# Computing Role-depth Bounded Generalizations in the Description Logic $\mathcal{ELOR}$

Andreas Ecke[1]*, Rafael Peñaloza[1,2]**, and Anni-Yasmin Turhan[1]***

[1] Institute for Theoretical Computer Science,
Technische Universität Dresden
[2] Center for Advancing Electronics Dresden
{ecke,penaloza,turhan}@tcs.inf.tu-dresden.de

**Abstract.** Description Logics (DLs) are a family of knowledge representation formalisms, that provides the theoretical basis for the standard web ontology language OWL. Generalization services like the least common subsumer (lcs) and the most specific concept (msc) are the basis of several ontology design methods, and form the core of similarity measures. For the DL $\mathcal{ELOR}$, which covers most of the OWL 2 EL profile, the lcs and msc need not exist in general, but they always exist if restricted to a given role-depth. We present algorithms that compute these role-depth bounded generalizations. Our method is easy to implement, as it is based on the polynomial-time completion algorithm for $\mathcal{ELOR}$.

## 1 Introduction

Description logics (DLs) are knowledge representation formalisms with formal and well-understood semantics [4]. They supply the foundation for the web ontology language OWL 2 standardized by the W3C [20]. Since then, DLs became more widely used for the representation of knowledge from several domains.

Each DL offers a set of concept constructors by which complex concepts can be built. These concepts describe categories from the application domain at hand. A DL knowledge base consists of two parts: the TBox captures the terminological knowledge about categories and relations, and the ABox captures the assertional knowledge, i.e., individual facts, from the application domain. Prominent inferences are *subsumption*, which determines subconcept relationships and *instance checking*, which tests for a given individual and concept whether the individual belongs to the concept.

The lightweight DL $\mathcal{EL}$ offers limited expressivity but allows for polynomial time reasoning [3]. These good computational properties are maintained by several extensions of $\mathcal{EL}$—most prominently by $\mathcal{EL}^{++}$, the DL underlying the OWL 2 EL profile [15], which allows for the use of nominals, i.e., singleton concepts, when building complex concept descriptions. The reasoning algorithms

* Supported by DFG Graduiertenkolleg 1763 (QuantLA).
** Partially supported by DFG within the Cluster of Excellence 'cfAED'
*** Partially supported by the German Research Foundation (DFG) in the Collaborative Research Center 912 "Highly Adaptive Energy-Efficient Computing".

for (fragments of) $\mathcal{EL}^{++}$ have been implemented in highly optimized reasoner systems such as jCEL [14] and ELK [10]. It is worth pointing out that the initial reasoning algorithm for handling nominals in $\mathcal{EL}$ [3] turned out to be incomplete, but a complete method has been recently devised in [11].

In this paper we describe methods for computing generalizations in the $\mathcal{EL}$-family with the help of the standard reasoning algorithms. We consider the following two inferences: The *least common subsumer* (lcs), which computes for a given set of concepts a new concept that subsumes the input concepts and is the least one w.r.t. subsumption; and the *most specific concept* which provides a concept which has a given individual as an instance and is the least one w.r.t. subsumption. Both inferences have been employed for several applications. Most prominently the lcs and the msc can be employed in the 'bottom-up approach' for generating TBoxes, where modellers can generate a new concept from picking ABox individuals that instantiate the desired concept and then generalizing this set into a single concept automatically–first by applying the msc to each of the individuals and then generalizing the obtained concepts by applying the lcs [5]. Other applications of the lcs and the msc include similarity measures [8, 6, 13], which are the core of ontology matching algorithms and more (see [7, 16]). In particular for large bio-medical ontologies the lcs can be used effectively to support construction and maintenance. Many of these bio-medical ontologies, notably SNOMED CT [19] and the FMA Ontology [18], are written in the $\mathcal{EL}$-family of lightweight DLs.

It is known that for concepts captured in a general TBox or even just a cyclic TBox, the lcs does not need to exist [1], since cycles cannot be captured in an $\mathcal{EL}$-concept. Therefore, an approximation has been introduced in [16], that limits the maximal nesting of quantifiers of the resulting concept descriptions. These so-called role-depth bounded lcs ($k$-lcs), can be computed for $\mathcal{EL}$ and for $\mathcal{EL}$ extended by role inclusions using completion sets produced by the subsumption algorithm [16, 9]. In this paper, we describe a subsumption algorithm for the DL $\mathcal{ELOR}$—building on the one for $\mathcal{ELO}$ ($\mathcal{EL}$ extended by nominals) from [11]. Our algorithm is given in terms of the completion algorithm in order to extend the methods for the $k$-lcs to $\mathcal{ELOR}$.

Recently, necessary and sufficient conditions for the existence of the lcs w.r.t. general $\mathcal{EL}$-TBoxes have been devised [21]. By the use of these conditions the bound $k$ for which the role-depth bounded lcs and the lcs coincide can be determined, if the lcs exists; i.e., if such $k$ is finite.

Similarly to the lcs, the msc does not need to exist, if the ABox [12] contain cycles. To obtain an approximative solution, the role-depth of the resulting concept can be limited as suggested in [12]. A computation algorithm for the role-depth bounded msc has been proposed in [17] for $\mathcal{EL}$. If nominals are allowed, the computation of the msc is trivial, since the msc of an individual $a$ is simply the nominal that contains $a$ (i.e., $\{a\}$). Thus, we consider the computation of the role-depth bounded msc in $\mathcal{EL}$ w.r.t. an $\mathcal{ELOR}$ knowledge base.

We introduce the basic notions of DL and the reasoning services considered in the next section. In Section 3 we give a completion-based classification algo-

| | Syntax | Semantics |
|---|---|---|
| concept name | $A \ \ (A \in N_C)$ | $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ |
| top concept | $\top$ | $\Delta^{\mathcal{I}}$ |
| nominal | $\{a\} \ \ (a \in N_I)$ | $\{a\}^{\mathcal{I}} = \{a^{\mathcal{I}}\}$ |
| conjunction | $C \sqcap D$ | $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$ |
| existential restriction | $\exists r.C \ \ (r \in N_R)$ | $(\exists r.C)^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid \exists e.(d,e) \in r^{\mathcal{I}} \wedge e \in C^{\mathcal{I}}\}$ |
| GCI | $C \sqsubseteq D$ | $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ |
| RIA | $r_1 \circ \cdots \circ r_n \sqsubseteq s$ | $r_1^{\mathcal{I}} \circ \cdots \circ r_n^{\mathcal{I}} \subseteq s^{\mathcal{I}}$ |
| Concept assertion | $C(a)$ | $a^{\mathcal{I}} \in C^{\mathcal{I}}$ |
| Role assertion | $r(a,b)$ | $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ |

**Table 1.** Concept constructors and TBox axioms for $\mathcal{ELOR}$.

rithm for $\mathcal{ELOR}$, which serves as a basis for the computation algorithms of the role-depth bounded lcs and msc presented subsequently. The paper ends with conclusions and future work.[3]

## 2 Preliminaries

*$\mathcal{ELOR}$-concepts* are built from mutually disjoint sets $N_C$ of *concept names*, $N_R$ of *role names* and $N_I$ of *individual names* using the syntax rule:

$$C, D ::= \top \mid A \mid \{a\} \mid C \sqcap D \mid \exists r.C,$$

where $A \in N_C$, $r \in N_R$ and $a \in N_I$. The individuals appearing in concepts are also called *nominals*. The sub-logic of $\mathcal{ELOR}$ that does not allow for individuals in concepts is called $\mathcal{ELR}$.

As usual, the semantics of $\mathcal{ELOR}$-concepts is defined through interpretations. An *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of an *interpretation domain* $\Delta^{\mathcal{I}}$ and an *interpretation function* $\cdot^{\mathcal{I}}$ that maps concept names $A$ to subsets $A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$ and role names to binary relations on the domain $\Delta^{\mathcal{I}}$. This function is extended to complex concepts as shown in the upper part of Table 1.

Concepts can be used to model notions from the application domain in the TBox. Given two concepts $C$ and $D$, a *general concept inclusion axiom* (GCI) is of the form $C \sqsubseteq D$. We use $C \equiv D$ as an abbreviation for $C \sqsubseteq D$ and $D \sqsubseteq C$. Given the roles $r_1, \ldots, r_n$ and $s$, a *role inclusion axiom* (RIA) is an expression of the form $r_1 \circ \cdots \circ r_n \sqsubseteq s$. An $\mathcal{ELOR}$-TBox is a finite set of GCIs and RIAs. An interpretation is a *model* for a TBox $\mathcal{T}$ if it satisfies all GCIs and RIAs in $\mathcal{T}$, as shown in the middle part of Table 1. An $\mathcal{EL}$-TBox is an $\mathcal{ELR}$-TBox (i.e., without the nominal constructor) that does not contain any RIAs.

