; \exists^{\geqslant 2} Q_{\tau_j}.\top \sqsubseteq \bot \right\} \\
\gamma_{lobj}(RN) &= \{ A_{RN} \sqsubseteq \exists Q_{RN}.A^{l}_{RN},\; \exists^{\geqslant 2} Q_{RN}.\top \sqsubseteq \bot, \\
& \qquad A^{l}_{RN} \sqsubseteq \exists Q^{-}_{RN}.A_{RN},\; \exists^{\geqslant 2} Q^{-}_{RN}.\top \sqsubseteq \bot \}
\end{aligned}
$$

Intuitively, γ_{dsj} ensures that relations with different signatures are disjoint, thus, e.g., enforcing the union compatibility. The axioms in γ_{rel} introduce classical reification axioms for each relation and its relevant projections. The axioms in γ_{lobj} make sure that each local objectification differs form the global one.

As for the mapping of the ABox, we map each individual O in the $\mathcal{DLR}^{\pm}$ ABox to an $\mathcal{ALCQI}$ individual O (the mapping for individuals behaves as the identity function). Each relation instance occurring in $\mathcal{A}$ is mapped via an injective function, ξ, to a distinct individual, i.e., $\xi : T_{\mathcal{O}}(\mathcal{U}) \rightarrow \mathcal{O}_{\mathcal{ALCQI}}$, with $\mathcal{O}_{\mathcal{ALCQI}} = \mathcal{O} \cup \mathcal{O}^t$ being the set of individuals in $\gamma(\mathcal{KB})$, $\mathcal{O} \cap \mathcal{O}^t = \varnothing$ and

$$
\xi(t) \;=\; \begin{cases} O \in \mathcal{O}, & \text{if } t = \langle U : O \rangle \\ O \in \mathcal{O}^t, & \text{otherwise.} \end{cases}
$$

The mapping $\gamma(\mathcal{A})$ of the ABox, similarly to the mapping presented in [Horrocks *et al.*, 2000], introduces a new concept name Q_o for every individual in $O \in \mathcal{O}$ and a new concept name Q_t for every relation instance occurring in $\mathcal{A}$. Then,

$\gamma(\mathcal{A})$ is as follows:

$$\begin{aligned}
\gamma(\mathcal{A}) = &\{CN^{\dagger}(O) \mid CN(O) \in \mathcal{A}\} \cup && (2)\\
&\{O_1 \neq O_2 \mid O_1 \neq O_2 \in \mathcal{A}\} \ \cup \ \{O_1 = O_2 \mid O_1 = O_2 \in \mathcal{A}\} \cup && (3)\\
&\{A^{\tau_i}_{RN}(\xi(t[\tau_i])) \mid RN(t) \in \mathcal{A} \textit{ and } \tau_i \in \mathscr{T}_{\tau(RN)}\} \cup && (4)\\
&\{Q_{\tau_j}\big(\xi(t[\tau_i]), \xi(t[\tau_j])\big) \mid \text{CHILD}_{\mathscr{T}}(\tau_i, \tau_j)\} && (5)\\
&\{Q_o(O) \mid O \in \mathcal{O}\} \ \cup \ \{Q_t(O_1),\ Ax(Q_t) \mid t = \langle U_1 \!:\! O_1, \ldots, U_n \!:\! O_n\rangle \in A\}, && (6)
\end{aligned}$$

where $Ax(Q_t)$ stands for the following axiom:

$$Q_t \sqsubseteq \exists^{\leq 1}\big(\text{PATH}_{\mathscr{T}}(\tau(t), \{U_1\})^{\dagger}\big)^{-}.\Big(\exists\big(\text{PATH}_{\mathscr{T}}(\tau(t), \{U_2\})^{\dagger}\big).\,Q_{o_2} \sqcap \ldots \sqcap \exists\big(\text{PATH}_{\mathscr{T}}(\tau(t), \{U_n\})^{\dagger}\big).\,Q_{o_n}\Big) \quad (7)$$

Intuitively, (4) and (5) reify each relation instance occurring in $\mathcal{A}$ using the projection signature of the relation instance itself. The formulas (6) and (7) guarantee that there is exactly one $\mathcal{ALCQI}$ individual reifying a given relation instance.

Clearly, the size of $\gamma(\mathcal{KB})$ is polynomial in the size of $\mathcal{KB}$ (under the same coding of the numerical parameters), and thus we are able to state the main result of this paper.†

A: the Lemma must be moved to another Section!

Lemma 3. *The problems of concept and relation satisfiability and of entailment in $\mathcal{DLR}^{\pm}$ are reducible to $\mathcal{DLR}^{\pm}$ KB satisfiability.*

Proof. TO BE PROVED

Theorem 4. *A $\mathcal{DLR}^{\pm}$ knowledge base $\mathcal{KB}$ is satisfiable iff the $\mathcal{ALCQI}$ knowledge base $\gamma(\mathcal{KB})$ is satisfiable.*

Proof. We assume that the $\mathcal{KB}$ is consistently rewritten by substituting each attribute with its canonical representative, thus, we do not have to deal with the renaming of attributes. Furthermore, we extend the function $\imath$ to singleton tuples with the meaning that $\imath(\langle U_i : d_i\rangle) = d_i$.
($\Rightarrow$) Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, \rho, \imath, \ell_{RN_1}, \ldots)$ be a model for a $\mathcal{DLR}^{\pm}$ knowledge base $\mathcal{KB}$. To construct a model $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ for the $\mathcal{ALCQI}$ knowledge base $\gamma(\mathcal{KB})$ we set $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$. Furthermore, we set: $(CN^{\dagger})^{\mathcal{J}} = (CN)^{\mathcal{I}}$, for every atomic concept $CN \in \mathcal{C}$, while for every $RN \in \mathcal{R}$ and $\tau_i \in \mathscr{T}_{\tau(RN)}$ we set

$$(A^{\tau_i}_{RN})^{\mathcal{J}} = \{\imath(\langle U_1 : d_1, \ldots, U_k : d_k\rangle) \mid \{U_1, \ldots, U_k\} = \tau_i \text{ and } \exists t \in RN^{\mathcal{I}}.\, t[U_1] = d_1, \ldots, t[U_k] = d_k\}. \quad (8)$$

For each role name Q_{τ_i}, $\tau_i \in \mathscr{T}$, we set

$$(Q_{\tau_i})^{\mathcal{J}} = \{(d_1, d_2) \in \Delta^{\mathcal{J}} \times \Delta^{\mathcal{J}} \mid \exists t \in RN^{\mathcal{I}} \text{ s.t. } d_1 = \imath(t[\tau_j]), d_2 = \imath(t[\tau_i]) \text{ and } \text{CHILD}_{\mathscr{T}}(\tau_j, \tau_i), \text{ for some } RN \in \mathcal{R}\}. \quad (9)$$

For every $RN \in \mathcal{R}$ we set

$$Q^{\mathcal{J}}_{RN} = \{(d_1, d_2) \in \Delta^{\mathcal{J}} \times \Delta^{\mathcal{J}} \mid \exists t \in RN^{\mathcal{I}} \text{ s.t. } d_1 = \imath(t) \text{ and } d_2 = \ell_{RN}(t)\}, \quad (10)$$

and

$$(A^l_{RN})^{\mathcal{J}} = \{\ell_{RN}(t) \mid t \in RN^{\mathcal{I}}\}. \quad (11)$$

We now show that $\mathcal{J}$ is indeed a model of $\gamma(\mathcal{KB})$.