Knowledge about individual facts of the application domain can be captured by assertions. Let $a, b \in N_I$, $r \in N_R$ and $C$ a concept, then $C(a)$ is a *concept*

[3] Because of space constraints, some proofs are deferred to the appendix of long version of this paper at `http://lat.inf.tu-dresden.de/research/papers.html`.

*assertion* and $r(a, b)$ a *role assertion*. An *ABox* $\mathcal{A}$ is a finite set of (concept or role) assertions. An interpretation is a *model* for an ABox $\mathcal{A}$ if it satisfies all concept and role assertions in $\mathcal{A}$, as shown in the lower part of Table 1.

A *knowledge base* (KB) $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$. An interpretation is a model of $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ if it is a *model* of both $\mathcal{T}$ and $\mathcal{A}$. With $\mathsf{Sig}(\mathcal{T})$ we denote the *signature* of a TBox $\mathcal{T}$, i.e. the set of all concept names, role names, and individual names that appear in $\mathcal{T}$. By $\mathsf{Sig}(\mathcal{A})$ and $\mathsf{Sig}(\mathcal{K})$ we denote the analogous notions for ABoxes and KBs, respectively.

Important reasoning tasks considered for DLs are *subsumption* and *instance checking*. A concept $C$ *is subsumed by* a concept $D$ w.r.t. a TBox $\mathcal{T}$ (denoted $C \sqsubseteq_{\mathcal{T}} D$) if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds in all models $\mathcal{I}$ of $\mathcal{T}$. A concept $C$ is *equivalent* to a concept $D$ w.r.t. a TBox $\mathcal{T}$ (denoted $C \equiv_{\mathcal{T}} D$) if $C \sqsubseteq_{\mathcal{T}} D$ and $D \sqsubseteq_{\mathcal{T}} C$ hold. The reasoning service *classification* of a TBox $\mathcal{T}$ computes all subsumption relationships between the named concepts occurring in $\mathcal{T}$. A reasoning service dealing with a whole KB is *instance checking*. An individual $a$ *is an instance of* a given concept $C$ w.r.t. $\mathcal{K}$ (denoted $\mathcal{K} \models C(a)$) if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ holds in all models $\mathcal{I}$ of $\mathcal{K}$. *ABox realization* computes, for every concept name in $\mathcal{K}$, the set of individuals from the ABox that belong to that concept. These reasoning problems can all be decided for $\mathcal{ELOR}$, and hence also in $\mathcal{EL}$, in polynomial time [3].

There are two central inferences discussed in this paper that compute generalizations. The first is called the *least common subsumer* (lcs); it computes, for two given concepts, a (possibly complex) concept that subsumes both input concepts and that is the least concept with this property w.r.t. subsumption. The second is called the *most specific concept* (msc), which computes for a given individual $a$ the least concept w.r.t. subsumption that has $a$ as an instance w.r.t. $\mathcal{K}$.

The lcs does not need to exist if computed w.r.t. general $\mathcal{EL}$-TBoxes, i.e., TBoxes that use complex concepts in the left-hand sides of GCIs, or even just cyclic TBoxes [2]. The reason is that the resulting concept cannot capture cycles. Thus, we follow here the idea from [16] and compute only approximations of the lcs and of the msc by limiting the nesting of quantifiers of the resulting concept.

The *role depth* $rd(C)$ of a concept $C$ denotes the maximal nesting depth of the existential quantifier in $C$. Sometimes it is convenient to write the resulting concept in a different DL than the one the inputs concepts are written in. Thus we distinguish a 'source DL' $\mathcal{L}_s$ and a 'target DL' $\mathcal{L}_t$. With these notions at hand, we can define the first generalization inference.

**Definition 1 (lcs, role-depth bounded lcs).** *The* least common subsumer *of two $\mathcal{L}_s$-concepts $C_1, C_2$ w.r.t. an $\mathcal{L}_s$-TBox $\mathcal{T}$ (written: $lcs_{\mathcal{T}}(C_1, C_2)$) is the $\mathcal{L}_t$-concept description $D$ s.t.:*

1. *$C_1 \sqsubseteq_{\mathcal{T}} D$ and $C_2 \sqsubseteq_{\mathcal{T}} D$, and*
2. *for all $\mathcal{L}_t$-concepts $E$, $C_1 \sqsubseteq_{\mathcal{T}} E$ and $C_2 \sqsubseteq_{\mathcal{T}} E$ implies $D \sqsubseteq_{\mathcal{T}} E$.*

*Let $k \in \mathbb{N}$. If the concept $D$ has a role-depth up to $k$ and Condition 2 holds for all such $E$ with role-depth up to $k$, then $D$ is the* role-depth bounded lcs *($k$-$lcs_{\mathcal{T}}(C_1, C_2)$) of $C_1$ and $C_2$ w.r.t. $\mathcal{T}$ and $k$.*

The role-depth bounded lcs is unique up to equivalence, thus we speak of *the* $k$-*lcs*. In contrast, common subsumers need not be unique. Note that for target DLs that offer disjunction, the lcs is always trivial: $lcs(C_1, C_2) = C_1 \sqcup C_2$. Thus target DLs without disjunction may yield more informative lcs.

Similarly to the lcs, the msc does not need to exist if computed w.r.t. cyclic ABoxes. Again we compute here approximations of the msc by limiting the role-depth of the resulting concept as suggested in [12].

**Definition 2.** *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a KB written in $\mathcal{L}_s$ and $a$ be an individual from $\mathcal{A}$. An $\mathcal{L}_t$-concept description $C$ is the* most specific concept of $a$ w.r.t. $\mathcal{K}$ *(written $msc_{\mathcal{K}}(a)$) if it satisfies:*

1. *$\mathcal{K} \models C(a)$, and*
2. *for all $\mathcal{L}_t$-concepts $D$, $\mathcal{K} \models D(a)$ implies $C \sqsubseteq_{\mathcal{T}} D$.*

*If the concept $C$ has a role-depth up to $k$ and Condition 2 holds for all such $D$ with role-depth up to $k$, then $C$ is the* role depth bounded msc *of $a$ w.r.t. $\mathcal{K}$ and $k$ ($k$-$msc_{\mathcal{K}}(a)$).*

The msc and the $k$-*msc* are unique up to equivalence in $\mathcal{EL}$ and $\mathcal{ELOR}$. In $\mathcal{ELOR}$ the msc is trivial, since $\text{msc}_{\mathcal{K}}(a) = \{a\}$. Thus we consider in this paper a more interesting case, where the target DL $\mathcal{L}_t$ for the resulting concept is a less expressive one without nominals, namely $\mathcal{EL}$ or $\mathcal{ELR}$.

# 3 The $k$-lcs in $\mathcal{ELOR}$

The algorithms to compute the role-depth bounded lcs are based on completion-based classification algorithms for the corresponding DL. For the DL $\mathcal{ELOR}$, a consequence-based algorithm for classification of TBoxes was presented in [11], building upon the completion algorithm developed in [3]. The completion algorithm presented next adapts the ideas of the complete algorithm.

## 3.1 Completion Algorithm for $\mathcal{ELOR}$-TBoxes

The completion algorithms work on normalized TBoxes. We define for $\mathcal{ELOR}$ the set of *basic concepts* for a TBox $\mathcal{T}$:

$$\mathsf{BC}_{\mathcal{T}} = (\mathsf{Sig}(\mathcal{T}) \cap (N_C \cup N_I)) \cup \{\top\}.$$

Let $\mathcal{T}$ be an $\mathcal{ELOR}$-TBox and $A, A_1, A_2, B \in \mathsf{BC}_{\mathcal{T}}$; then $\mathcal{T}$ is in *normal form* if

- each GCI in $\mathcal{T}$ is of the form: $A \sqsubseteq B$, $A_1 \sqcap A_2 \sqsubseteq B$, $A \sqsubseteq \exists r.B$, or $\exists r.A \sqsubseteq B$.
- each RIA in $\mathcal{T}$ is of the form: $r \sqsubseteq s$ or $r_1 \circ r_2 \sqsubseteq s$.