1. $\mathcal{J} \models \gamma_{dsj}$. This is a direct consequence of the fact that $\imath$ is an injective function and that tuples with different aryties are different tuples.
2. $\mathcal{J} \models \gamma_{rel}(RN)$, for every $RN \in \mathcal{R}$. We show that, for each τ_i, τ_j s.t. $\text{CHILD}_{\mathscr{T}}(\tau_i, \tau_j)$ and $\tau_i \in \mathscr{T}_{\tau(RN)}$, $\mathcal{J} \models A^{\tau_i}_{RN} \sqsubseteq \exists Q_{\tau_j} . A^{\tau_j}_{RN}$ and $\mathcal{J} \models \exists^{\geqslant 2} Q_{\tau_j} . \top \sqsubseteq \bot$:
   - $\mathcal{J} \models A^{\tau_i}_{RN} \sqsubseteq \exists Q_{\tau_j} . A^{\tau_j}_{RN}$. Let $d \in (A^{\tau_i}_{RN})^{\mathcal{J}}$, by (8), $\exists t \in RN^{\mathcal{I}}$ s.t. $d = \imath(t[\tau_i])$. Since $\text{CHILD}_{\mathscr{T}}(\tau_i, \tau_j)$, then $\exists d' = \imath(t[\tau_j])$ and, by (9), $(d, d') \in Q^{\mathcal{J}}_{\tau_j}$, while by (8), $d' \in (A^{\tau_j}_{RN})^{\mathcal{J}}$. Thus, $d \in (\exists Q_{\tau_j} . A^{\tau_j}_{RN})^{\mathcal{J}}$.
   - $\mathcal{J} \models \exists^{\geqslant 2} Q_{\tau_j} . \top \sqsubseteq \bot$. The fact that each Q_{τ_j} is interpreted as a funcional role is a direct consequence of the construction (9) and the fact that $\imath$ is an injective function.
3. $\mathcal{J} \models \gamma_{lobj}(RN)$, for every $RN \in \mathcal{R}$. Similar as above, considering the fact that each ℓ_{RN} is an injective function and equations (10)-(11).
4. $\mathcal{J} \models C^{\dagger}_1 \sqsubseteq C^{\dagger}_2$ and $\mathcal{J} \models R^{\dagger}_1 \sqsubseteq R^{\dagger}_2$. Since $\mathcal{I} \models C_1 \sqsubseteq C_2$ and $\mathcal{I} \models R_1 \sqsubseteq R_2$, It is enough to show the following:
   - $d \in C^{\mathcal{I}}$ iff $d \in (C^{\dagger})^{\mathcal{J}}$, for all $\mathcal{DLR}^{\pm}$ concepts;
   - $t \in R^{\mathcal{I}}$ iff $\imath(t) \in (R^{\dagger})^{\mathcal{J}}$, for all $\mathcal{DLR}^{\pm}$ relations.

   Before we proceed with the proof, it is easy to show by structural induction that the following property holds:

   $$\text{If } \imath(t) \in R^{\dagger\mathcal{J}} \text{ then } \exists \imath(t') \in RN^{\dagger\mathcal{J}} \text{ s.t. } t = t'[\tau(R)], \text{ for some } RN \in \mathcal{R}. \quad (12)$$

   We now proceed with the proof by structural induction. The base cases, for atomic concepts and roles, are immediate form the definition of both $CN^{\mathcal{J}}$ and $RN^{\mathcal{J}}$. The cases where complex concepts and relations are constructed using either boolean operators or global reification are easy to show. We thus show only the following cases.
   Let $d \in (\odot\, RN)^{\mathcal{I}}$. Then, $d = \ell_{RN}(t)$ with $t \in RN^{\mathcal{I}}$. By induction, $\imath(t) \in A^{\mathcal{J}}_{RN}$ and, by $\gamma_{lobj}(RN)$, there is a $d' \in \Delta^{\mathcal{J}}$ s.t. $(\imath(t), d') \in Q^{\mathcal{J}}_{RN}$ and $d' \in (A^l_{RN})^{\mathcal{J}}$. By (10), $d' = \ell_{RN}(t)$ and, since ℓ_{RN} is injective, $d' = d$. Thus, $d \in (\odot\, RN)^{\dagger\mathcal{J}}$.
   Let $d \in (\exists^{\geqslant q}[U_i]R)^{\mathcal{I}}$. Then, there are different $t_1, \ldots, t_q \in R^{\mathcal{I}}$ s.t. $t_l[U_i] = d$, for all $l = 1, \ldots, q$. By induction, $\imath(t_l) \in R^{\dagger\mathcal{J}}$ while, by (12), $\imath(t'_l) \in RN^{\dagger\mathcal{J}}$, for some atomic relation $RN \in \mathcal{R}$ and a tuple t'_l s.t. $t_l = t'_l[\tau(R)]$. By $\gamma_{rel}(RN)$ and (9), $(\imath(t'_l), \imath(t_l)) \in (\text{PATH}_{\mathscr{T}}(\tau(RN), \tau(R))^{\dagger})^{\mathcal{J}}$ and $(\imath(t_l), d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^{\dagger})^{\mathcal{J}}$. Since $\imath$ is injective, $\imath(t_l) \neq \imath(t_j)$ when $l \neq j$, thus, $d \in (\exists^{\geqslant q}[U_i]R)^{\dagger\mathcal{J}}$.
   Let $t \in (\sigma_{U_i:C}R)^{\mathcal{I}}$. Then, $t \in R^{\mathcal{I}}$ and $t[U_i] \in C^{\mathcal{I}}$ and, by induction, $\imath(t) \in R^{\dagger\mathcal{J}}$ and $t[U_i] \in C^{\dagger\mathcal{J}}$. As before, by $\gamma_{rel}(RN)$ and by (9) and (12), $(\imath(t), t[U_i]) \in$

$(\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger)^{\mathcal{J}}$. Since $\text{PATH}_{\mathscr{T}}(\tau(R), U_i)^\dagger$ is functional, then we have that $\imath(t) \in (\sigma_{U_i:C} R)^{\dagger\mathcal{J}}$.

Let $t \in (\exists[U_1, \ldots, U_k]R)^{\mathcal{I}}$. Then, there is a tuple $t' \in R^{\mathcal{I}}$ s.t. $t'[U_1, \ldots, U_k] = t$ and, by induction, $\imath(t') \in R^{\dagger\mathcal{J}}$. As before, by $\gamma_{rel}(RN)$ and by (9) and (12), we can show that $(\imath(t'), \imath(t)) \in \text{PATH}_{\mathscr{T}}(\tau(R), \{U_1, \ldots, U_k\})^{\dagger\mathcal{J}}$ and thus $\imath(t) \in (\exists[U_1, \ldots, U_k]R)^{\dagger\mathcal{J}}$.

All the other cases can be proved in a similar way. We now show the vice versa.

Let $d \in (\bigodot RN)^{\dagger\mathcal{J}}$. Then, $d \in (A^l_{RN})^{\mathcal{J}}$ and $d = l_{RN}(t)$, for some $t \in RN^{\mathcal{I}}$, i.e., $d \in (\bigodot RN)^{\mathcal{I}}$.

Let $d \in (\exists^{\geqslant q}[U_i]R)^{\dagger\mathcal{J}}$. Then, there are different $d_1, \ldots, d_q \in \Delta^{\mathcal{J}}$ s.t. $(d_l, d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger)^{\mathcal{J}}$ and $d_l \in R^{\dagger\mathcal{J}}$, for $l = 1, \ldots, q$. By induction, each $d_l = \imath(t_l)$ and $t_l \in R^{\mathcal{I}}$. Since $\imath$ is injective, then $t_l \neq t_j$ for all $l, j = 1, \ldots, q$, $l \neq j$. We need to show that $t_l[U_i] = d$, for all $l = 1, \ldots, q$. By (9) and the fact that $(d_l, d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger)^{\mathcal{J}}$, then $d = \imath(t_l[U_i]) = t_l[U_i]$.