Every $\mathcal{ELOR}$-TBox can be transformed into normal form in linear time by applying a set of normalization rules given in [3]. These normalization rules essentially introduce new named concepts for complex concepts used in GCIs or new roles used in complex RIAs.

Before describing the completion algorithm in detail, we introduce the reachability relation $\leadsto_R$, which plays a fundamental role in the correct treatment of nominals in TBox classification algorithms [3, 11].

**Definition 3** ($\leadsto_R$)**.** *Let $\mathcal{T}$ be an $\mathcal{ELOR}$-TBox in normal form, $G \in N_C$ a concept name, and $D \in \mathsf{BC}_\mathcal{T}$. $G \leadsto_R D$ iff there exist roles $r_1, \dots, r_n \in N_R$ and basic concepts $A_0, \dots, A_n, B_0, \dots, B_n \in \mathsf{BC}_\mathcal{T}$, $n \geq 0$ such that $A_i \sqsubseteq_\mathcal{T} B_i$ for all $0 \leq i \leq n$, $B_{i-1} \sqsubseteq \exists r_i.A_i \in \mathcal{T}$ for all $1 \leq i \leq n$, $A_0$ is either $G$ or a nominal, and $B_n = D$.*

Informally, the concept name $D$ is reachable from $G$ if there is a chain of existential restrictions starting from $G$ or a nominal and ending in $D$. This implies that, for $G \leadsto_R D$, if the interpretation of $G$ is not empty, then the interpretation of $D$ cannot be empty either. This in turn causes additional subsumption relationships to hold. Note that, if $D$ is reachable from a nominal, then $G \leadsto_R D$ holds for all concept names $G$, since the interpretation of $D$ can never be empty.

The basic idea of completion algorithms in general is to generate canonical models of the TBox. To this end, the elements of the interpretation domain are represented by named concepts or nominals from the normalized TBox. These elements are then related via roles according to the existential restrictions derived for the TBox. More precisely, let $\mathcal{T}$ be a normalized TBox, $G \in \mathsf{Sig}(\mathcal{T}) \cap N_C \cup \{\top\}$ and $A \in \mathsf{BC}_\mathcal{T}$, we introduce a *completion set* $S^G(A)$. We store all basic concepts that subsume a basic concept $A$ in the completion set $S^A(A)$ and all basic concepts $B$ for which $\exists r.B$ subsumes $A$ in the completion set $S^A(A, r)$. These completion sets are then extended using a set of rules. However, the algorithm needs to keep track also of completion sets of the form $S^G(A)$ and $S^G(A, r)$ for every $G \in (\mathsf{Sig}(\mathcal{T}) \cap N_C) \cup \{\top\}$, since the non-emptiness of an interpretation of a concept $G$ may imply additional subsumption relationships for $A$. The completion set $S^G(A)$ therefore stores all basic concepts that subsume $A$ under the assumption that $G$ is not empty. Similarly $S^G(A, r)$ stores all concepts $B$ for which $\exists r.B$ subsumes $A$ under the same assumption.

For every $G \in (\mathsf{Sig}(\mathcal{T}) \cap N_C) \cup \{\top\}$, every basic concept $A$ and every role name $r$, the completion sets are initialized as $S^G(A) = \{A, \top\}$ and $S^G(A, r) = \emptyset$. These sets are then extended by applying the completion rules shown in Figure 1 (adapted from [11]) exhaustively.

To compute the reachability relation $\leadsto_R$ used in rule **OR7**, the algorithm can use Definition 3 with all previously derived subsumption relationships; that is, $A_i \sqsubseteq B_i$ if it finds $B_i \in S^{A_i}(A_i)$. Thus the computation of $\leadsto_R$ and the application of the completion rules need to be carried out simultaneously.

It can be shown that the algorithm terminates in polynomial time, and is sound and complete for classifying the TBox $\mathcal{T}$. In particular, when no rules are applicable anymore the completion sets have the following properties.

**Proposition 1.** *Let $\mathcal{T}$ be an $\mathcal{ELOR}$-TBox in normal form, $C, D \in \mathsf{BC}_\mathcal{T}$, $r \in \mathsf{Sig}(\mathcal{T}) \cap N_R$, and $G = C$ if $C \in N_C$ and $G = \top$ otherwise. Then, the following properties hold:*

$C \sqsubseteq_\mathcal{T} D$ *iff* $D \in S^G(C)$, *and*
$C \sqsubseteq_\mathcal{T} \exists r.D$ *iff there exists* $E \in \mathsf{BC}_\mathcal{T}$ *such that* $E \in S^G(C, r)$ *and* $D \in S^G(E)$.

We now show how to use these completion sets for computing the role-depth bounded lcs for $\mathcal{ELOR}$-concept w.r.t. a general $\mathcal{ELOR}$-TBox.

**OR1** If $A_1 \in S^G(A)$, $A_1 \sqsubseteq B \in \mathcal{T}$ and $B \notin S^G(A)$,
then $S^G(A) := S^G(A) \cup \{B\}$

**OR2** If $A_1, A_2 \in S^G(A)$, $A_1 \sqcap A_2 \sqsubseteq B \in \mathcal{T}$ and $B \notin S^G(A)$,
then $S^G(A) := S^G(A) \cup \{B\}$

**OR3** If $A_1 \in S^G(A)$, $A_1 \sqsubseteq \exists r.B \in \mathcal{T}$ and $B \notin S^G(A, r)$,
then $S^G(A, r) := S^G(A, r) \cup \{B\}$

**OR4** If $B \in S^G(A, r)$, $B_1 \in S^G(B)$, $\exists r.B_1 \sqsubseteq C \in \mathcal{T}$ and $C \notin S^G(A)$,
then $S^G(A) := S^G(A) \cup \{C\}$

**OR5** If $B \in S^G(A, r)$, $r \sqsubseteq s \in \mathcal{T}$ and $B \notin S^G(A, s)$,
then $S^G(A, s) := S^G(A, s) \cup \{B\}$

**OR6** If $B \in S^G(A, r_1), C \in S^G(B, r_2)$, $r_1 \circ r_2 \sqsubseteq s \in \mathcal{T}$ and $C \notin S^G(A, s)$,
then $S^G(A, s) := S^G(A, s) \cup \{C\}$

**OR7** If $\{a\} \in S^G(A_1) \cap S^G(A_2)$, $G \leadsto_R A_2$, and $A_2 \notin S^G(A_1)$,
then $S^G(A_1) := S^G(A_1) \cup \{A_2\}$

**Fig. 1.** Completion rules for $\mathcal{ELOR}$

### 3.2 Computing the Role-depth Bounded $\mathcal{ELOR}$-lcs

In order to compute the role-depth bounded lcs of two $\mathcal{ELOR}$-concepts $C$ and $D$, we extend the methods from [16] for $\mathcal{EL}$-concepts and from [9] for $\mathcal{ELR}$-concepts, where we compute the cross-product of the tree unravelings of the canonical model represented by the completion sets for $C$ and $D$ up to the role-depth $k$. Clearly, in the presence of nominals, the right completion sets need to be chosen that preserve the non-emptiness of the interpretation of concepts derived by $\leadsto_R$.

An algorithm that computes the role-depth bounded $\mathcal{ELOR}$-lcs using completion sets is shown in Figure 2. In the first step, the algorithm introduces two new concept names $A$ and $B$ as abbreviations for the (possibly complex) concepts $C$ and $D$, and the augmented TBox is normalized. The completion sets are then initialized and the completion rules from Figure 1 are applied exhaustively, yielding the saturated completion sets $\mathsf{S}_\mathcal{T}$. In the recursive procedure k-lcs-r for concepts $A$ and $B$, we first obtain all the basic concepts that subsume both $A$ and $B$ from the sets $S^A(A)$ and $S^B(B)$. For every role name $r$, the algorithm then recursively computes the $(k-1)$-lcs of the concepts $A'$ and $B'$ in the subsumer sets $S^A(A, r)$ and $S^B(B, r)$, i.e. for which $A \sqsubseteq_\mathcal{T} \exists r.A'$ and $B \sqsubseteq_\mathcal{T} \exists r.B'$. These concepts are added as existential restrictions to the $k$-lcs.

The algorithm only introduces concept and role names that occur in the original TBox $\mathcal{T}$. Therefore those names introduced by the normalization are not used in the concept for the $k$-lcs and an extra denormalization step as in [16, 9] is not necessary.