Let $\imath(t) \in (\sigma_{U_i:C} R)^{\dagger\mathcal{J}}$. Then, $\imath(t) \in R^{\dagger\mathcal{J}}$ and, by induction, $t \in R^{\mathcal{I}}$. Let $t[U_i] = d$. We need to show that $d \in C^{\mathcal{I}}$. By $\gamma_{rel}(RN)$ and by (9) and (12), $(\imath(t), d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger)^{\mathcal{J}}$, then $d \in C^{\dagger\mathcal{J}}$ and, by induction, $d \in C^{\mathcal{I}}$.

Let $\imath(t) \in (\exists[U_1, \ldots, U_k]R)^{\dagger\mathcal{J}}$. Then, there is $d \in \Delta^{\mathcal{J}}$ s.t.

$$(d, \imath(t)) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_1, \ldots, U_k\})^\dagger)^{\mathcal{J}}$$

and $d \in R^{\mathcal{J}}$. By induction, $d = \imath(t')$ and $t' \in R^{\mathcal{I}}$. By (9), $\imath(t) = \imath(t'[U_1, \ldots, U_k])$, i.e., $t = t'[U_1, \ldots, U_k]$. Thus, $t \in (\exists[U_1, \ldots, U_k]R)^{\mathcal{I}}$.

($\Leftarrow$) Let $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ be a model for the knowledge base $\gamma(\mathcal{KB})$. Without loss of generality, we can assume that $\mathcal{J}$ is a *tree model*. We then construct a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}, \rho, \imath, \ell_{RN_1}, \ldots)$ for a $\mathcal{DLR}^{\pm}$ knowledge base $\mathcal{KB}$. We set: $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$, $CN^{\mathcal{I}} = (CN^\dagger)^{\mathcal{J}}$, for every atomic concept $CN \in \mathcal{C}$, while, for every $RN \in \mathcal{R}$, we set:

$$\begin{aligned} RN^{\mathcal{I}} = \{t = \langle U_1 : d_1, \ldots, U_n : d_n \rangle \in T_{\Delta^{\mathcal{I}}}(\tau(RN)) \mid \exists d \in A^{\mathcal{J}}_{RN} \text{ s.t.} \\ (d, t[U_i]) \in (\text{PATH}_{\mathscr{T}}(\tau(RN), \{U_i\})^\dagger)^{\mathcal{J}} \text{ for } i = 1, \ldots, n\}. \quad (13) \end{aligned}$$

Since $\mathcal{J}$ satisfies $\gamma_{rel}(RN)$, then, for every $d \in A^{\mathcal{J}}_{RN}$ there is a unique tuple $\langle U_1 : d_1, \ldots, U_n : d_n \rangle \in RN^{\mathcal{I}}$, we say that d *generates* $\langle U_1 : d_1, \ldots, U_n : d_n \rangle$ and, in symbols, $d \to \langle U_1 : d_1, \ldots, U_n : d_n \rangle$. Furthermore, since $\mathcal{J}$ is tree shaped, to each tuple corresponds a unique d that generates it. Thus, let $d \to \langle U_1 : d_1, \ldots, U_n : d_n \rangle$, by setting $\imath(\langle U_1 : d_1, \ldots, U_n : d_n \rangle) = d$ and

$$\begin{aligned} \imath(\langle U_1 : d_1, \ldots, U_n : d_n \rangle[\tau_i]) = d_{\tau_i}, \text{ s.t.} \\ (d, d_{\tau_i}) \in (\text{PATH}_{\mathscr{T}}(\{U_1, \ldots, U_n\}, \tau_i)^\dagger)^{\mathcal{J}}, \quad (14) \end{aligned}$$

for all $\tau_i \in \mathscr{T}_{\{U_1, \ldots, U_n\}}$, then, the function $\imath$ is as required.