Notice that for every pair $(A', B')$ of $r$-successors of $A$ and $B$ it holds that $A \leadsto_R A'$ and $B \leadsto_R B'$. Intuitively, we are assuming that the interpretation of both $A$ and $B$ is not empty. This in turn causes the interpretation of $\exists r.A'$ and $\exists r.B'$ to be not empty, either. Thus, it suffices to consider the completion

**Procedure** k-lcs$(C, D, \mathcal{T}, k)$
**Input:** $C, D$: $\mathcal{ELOR}$-concepts; $\mathcal{T}$: $\mathcal{ELOR}$-TBox; $k \in \mathbb{N}$
**Output:** role-depth bounded $\mathcal{ELOR}$-lcs of $C, D$ w.r.t. $\mathcal{T}$ and $k$

$\mathcal{T}' := \mathsf{normalize}(\mathcal{T} \cup \{A \equiv C, B \equiv D\})$
$\mathsf{S}_{\mathcal{T}} := \mathsf{apply\text{-}completion\text{-}rules}(\mathcal{T}')$
**return** $\mathsf{k\text{-}lcs\text{-}r}(A, B, \mathsf{S}_{\mathcal{T}}, k, A, B, \mathsf{Sig}(\mathcal{T}))$

**Procedure** k-lcs-r$(X, Y, \mathsf{S}_{\mathcal{T}}, k, A, B, \mathsf{Sig}(\mathcal{T}))$
**Input:** $A, B$: concept names, $X, Y$: basic concepts with $A \leadsto_R X, B \leadsto_R Y$; $k \in \mathbb{N}$;
$\mathsf{S}_{\mathcal{T}}$: set of saturated completion sets; $\mathsf{Sig}(\mathcal{T})$: signature of $\mathcal{T}$
**Output:** role-depth bounded $\mathcal{ELOR}$-lcs of $X, Y$ w.r.t. $\mathcal{T}$ and $k$

common-names $:= S^A(X) \cap S^B(Y) \cap \mathsf{BC}_{\mathcal{T}}$
**if** $k = 0$ **then**
  **return** $\displaystyle\bigsqcap_{P \in \text{common-names}} P$
**else**
  **return** $\displaystyle\bigsqcap_{P \in \text{common-names}} P \sqcap \bigsqcap_{r \in \mathsf{Sig}(\mathcal{T}) \cap N_R} \Bigg( \bigsqcap_{\substack{C \in S^A(X,r),\\ D \in S^B(Y,r)}} \exists r.\mathsf{k\text{-}lcs\text{-}r}\big(C, D, \mathsf{S}_{\mathcal{T}}, k-1, A, B, \mathsf{Sig}(\mathcal{T})\big)\Bigg)$

**Fig. 2.** Computation algorithm for role-depth bounded $\mathcal{ELOR}$-lcs.

sets $S^A$ and $S^B$, without the need to additionally compute $S^{A'}$ and $S^{B'}$, or the completion sets $S^C$ for any other basic concept $C$ encountered during the recursive computation of the $k$-lcs. This allows for a goal-oriented optimization in cases where there is no need to classify the full TBox.

### 3.3 Computing the Role-depth Bounded msc w.r.t. $\mathcal{ELOR}$-KBs

We now turn our attention to the other generalization inference: the computation of the most specific concept representing a given individual. Recall that, since $\mathcal{ELOR}$ allows the use of nominals, computing the (exact) $\mathcal{ELOR}$-msc for a given individual is a trivial task: the most specific $\mathcal{ELOR}$-concept describing an individual $a \in N_I$ is simply the nominal $\{a\}$. However, it may be of interest to compute the msc w.r.t. a less expressive target DL. Next, we describe how to compute the depth-bounded $\mathcal{EL}$-msc of an individual w.r.t. an $\mathcal{ELOR}$-KB.

As we have defined them, KBs consist of two parts: the TBox, which represents the conceptual knowledge of the domain, and the ABox, which states information about individuals. In the presence of nominals, this division between concepts and individuals is blurred. In fact, it is possible to simulate ABox assertions using GCIs as described by the following proposition.

**Lemma 1.** *An interpretation $\mathcal{I}$ satisfies the concept assertion $C(a)$ iff it satisfies the GCI $\{a\} \sqsubseteq C$. It satisfies the role assertion $r(a, b)$ iff it satisfies the GCI $\{a\} \sqsubseteq \exists r.\{b\}$.*

**Procedure** k-msc $(a, \mathcal{K}, k)$
**Input:** $a$: individual from $\mathcal{K}$; $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ an $\mathcal{ELOR}$-KB; $k \in \mathbb{N}$
**Output:** role-depth bounded $\mathcal{EL}$-msc of $a$ w.r.t. $\mathcal{K}$ and $k$.

$\mathcal{T}' := \mathcal{T} \cup \mathsf{absorb\text{-}ABox}(\mathcal{K})$
$\mathcal{T}'' := \mathsf{normalize}(\mathcal{T}')$
$\mathsf{S}_{\mathcal{K}} := \mathsf{apply\text{-}completion\text{-}rules}(\mathcal{T}'')$
**return** $\mathsf{traversal\text{-}concept}\,(\{a\}, \mathsf{S}_{\mathcal{K}}, k, \mathsf{Sig}(\mathcal{K}))$

**Procedure** traversal-concept $(A, \mathsf{S}_{\mathcal{K}}, k, \mathsf{Sig}(\mathcal{K}))$
**Input:** $A$: basic concept from $\mathcal{T}'$; $\mathsf{S}_{\mathcal{K}}$: set of completion sets; $k \in \mathbb{N}$;
$\mathsf{Sig}(\mathcal{K})$: signature of original KB $\mathcal{K}$
**Output:** role-depth bounded traversal concept w.r.t. $\mathcal{K}$ and $k$.

**if** $k = 0$ **then**
  **return** $\bigsqcap_{B \in S^{\top}(A) \cap (\mathsf{BC}_{\mathcal{T}} \setminus N_I)} B$
**else**
  **return** $\displaystyle\bigsqcap_{B \in S^{\top}(A) \cap (\mathsf{BC}_{\mathcal{T}} \setminus N_I)} B \ \sqcap \bigsqcap_{r \in \mathsf{Sig}(\mathcal{K}) \cap N_R} \ \bigsqcap_{B \in S^{\top}(A,r)} \exists r.\mathsf{traversal\text{-}concept}\,(B, \mathsf{S}_{\mathcal{K}}, k-1, \mathsf{Sig}(\mathcal{K}))$

**Fig. 3.** Computation algorithm for the role-depth bounded $\mathcal{EL}$-msc w.r.t. $\mathcal{ELOR}$-KBs.

Using this result, we can 'absorb' the ABox into the TBox and restrict our attention to reasoning w.r.t. TBoxes only, without losing generality. Figure 3 describes the algorithm for computing the $\mathcal{EL}$-$k$-msc w.r.t. an $\mathcal{ELOR}$-KB.

As before, correctness of this algorithm is a consequence of the invariants described by Proposition 1. The set $S^{\top}(\{a\})$ contains all the basic concepts that subsume the nominal $\{a\}$; that is, all concepts whose interpretation must contain the individual $a^{\mathcal{I}}$. Likewise, $S^{\top}(\{a\}, r)$ contains all the existential restrictions subsuming $\{a\}$. Thus, a recursive conjunction of all these subsumers provides the most specific representation of the individual $a$.

Since the target language is $\mathcal{EL}$, no nominals may be included in the output. However, the recursion includes also the $\mathcal{EL}$-msc of the removed nominals, hence indeed providing the most specific $\mathcal{EL}$ representation of the input individual. As in the computation of the lcs presented above, the only completion sets relevant for computing the msc are those of the form $S^{\top}(A)$ and $S^{\top}(A, r)$. Once again, this means that it is possible to implement a goal-oriented approach that computes only these sets, as needed, when building the msc for a given individual.

In this section we have shown how to compute generalization inferences with a bounded role-depth in the DL $\mathcal{ELOR}$ that extends $\mathcal{EL}$ by allowing nominals and complex role inclusion axioms in the KB. With the exception of data-types and disjointness axioms, this covers the full expressivity of the OWL 2 EL profile of the standard ontology language OWL 2. Given its status as W3C standard, it is likely that more and bigger ontologies built using this profile, thus the generalization inferences investigated in this paper and their computation algorithms for approximation will become more useful to ontology engineers. In fact, there already exist ontologies that use nominals in their representation. For example, the FMA ontology [18] is written in $\mathcal{ELOR}$ and currently contains 85 nominals.

## 4 Conclusions

We have studied reasoning services in extensions of the light-weight description logic $\mathcal{EL}$ by nominals and role inclusions, which yields the DL $\mathcal{ELOR}$. One of the characterizing features of $\mathcal{EL}$ and its extension $\mathcal{ELOR}$ is that they allow for polynomial time reasoning. Efficient reasoning becomes expedient when dealing with huge knowledge bases such as the biomedical ontologies SNOMED and the Gene Ontology. Additionally, $\mathcal{ELOR}$ covers a large part of the OWL 2 EL profile. Given its status as a W3C recommendation, it is likely that the usage of the $\mathcal{EL}$-family of DLs becomes more widespread in the future.

Especially for the huge ontologies written in extensions of $\mathcal{EL}$, tools that aid the user with the construction and maintenance of the knowledge base are necessary. As previous work has shown, the generalization inferences lcs and msc can be effectively used for such tasks. Besides this, the generalizations can be used as a basis for other inferences, like the construction of semantic similarity measures and information retrieval procedures.

The contributions of the paper are manyfold. First, we have given a completion algorithm for $\mathcal{ELOR}$ knowledge bases, inspired by a consequence-based classification algorithm for $\mathcal{EL}$ with nominals [11]. This completion algorithm is then employed to extend the algorithms for computing approximations of the lcs and of the msc for the DL $\mathcal{ELOR}$. In general, the lcs and msc do not need to exist, even for $\mathcal{EL}$, thus we approximate them by limiting the role-depth of the resulting concept description, up to a maximal bound specified by the user.

We extended here the computation algorithm of the $k$-lcs to the DL $\mathcal{ELOR}$, using the new completion algorithm, by allowing nominals as part of the resulting concept. Since the $k$-msc is trivial in $\mathcal{ELOR}$ due to nominals, we give a computation algorithm for the $k$-msc for the target language $\mathcal{EL}$, which works for $\mathcal{ELOR}$-KBs. Using these algorithms, the generalization inferences can be used for a large set of ontologies built for the OWL 2 EL profile. Both algorithms have the property that, if the exact lcs or msc exist, then our algorithms compute the exact solution for a sufficiently large role-depth bound $k$. Such a $k$ can be computed for $\mathcal{EL}$ using the necessary and sufficient conditions for the existence of the lsc and msc given in [21].

As future work we intend to study methods of finding these generalizations in further extensions of $\mathcal{EL}$. Initial steps in this direction have been made by considering $\mathcal{EL}$ with subjective probability constructors [17]. In a different direction, we also intend to implement a system that can compute the lcs and the msc, by modifying and improving existing completion-based reasoners.

## References


1. F. Baader. Least common subsumers and most specific concepts in a description logic with existential restrictions and terminological cycles. In G. Gottlob and T. Walsh, editors, *Proc. of the 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*, pages 325–330. Morgan Kaufmann, 2003.
2. F. Baader. Terminological cycles in a description logic with existential restrictions. In G. Gottlob and T. Walsh, editors, *Proc. of the 18th Int. Joint Conf. on Artificial Intelligence (IJCAI-03)*, pages 319–324. Morgan Kaufmann, 2003.
3. F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proc. of the 19th Int. Joint Conf. on Artificial Intelligence (IJCAI-05)*, Edinburgh, UK, 2005. Morgan-Kaufmann Publishers.
4. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications.* Cambridge University Press, 2003.
5. F. Baader, R. Küsters, and R. Molitor. Computing least common subsumers in description logics with existential restrictions. In T. Dean, editor, *Proc. of the 16th Int. Joint Conf. on Artificial Intelligence (IJCAI-99)*, pages 96–101, Stockholm, Sweden, 1999. Morgan Kaufmann, Los Altos.
6. A. Borgida, T. Walsh, and H. Hirsh. Towards measuring similarity in description logics. In *Proc. of the 2005 Description Logic Workshop (DL 2005)*, volume 147 of *CEUR Workshop*, 2005.
7. S. Brandt and A.-Y. Turhan. Using non-standard inferences in description logics — what does it buy me? In G. Görz, V. Haarslev, C. Lutz, and R. Möller, editors, *Proc. of the 2001 Applications of Description Logic Workshop (ADL 2001)*, number 44 in CEUR Workshop, Vienna, Austria, September 2001. RWTH Aachen. See `http://CEUR-WS.org/Vol-44/`.
8. C. d'Amato, N. Fanizzi, and F. Esposito. A semantic similarity measure for expressive description logics. In *Pro.c of Convegno Italiano di Logica Computazionale, CILC05*, 2005.
9. A. Ecke and A.-Y. Turhan. Role-depth bounded least common subsumers for $\mathcal{EL}^+$ and $\mathcal{ELI}$. In Y. Kazakov, D. Lembo, and F. Wolter, editors, *Proc. of the 2012 Description Logic Workshop (DL 2012)*, volume 846 of *CEUR Workshop Proceedings*. CEUR Workshop, 2012.
10. Y. Kazakov, M. Krötzsch, and F. Simančík. ELK reasoner: Architecture and evaluation. In I. Horrocks, M. Yatskevich, and E. Jimenez-Ruiz, editors, *Proceedings of the OWL Reasoner Evaluation Workshop (ORE'12)*, volume 858 of *CEUR Workshop*. CEUR-WS.org, 2012.
11. Y. Kazakov, M. Krötzsch, and F. Simančík. Practical reasoning with nominals in the $\mathcal{EL}$ family of description logics. In G. Brewka, T. Eiter, and S. A. McIlraith, editors, *Proc. of the 12th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR-12)*, pages 264–274. AAAI Press, 2012.
12. R. Küsters and R. Molitor. Approximating most specifc concepts in description logics with existential restrictions. In F. Baader, G. Brewka, and T. Eiter, editors, *Proc. of the 24th German Annual Conf. on Artificial Intelligence (KI'01)*, volume 2174 of *Lecture Notes In Artificial Intelligence*, pages 33–47. Springer, 2001.
13. K. Lehmann and A.-Y. Turhan. A framework for semantic-based similarity measures for $\mathcal{ELH}$-concepts. In L. F. del Cerro, A. Herzig, and J. Mengin, editors, *Proceedings of the 13th European Conference on Logics in Artificial Intelligence*, Lecture Notes in Artificial Intelligence, pages 307–319. Springer Verlag, 2012.

14. J. Mendez. jCel: A modular rule-based reasoner. In *In Proc. of the 1st Int. Workshop on OWL Reasoner Evaluation (ORE'12)*, volume 858 of *CEUR Workshop*, 2012.
15. B. Motik, B. Cuenca Grau, I. Horrocks, Z. Wu, A. Fokoue, and C. Lutz. OWL 2 web ontology language profiles. W3C Recommendation, 27 October 2009. `http://www.w3.org/TR/2009/REC-owl2-profiles-20091027/`.
16. R. Peñaloza and A.-Y. Turhan. A practical approach for computing generalization inferences in $\mathcal{EL}$. In M. Grobelnik and E. Simperl, editors, *Proc. of the 8th European Semantic Web Conf. (ESWC'11)*, Lecture Notes in Computer Science. Springer, 2011.
17. R. Peñaloza and A.-Y. Turhan. Instance-based non-standard inferences in $\mathcal{EL}$ with subjective probabilities. In F. Bobillo, P. C. G. Costa, C. d'Amato, N. Fanizzi, K. B. Laskey, K. J. Laskey, T. Lukasiewicz, M. Nickles, and M. Pool, editors, *Uncertainty Reasoning for the Semantic Web II*, number 7123 in Lecture Notes in Computer Science, pages 80–98. Springer, 2013.
18. C. Rosse and J. L. V. Mejino. A reference ontology for biomedical informatics: the foundational model of anatomy. *Journal of Biomedical Informatics*, 36:478–500, 2003.
19. K. Spackman. Managing clinical terminology hierarchies using algorithmic calculation of subsumption: Experience with SNOMED-RT. *Journal of the American Medical Informatics Assoc.*, 2000. Fall Symposium Special Issue.
20. W3C OWL Working Group. OWL 2 web ontology language document overview. W3C Recommendation, 27th October 2009. `http://www.w3.org/TR/2009/REC-owl2-overview-20091027/`.
21. B. Zarrieß and A.-Y. Turhan. Most specific generalizations w.r.t. general $\mathcal{EL}$-tboxes. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence (IJCAI'13)*, Beijing, China, 2013. AAAI Press. To appear.

## A Omitted Proofs

### A.1 Correctness of the classification algorithm for $\mathcal{ELOR}$

To proof the correctness of the completion algorithm for $\mathcal{ELOR}$ as given in Proposition 1, we split it in two parts: soundness and completeness.