By setting

$$\begin{aligned} \ell_{RN}(\langle U_1 : d_1, \ldots, U_n : d_n \rangle) = d, \text{ s. t.} \\ (\imath(\langle U_1 : d_1, \ldots, U_n : d_n \rangle), d) \in Q^{\mathcal{J}}_{RN}, \quad (15) \end{aligned}$$

by $\gamma_{lobj}(RN)$, both Q_{RN} and its inverse are interpreted as a functional roles by $\mathcal{J}$, thus the function ℓ_{RN} is as required.
It is easy to show by structural induction that the following property holds:

$$\text{If } t \in R^{\mathcal{I}} \text{ then } \exists t' \in RN^{\mathcal{I}} \text{ s.t. } t = t'[\tau(R)], \text{ for some } RN \in \mathcal{R}. \tag{16}$$

We now show that $\mathcal{I}$ is indeed a model of $\mathcal{KB}$, i.e., $\mathcal{I} \models C_1 \sqsubseteq C_2$ and $\mathcal{I} \models R_1 \sqsubseteq R_2$. As before, since $\mathcal{J} \models C_1^\dagger \sqsubseteq C_2^\dagger$ and $\mathcal{J} \models R_1^\dagger \sqsubseteq R_2^\dagger$, it is enough to show the following:

- $d \in C^{\mathcal{I}}$ iff $d \in (C^\dagger)^{\mathcal{J}}$, for all $\mathcal{DLR}^\pm$ concepts;
- $t \in R^{\mathcal{I}}$ iff $\imath(t) \in (R^\dagger)^{\mathcal{J}}$, for all $\mathcal{DLR}^\pm$ relations.

The proof is by structural induction. The base cases are trivially true. Similarly for the boolean operators and global reification. We thus show only the following cases.
Let $d \in (\bigodot RN)^{\mathcal{I}}$. Then, $d = \ell_{RN}(t)$ with $t \in RN^{\mathcal{I}}$. By induction, $\imath(t) \in A_{RN}^{\mathcal{J}}$ and, by $\gamma_{lobj}(RN)$, there is a $d' \in \Delta^{\mathcal{J}}$ s.t. $(\imath(t), d') \in Q_{RN}^{\mathcal{J}}$ and $d' \in (A_{RN}^l)^{\mathcal{J}}$. By (15), $d = d'$ and thus, $d \in (\bigodot RN)^{\dagger\mathcal{J}}$.
Let $d \in (\exists^{\geqslant q}[U_i]R)^{\mathcal{I}}$. Then, there are different $t_1, \ldots, t_q \in R^{\mathcal{I}}$ s.t. $t_l[U_i] = d$, for all $l = 1, \ldots, q$. For each t_l, by (16), there is a $t'_l \in RN^{\mathcal{I}}$ s.t. $t_l = t'_l[\tau(R)]$, for some $RN \in \mathcal{R}$, while, by induction, $\imath(t_l) \in R^{\dagger\mathcal{J}}$ and $\imath(t'_l) \in RN^{\dagger\mathcal{J}}$. Thus, $t'_l[U_i] = t_l[U_i] = d$ and, by (13), $(\imath(t'_l), d) \in (\text{PATH}_{\mathscr{T}}(\tau(RN), \{U_i\})^\dagger)^{\mathcal{J}}$ while, by (14), $(\imath(t'_l), \imath(t_l)) \in (\text{PATH}_{\mathscr{T}}(\tau(RN), \tau(R)))^{\dagger\mathcal{J}}$. Since $\mathcal{DLR}^\pm$ allows only for knowledge bases with a projection signature graph being a multitree, then,

$$\text{PATH}_{\mathscr{T}}(\tau(RN), \{U_i\})^\dagger = \text{PATH}_{\mathscr{T}}(\tau(RN), \tau(R))^\dagger \circ \text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger.$$