**Lemma 2 (Soundness of the $\mathcal{ELOR}$ completion algorithm).** *Let $\mathcal{T}$ be an $\mathcal{ELOR}$-TBox in normal form, $A, B \in \mathsf{BC}_{\mathcal{T}}$, $r \in \mathsf{Sig}(\mathcal{T}) \cap N_R$. Then, the following properties hold:*

$$B \in S^G(A) \Longrightarrow G : A \sqsubseteq_{\mathcal{T}} B \tag{1}$$

$$B \in S^G(A, r) \Longrightarrow G : A \sqsubseteq_{\mathcal{T}} \exists r.B \tag{2}$$

*Proof.* We proof soundness by induction on the number of rule applications. More precisely, we show that the properties (1) and (2) hold for the initial subsumer sets and are preserved by any rule application.

- Initially, $S^G(A) = \{A, \top\}$ and $S^G(A, r) = \emptyset$. Since $G : A \sqsubseteq_{\mathcal{T}} A$ and $G : A \sqsubseteq_{\mathcal{T}} \top$ always holds, (1) and (2) are satisfied.
- Assume that rule **OR1** has been applied to $A_1 \in S^G(A)$ and $A_1 \sqsubseteq B \in \mathcal{T}$. By induction, $A_1 \in S^G(A)$ implies $G : A \sqsubseteq_{\mathcal{T}} A_1$, which then yields $G : A \sqsubseteq_{\mathcal{T}} B$. Thus, after adding $B$ to $S^G(A)$, (1) and (2) are still satisfied.
- Assume that rule **OR2** has been applied to $A_1, A_2 \in S^G(A)$ and $A_1 \sqcap A_2 \sqsubseteq B \in \mathcal{T}$. By induction, $A_1 \in S^G(A)$ implies $G : A \sqsubseteq_{\mathcal{T}} A_1$ and similarly, $A_2 \in S^G(A)$ implies $G : A \sqsubseteq_{\mathcal{T}} A_2$, which then yields $G : A \sqsubseteq_{\mathcal{T}} B$. Thus, after adding $B$ to $S^G(A)$, (1) and (2) are still satisfied.
- Assume that rule **OR3** has been applied to $A_1 \in S^G(A)$ and $A_1 \sqsubseteq \exists r.B \in \mathcal{T}$. By induction, $A_1 \in S^G(A)$ implies $G : A \sqsubseteq_{\mathcal{T}} A_1$, which then yields $G : A \sqsubseteq_{\mathcal{T}} \exists r.B$. Thus, after adding $B$ to $S^G(A, r)$, (1) and (2) are still satisfied.
- Assume that rule **OR4** has been applied to $B \in S^G(A, r)$, $B_1 \in S^G(B)$, and $\exists r.B_1 \sqsubseteq C \in \mathcal{T}$. By induction, $B \in S^G(A, r)$ implies $G : A \sqsubseteq_{\mathcal{T}} \exists r.B$ and similarly, $B_1 \in S^G(B)$ implies $G : B \sqsubseteq_{\mathcal{T}} B_1$, and hence $G : A \sqsubseteq_{\mathcal{T}} \exists r.B_1 \sqsubseteq_{\mathcal{T}} C$. Thus, after adding $C$ to $S^G(A)$, (1) and (2) are still satisfied.
- Assume that rule **OR5** has been applied to $B \in S^G(A, r)$ and $r \sqsubseteq s \in \mathcal{T}$. By induction, $B \in S^G(A, r)$ implies $G : A \sqsubseteq_{\mathcal{T}} \exists r.B$, which then yields $G : A \sqsubseteq_{\mathcal{T}} \exists s.B$. Thus, after adding $B$ to $S^G(A, s)$, (1) and (2) are still satisfied.
- Assume that rule **OR6** has been applied to $B \in S^G(A, r_1)$, $C \in S^G(B, r_2)$ and $r_1 \circ r_2 \sqsubseteq s \in \mathcal{T}$. By induction, $B \in S^G(A, r_1)$ implies $G : A \sqsubseteq_{\mathcal{T}} \exists r_1.B$ and similarly, $C \in S^G(B, r_2)$ implies $G : B \sqsubseteq_{\mathcal{T}} \exists r_2.C$, which then yield $G : A \sqsubseteq_{\mathcal{T}} \exists r_1.\exists r_2.C \sqsubseteq_{\mathcal{T}} \exists s.C$. Thus, after adding $C$ to $S^G(A, s)$, (1) and (2) are still satisfied.
- Assume that rule **OR7** has been applied to $\{a\} \in S^G(A_1) \cap S^G(A_2)$ and $G \rightsquigarrow_R A_2$. By induction, $\{a\} \in S^G(A_i)$ implies $G : A_1 \sqsubseteq_{\mathcal{T}} \{a\}$ for $i = 1, 2$. If $A_2$ has a nonempty interpretation, then $A_2 : A_2 \equiv_{\mathcal{T}} \{a\}$ and this $A_2$ is reachable from $G$, also $G : A_2 \equiv_{\mathcal{T}} \{a\}$. This yields $G : A_1 \equiv_{\mathcal{T}} A_2$. Thus, after adding $A_1$ to $S^G(A_1)$, (1) and (2) are still satisfied. □

**Lemma 3 (Completeness of the $\mathcal{ELOR}$ completion algorithm).** *Let $\mathcal{T}$ be an $\mathcal{ELOR}$-TBox in normal form, $A, B \in \mathsf{BC}_\mathcal{T}$, $r \in \mathsf{Sig}(\mathcal{T}) \cap N_R$, and $G = A$ or $G \leadsto_R A$ if $A \in N_C$ and $G \in \mathsf{BC}_\mathcal{T}$ otherwise. Then, the following properties hold:*

- *If $A \sqsubseteq_\mathcal{T} B$, then $B \in S^G(A)$, and*
- *if $A \sqsubseteq_\mathcal{T} \exists r.B$, then there exists $E \in \mathsf{BC}_\mathcal{T}$ s.t. $E \in S^G(A, r)$ and $B \in S^G(E)$.*

*Proof.* To show completeness of the completion algorithm, we assume that $B \notin S^G(A)$ (that there is no $E \in \mathsf{BC}_\mathcal{T}$ s.t. $E \in S^G(A, r)$ and $B \in S^G(E)$) and then construct a model $\mathcal{I}_G$ of $\mathcal{T}$ that shows $A \not\sqsubseteq_\mathcal{T} B$ ($A \not\sqsubseteq_\mathcal{T} \exists r.B$, respectively).

To construct the interpretation $\mathcal{I}_G$, we need to map each individual name to a single element of the domain. However, since do not make the unique name assumption, different individuals may be mapped to the same element, so that we need equivalence classes of (equivalent) individuals: $[a] = \{b \in \mathsf{Sig}(\mathcal{T}) \cap N_I \mid \{a\} \in S^G(\{b\})\}$. Because of rule **OR7** and the fact that nominals are always reachable from any $G$, these equivalence classes are well-defined: If $\{a\} \in S^G(\{b\})$, then $S^G(\{a\}) = S^G(\{b\})$. The domain of $\mathcal{I}_G$ will then contain all nominals modulo this equivalence and all concepts that are not subsumed by a nominal and can be reached from $G$ or a nominal using the relation $\leadsto_R$. Thus, we can define $\mathcal{I}_G$ as follows:

$$
\begin{aligned}
\Delta^{\mathcal{I}_G} &= \{[a] \mid a \in \mathsf{Sig}(\mathcal{T}) \cap N_I\} \cup \{A \in \mathsf{Sig}(\mathcal{T}) \cap N_C \mid G \leadsto_R A, \{a\} \notin S^G(A)\} \\
a^{\mathcal{I}_G} &= [a], \text{ for all } a \in N_I[\mathcal{T}] \\
A^{\mathcal{I}_G} &= \{x \mid A \in S^G(x)\} \\
r^{\mathcal{I}_G} &= \{(x, [a]) \in \Delta^{\mathcal{I}_G} \times \Delta^{\mathcal{I}_G} \mid \exists A : A \in S^G(x, r) \wedge \{a\} \in S^G(A)\} \cup \\
&\quad\; \{(x, A) \in \Delta^{\mathcal{I}_G} \times \Delta^{\mathcal{I}_G} \mid A \in S^G(x, r)\}
\end{aligned}
$$

where $x$ is either a nominal or a concept name from the domain $\Delta^{\mathcal{I}_G}$.