Thus, $(\imath(t_l), d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger)^{\mathcal{J}}$ and, since $\imath$ is injective, then, $\imath(t_l) \neq \imath(t_j)$ when $l \neq j$. Thus, $d \in (\exists^{\geqslant q}[U_i]R)^{\dagger\mathcal{J}}$.
Let $t \in (\sigma_{U_i:C}R)^{\mathcal{I}}$. Then, $t \in R^{\mathcal{I}}$ and $t[U_i] = d \in C^{\mathcal{I}}$. By induction, $\imath(t) \in R^{\dagger\mathcal{J}}$ and $d \in C^{\dagger\mathcal{J}}$. As before, by (13), (14) and (16), we can show that $(\imath(t), d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger)^{\mathcal{J}}$ and, since $\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger$ is functional, then $\imath(t) \in (\sigma_{U_i:C}R)^{\dagger\mathcal{J}}$.
Let $t \in (\exists[U_1, \ldots, U_k]R)^{\mathcal{I}}$. Then, there is a tuple $t' \in R^{\mathcal{I}}$ s.t. $t'[U_1, \ldots, U_k] = t$ and, by induction, $\imath(t') \in R^{\dagger\mathcal{J}}$. As before, by (14) and (16), we can show that $(\imath(t'), \imath(t)) \in \text{PATH}_{\mathscr{T}}(\tau(R), \{U_1, \ldots, U_k\})^{\dagger\mathcal{J}}$ and thus $\imath(t) \in (\exists[U_1, \ldots, U_k]R)^{\dagger\mathcal{J}}$.
All the other cases can be proved in a similar way. We now show the vice versa.

Let $d \in (\bigodot RN)^{\dagger\mathcal{J}}$. Then, $d \in (A_{RN}^l)^{\mathcal{J}}$ and, by $\gamma_{lobj}(RN)$, there is a $d' \in \Delta^{\mathcal{J}}$ s.t. $(d', d) \in Q_{RN}^{\mathcal{J}}$ and $d' \in A_{RN}^{\mathcal{J}}$. By induction, $d' = \imath(t')$ with $t' \in RN^{\mathcal{I}}$ and thus, $(\imath(t'), d) \in Q_{RN}^{\mathcal{J}}$ and, by (15), $\ell_{RN}(t') = d$, i.e., $d \in (\bigodot RN)^{\mathcal{I}}$.
Let $d \in (\exists^{\geqslant q}[U_i]R)^{\dagger\mathcal{J}}$. Thus, there are different $d_1, \ldots, d_q \in \Delta^{\mathcal{J}}$ s.t. $(d_l, d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^\dagger)^{\mathcal{J}}$ and $d_l \in R^{\dagger\mathcal{J}}$, for $l = 1, \ldots, q$. By induction, each $d_l = \imath(t_l)$ and $t_l \in R^{\mathcal{I}}$. Since $\imath$ is injective, then $t_l \neq t_j$ for all $l, j = 1, \ldots, q$, $l \neq j$. We need to show that $t_l[U_i] = d$, for all $l = 1, \ldots, q$. By (16), there is a $t'_l \in RN^{\mathcal{I}}$ s.t. $t_l = t'_l[\tau(R)]$, for some $RN \in \mathcal{R}$ and, by (14), $(\imath(t'_l), \imath(t_l)) \in$

$(\text{PATH}_{\mathscr{T}}(\tau(RN), \tau(R))^{\dagger})^{\mathcal{J}}$. Since $(\imath(t_l), d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^{\dagger})^{\mathcal{J}}$ and $\text{PATH}_{\mathscr{T}}$ is functional in $\mathcal{DLR}^{\pm}$, then, $(\imath(t'_l), d) \in (\text{PATH}_{\mathscr{T}}(\tau(RN), \{U_i\})^{\dagger})^{\mathcal{J}}$ and, by (13), $t'_l[U_i] = t_l[U_i] = d$.
Let $\imath(t) \in (\sigma_{U_i:C} R)^{\dagger\mathcal{J}}$. Thus, $\imath(t) \in R^{\dagger\mathcal{J}}$ and, by induction, $t \in R^{\mathcal{I}}$. Let $t[U_i] = d$. We need to show that $d \in C^{\mathcal{I}}$. As before, by (16) and (14), we have that $(\imath(t), d) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_i\})^{\dagger})^{\mathcal{J}}$. Then $d \in C^{\dagger\mathcal{J}}$ and, by induction, $d \in C^{\mathcal{I}}$.
Let $\imath(t) \in (\exists[U_1, \ldots, U_k]R)^{\dagger\mathcal{J}}$. Then, there is $d \in \Delta^{\mathcal{J}}$ s.t.

$$(d, \imath(t)) \in (\text{PATH}_{\mathscr{T}}(\tau(R), \{U_1, \ldots, U_k\})^{\dagger})^{\mathcal{J}}$$

and $d \in R^{\dagger\mathcal{J}}$. By induction, $d = \imath(t')$ and $t' \in R^{\mathcal{I}}$. As before, by (14) and (16), we can show that there is a tuple $t'' \in RN$ s.t. $(\imath(t''), \imath(t)) \in (\text{PATH}_{\mathscr{T}}(\tau(RN), \{U_1, \ldots, U_k\})^{\dagger})^{\mathcal{J}}$ and thus, $t = t'[U_1, \ldots, U_k]$, i.e., $t \in (\exists[U_1, \ldots, U_k]R)^{\mathcal{I}}$.
□

As a direct consequence of the above theorem and the fact that $\mathcal{DLR}$ is a sublanguage of $\mathcal{DLR}^{\pm}$, we have that

Corollary 5. *Reasoning in $\mathcal{DLR}^{\pm}$ is an* ExpTime*-complete problem.*

6 Implementation of a $\mathcal{DLR}^{\pm}$ API

7 Conclusions

8 Acknowledgements

We thank Alessandro Mosca for working with us on all the preliminary work necessary to understand how to get these technical results.

References

[Abiteboul *et al.*, 1995] Serge Abiteboul, Richard Hull, and Victor Vianu. *Foundations of Databases.* Addison-Wesley, 1995.

[Artale *et al.*, 2007] A. Artale, D. Calvanese, R. Kontchakov, V. Ryzhikov, and M. Zakharyaschev. Reasoning over extended ER models. In *Proc. of the 26th Int. Conf. on Conceptual Modeling (ER'07)*, volume 4801 of *Lecture Notes in Computer Science*, pages 277–292. Springer, 2007.

[Baader *et al.*, 2003] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications.* Cambridge University Press, 2003.

[Berardi *et al.*, 2005] D. Berardi, D. Calvanese, and G. De Giacomo. Reasoning on UML class diagrams. *Artificial Intelligence*, 168(1–2):70–118, 2005.

[Calvanese *et al.*, 1998] D. Calvanese, G. De Giacomo, and M. Lenzerini. On the decidability of query containment under constraints. In *Proc. of the 17th ACM Sym. on Principles of Database Systems (PODS'98)*, pages 149–158, 1998.

[Calvanese *et al.*, 2001] Diego Calvanese, Giuseppe De Giacomo, and Maurizio Lenzerini. Identification constraints and functional dependencies in description logics. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI-01*, pages 155–160. Morgan Kaufmann, 2001.

[Chandra and Vardi, 1985] Ashok K. Chandra and Moshe Y. Vardi. The implication problem for functional and inclusion dependencies is undecidable. *SIAM Journal on Compututing*, 14(3):671–677, 1985.

[Franconi and Mosca, 2013] Enrico Franconi and Alessandro Mosca. Towards a core ORM2 language (research note). In *OTM Workshops*, volume 8186 of *Lecture Notes in Computer Science*, pages 448–456. Springer, 2013.

[Halpin and Morgan, 2008] Terry Halpin and Tony Morgan. *Information Modeling and Relational Databases*. Morgan Kaufmann, 2nd edition, 2008.

[Horrocks *et al.*, 2000] Ian Horrocks, Ulrike Sattler, Sergio Tessaris, and Stephan Tobies. How to decide query containment under constraints using a description logic. In *7th International Conference on Logic for Programming and Automated Reasoning (LPAR00), 2000*, pages 326–343, 2000.

[Toman and Weddell, 2009] David Toman and Grant E. Weddell. Applications and extensions of PTIME description logics with functional constraints. In *IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence*, pages 948–954, 2009.