This interpretation $\mathcal{I}_G$ is indeed a model of $\mathcal{T}$, i.e., it satisfies all axioms in $\mathcal{T}$:

- Let $A \sqsubseteq B \in \mathcal{T}$ and $x \in A^{\mathcal{I}_G}$. By definition of $\mathcal{I}_G$, this implies $A \in S^G(x)$ and by rule **OR1** also $B \in S^G(x)$. But then we have $x \in B^{\mathcal{I}_G}$.
- Let $A_1 \sqcap A_2 \sqsubseteq B \in \mathcal{T}$ and $x \in (A_1 \sqcap A_2)^{\mathcal{I}_G}$, i.e., $x \in A_1^{\mathcal{I}_G}$ and $x \in A_2^{\mathcal{I}_G}$. By definition of $\mathcal{I}_G$, this implies $A_1, A_2 \in S^G(x)$ and by rule **OR1** also $B \in S^G(x)$. But then we have $x \in B^{\mathcal{I}_G}$.
- Let $A \sqsubseteq \exists r.B \in \mathcal{T}$ and $x \in A^{\mathcal{I}_G}$. By definition of $\mathcal{I}_G$, this implies $A \in S^G(x)$ and by rule **OR3** also $B \in S^G(x, r)$. Since $B \in S^G(B)$, we then have $B \in B^{\mathcal{I}_G}$ and $(x, B) \in r^{\mathcal{I}_G}$ and thus $x \in (\exists r.B)^{\mathcal{I}_G}$.
- Let $\exists r.A \sqsubseteq B \in \mathcal{T}$ and $x \in (\exists r.A)^{\mathcal{I}_G}$, i.e., there exists $x_1 \in \Delta^{\mathcal{I}_G}$ such that $(x, x_1) \in r^{\mathcal{I}_G}$ and $x_1 \in A^{\mathcal{I}_G}$, thus $A \in S^G(x_1)$ by definition of $\mathcal{I}_G$. There are two cases: If $x_1 \in \mathsf{Sig}(\mathcal{T}) \cap N_C$ with $G \leadsto_R x_1, \{a\} \notin S^G(x_1)$, this implies that $x_1 \in S^G(x, r)$, and thus by rule **OR4** we have $B \in S^G(x)$ and finally $x \in B^{\mathcal{I}_G}$. If $x_1 = [a]$ for an individual name $a \in \mathsf{Sig}(\mathcal{T}) \cap N_I$, then we have that there is $y \in \Delta^{\mathcal{I}_G}$ with $y \in S^G(x, r)$ and $\{a\} \in S^G(y)$ by the definition of $\mathcal{I}_G$. Then the completion algorithm will deduce $A \in S^G(y)$ the same way as it did for $A \in S^G(x_1)$, and thus **OR4** yields $B \in S^G(x)$ and hence $x \in B^{\mathcal{I}_G}$.

– Let $r \sqsubseteq s \in \mathcal{T}$ and $(x_1, x_2) \in r^{\mathcal{I}_G}$. There are two cases: If $x_1 \in \mathsf{Sig}(\mathcal{T}) \cap N_C$ with $G \rightsquigarrow_R x_1, \{a\} \notin S^G(x_1)$, then the definition of $\mathcal{I}_G$ implies that $x_2 \in S^G(x_1, r)$ and by rule **OR5** we have $x_2 \in S^G(x_1, s)$. But then $(x_1, x_2) \in s^{\mathcal{I}_G}$. If $x_2 = [a]$ for an individual name $a \in \mathsf{Sig}(\mathcal{T}) \cap N_I$, then we have that there is $y \in \Delta^{\mathcal{I}_G}$ with $y \in S^G(x_1, r)$ and $\{a\} \in S^G(y)$ by the definition of $\mathcal{I}_G$. Then **OR5** yields $y \in S^G(x_1, s)$ by together with $\{a\} \in S^G(y)$ we have $(x_1, x_2) \in s^{\mathcal{I}_G}$.
– Let $r_1 \circ r_2 \sqsubseteq s \in \mathcal{T}$ and $(x_1, x_3) \in (r_1 \circ r_2)^{\mathcal{I}_G}$, i.e., there exists $x_2 \in \Delta^{\mathcal{I}_G}$ with $(x_1, x_2) \in r_1^{\mathcal{I}_G}$ and $(x_2, x_3) \in r_2^{\mathcal{I}_G}$. Again, there are two cases:
  If $x_2 \in \mathsf{Sig}(\mathcal{T}) \cap N_C$ with $G \rightsquigarrow_R x_2, \{a\} \notin S^G(x_2)$, then the definition of $\mathcal{I}_G$ implies that $x_2 \in S^G(x_1, r_1)$. Then we know that, similar to the above case, there is an $y$ with $y \in S^G(x_2, r_2)$ with $x_3 \in S^G(y)$ if $x_3 = [b]$ for an individual $b$ or $y = x_3$ otherwise.
  If $x_2 = [a]$ for an individual name $a \in \mathsf{Sig}(\mathcal{T}) \cap N_I$, then there exists an $y_2 \in \Delta^{\mathcal{I}_G}$ with $y_1 \in S^G(x_1, r_1)$ and $\{a\} \in S^G(y_1)$. Again, similar to above, there is an $y$ with $y \in S^G(x_2, r_2)$ with $x_3 \in S^G(y)$ if $x_3 = [b]$ for an individual $b$ or $y = x_3$ otherwise. The completion algorithm will deduce $y \in S^G(y_1, r_2)$ the same way as it did for $y \in S^G(x_2, r_2)$.
  In both cases, rule **OR6** will yield $y \in S^G(x_1, s)$, and by definition of $\mathcal{I}_G$ we finally have $(x_1, x_3) \in s^{\mathcal{I}_G}$.

Since we assume that $B \notin S^G(A)$ (that there is no $E \in \mathsf{BC}_{\mathcal{T}}$ s.t. $E \in S^G(A, r)$ and $B \in S^G(E)$), we have by definition of $\mathcal{I}_G$ that $A \notin B^{\mathcal{I}_G}$ ($A \notin (\exists r.B)^{\mathcal{I}_G}$). Since we always have $A \in A^{\mathcal{I}_G}$, $\mathcal{I}_G$ shows that $A \not\sqsubseteq_{\mathcal{T}} B$ ($A \not\sqsubseteq_{\mathcal{T}} \exists r.B$, respectively), and thus completeness of the completion algorithm. ☐

Proposition 1 then follows directly from Lemmas 2 and 3.

### A.2 Correctness of the k-lcs algorithm for $\mathcal{ELOR}$

Again, we split the proof for the correctness of the k-lcs algorithm for $\mathcal{ELOR}$ in two parts: first we show that k-lcs computes indeed a common subsumer of the input concepts, then we show that this common subsumer is the least one w.r.t. subsumption.

**Lemma 4.** *Let $\mathcal{T}$ be a $\mathcal{ELOR}$-TBox, $\mathcal{T}'$ be the TBox obtained from $\mathcal{T}$ by applying the normalization rules, $\mathsf{S}$ be the set of completion sets obtained from $\mathcal{T}'$, $A, B$ be concept names, $X, Y$ be basic concepts with $A \rightsquigarrow_R X, B \rightsquigarrow_R Y$, $k$ be a natural number and $L =$ k-lcs-r$(X, Y, \mathsf{S}, k, A, B)$. Then $X \sqsubseteq_{\mathcal{T}'} L$ and $Y \sqsubseteq_{\mathcal{T}'} L$.*

*Proof.* This lemma can be shown by induction on $k$ for the recursive procedure k-lcs-r. For the case $k = 0$, the result

$$L = \bigsqcap_{E \in S^A(X) \cap S^B(Y) \cap \mathsf{BC}_{\mathcal{T}}} E$$

of k-lcs-r is a conjunction of basic concepts, but no existential restrictions. By soundness of the completion rules, we know that $E \in S^A(X) \cap S^B(Y)$ implies

$X \sqsubseteq_{\mathcal{T}'} E$ and $Y \sqsubseteq_{\mathcal{T}'} E$. Since $L$ contains exactly those conjuncts, we also have $X \sqsubseteq_{\mathcal{T}'} L$ and $Y \sqsubseteq_{\mathcal{T}'} L$.

For the case $k > 0$, $L$ is a conjunction of concept names and existential restrictions $\exists r.E$. For the concept names, the same argument as for the case $k = 0$ applies. For existential restrictions of the form $\exists r.\mathsf{k\text{-}lcs\text{-}r}(E, F, \mathsf{S}, k{-}1, A, B)$ with $(E, F) \in S^A(X, r) \times S^B(Y, r)$, $E \in S^A(X, r)$ implies $X \sqsubseteq_{\mathcal{T}'} \exists r.E$ by soundness of the completion algorithm, and similar $Y \sqsubseteq_{\mathcal{T}'} \exists r.F$. Then the induction hypothesis yields that for $L' = \mathsf{k\text{-}lcs\text{-}r}(E, F, \mathsf{S}, k{-}1, A, B)$ we have $E \sqsubseteq_{\mathcal{T}'} L'$ and $F \sqsubseteq_{\mathcal{T}'} L'$ and thus also $X \sqsubseteq_{\mathcal{T}'} \exists r.L'$ and $Y \sqsubseteq_{\mathcal{T}'} \exists r.L'$. All together, this means $X \sqsubseteq_{\mathcal{T}'} L$ and $Y \sqsubseteq_{\mathcal{T}'} L$ is analog. □

**Lemma 5.** *Let $\mathcal{T}$ be a $\mathcal{ELOR}$-TBox, $\mathcal{T}'$ be the TBox obtained from $\mathcal{T}$ by applying the normalization rules, $\mathsf{S}$ be the set of completion sets obtained from $\mathcal{T}'$, $A, B$ be concept names, $X, Y$ be basic concepts with $A \rightsquigarrow_R X$, $B \rightsquigarrow_R Y$, $k$ be a natural number and $L = \mathsf{k\text{-}lcs\text{-}r}(X, Y, \mathsf{S}, k, A, B)$. Then for each $\mathcal{ELOR}$-concept $F$ with $\mathsf{Sig}(F) \subseteq \mathsf{Sig}(\mathcal{T})$ and $rd(F) \leq k$, $X \sqsubseteq_{\mathcal{T}'} F$ and $Y \sqsubseteq_{\mathcal{T}'} F$ imply $L \sqsubseteq_{\mathcal{T}'} F$.*

*Proof.* By induction on the role-depth $rd(F)$. Let $rd(F) = 0$, i.e. $F = \bigsqcap_{i \in I} E_i$ contains no existential restrictions but only basic concepts $E_i$. Since $X \sqsubseteq_{\mathcal{T}'} F$ and $Y \sqsubseteq_{\mathcal{T}'} F$, we also have $X \sqsubseteq_{\mathcal{T}'} E_i$ and $Y \sqsubseteq_{\mathcal{T}'} E_i$ for all conjuncts $E_i$ of $F$. Then, completeness of the completion algorithm yields that $E_i \in S^X(X)$ and since $A \rightsquigarrow_R X$ also $E \in S^A(X)$. Similarly, we have $E_i \in S^B(Y)$ for all conjuncts $E_i$ of $F$ and thus

$$L = \bigsqcap_{E \in S^A(X) \cap S^B(Y) \cap \mathsf{BC}_{\mathcal{T}}} E \sqsubseteq_{\mathcal{T}'} F.$$

If $rd(F) > 0$, $F$ may contain two kinds of conjuncts: basic concepts and existential restrictions. The basic concepts in $F$ must appear in $L$ as well by an argument analog to the case $rd(F) = 0$. Let $\exists r.F'$ be a top-level conjunct of $F$. Since $X \sqsubseteq_{\mathcal{T}'} F$ and $Y \sqsubseteq_{\mathcal{T}'} F$, completeness yields that there exists an $E \in S^A(X, r)$ such that $F' \in S^A(E)$ (i.e. $E \sqsubseteq_{\mathcal{T}'} F'$), and an $E' \in S^B(Y, r)$ such that $F' \in S^B(E')$ (i.e. $E' \sqsubseteq_{\mathcal{T}'} F'$). By induction hypothesis, it follows that $\mathsf{k\text{-}lcs\text{-}r}(E, E', \mathsf{S}, k{-}1, A, B) \sqsubseteq_{\mathcal{T}'} F'$, and hence we also have that $L \sqsubseteq_{\mathcal{T}'} \exists r.\mathsf{k\text{-}lcs\text{-}r}(E, E', \mathsf{S}, k{-}1, A, B) \sqsubseteq_{\mathcal{T}'} \exists r.F'$. All together, we get $L \sqsubseteq_{\mathcal{T}'} F$. □

The correctness of the k-lcs algorithm for $\mathcal{ELOR}$ follows directly from Lemmas 4 and 5:

**Proposition 2.** *Let $\mathcal{T}$ be an $\mathcal{ELOR}$-TBox, $C$ and $D$ be $\mathcal{ELOR}$-concept description and $k$ be a natural number. Then $L = \mathsf{k\text{-}lcs}(C, D, \mathcal{T}, k)$ is the $\mathcal{ELOR}$-lcs of $C$ and $D$ w.r.t. $\mathcal{T}$ and the role-depth bound $k$.*

### A.3 Correctness of the $k$-msc algorithm for $\mathcal{ELOR}$-KBs

As for the k-lcs, we split the proof for Proposition 3 in two parts: first we show that k-msc computes indeed a concept that has the given individual as instance, then we show that this concept is the least one w.r.t. subsumption.

**Lemma 6.** *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a $\mathcal{ELOR}$-KB, $\mathcal{T}'$ be the TBox obtained from $\mathcal{T}$ by absorbing $\mathcal{A}$ and applying the normalization rules,* $\mathsf{S}$ *be the set of completion sets obtained from $\mathcal{T}'$, $X$ be a basic concept with $\top \rightsquigarrow_R X$, $k$ be a natural number and $L =$ traversal-concept$(X, \mathsf{S}, k)$. Then $X \sqsubseteq_{\mathcal{T}'} L$.*

*Proof.* By induction on $k$. For the case $k = 0$, the result $L = \bigsqcap_{i \in I} E_i$ of traversal-concept is a conjunction of concepts names $E_i \in S^{\top}(X) \cap \mathsf{BC}_{\mathcal{K}}$, but no existential restrictions. By soundness of the completion rules, we know that $E_i \in S^{\top}(X)$ implies $X \sqsubseteq_{\mathcal{T}'} E$. Since $L$ contains exactly those conjuncts, we also have $X \sqsubseteq_{\mathcal{T}'} L$.

For the case $k > 0$, $L$ is a conjunction of concept names and existential restrictions $\exists r.E$. For the concept names, the same argument as for the case $k = 0$ applies. For existential restrictions of the form $\exists r.$traversal-concept$(E, \mathsf{S}, k{-}1)$ with $E \in S^{\top}(X, r)$, soundness yields $X \sqsubseteq_{\mathcal{T}'} \exists r.E$. Then the induction hypothesis yields that for $L' =$ traversal-concept$(E, \mathsf{S}, k{-}1)$ we have $E \sqsubseteq_{\mathcal{T}'} L'$ and thus also $X \sqsubseteq_{\mathcal{T}'} \exists r.L'$. All together, this means $X \sqsubseteq_{\mathcal{T}'} L$. □

**Lemma 7.** *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a $\mathcal{ELOR}$-KB, $\mathcal{T}'$ be the TBox obtained from $\mathcal{T}$ by absorbing $\mathcal{A}$ and applying the normalization rules,* $\mathsf{S}$ *be the set of completion sets obtained from $\mathcal{T}'$, $X$ be a basic concept with $\top \rightsquigarrow_R X$, $k$ be a natural number and $L =$ traversal-concept$(X, \mathsf{S}, k)$. Then for each $\mathcal{ELR}$-concept $F$ with* $\mathsf{Sig}(\mathsf{F}) \subseteq \mathsf{Sig}(\mathcal{K})$ *and $rd(F) \leq k$, $X \sqsubseteq_{\mathcal{T}'} F$ implies $L \sqsubseteq_{\mathcal{T}'} F$.*

*Proof.* By induction on the role-depth $rd(F)$. Let $rd(F) = 0$, i.e. $F = \bigsqcap_{i \in I} E_i$ contains no existential restrictions. Since $X \sqsubseteq_{\mathcal{T}'} F$, we also have $X \sqsubseteq_{\mathcal{T}'} E_i$ for all conjuncts $E_i$ of $F$. Then, completeness of the completion algorithm yields that $E_i \in S^X(X)$ and since $\top \rightsquigarrow_R X$ also $E_i \in S^{\top}(X)$. Therefore

$$L = \bigsqcap_{E \in S^{\top}(X) \cap (\mathsf{Sig}(\mathcal{K}) \cap N_C \cup \{\top\})} E \sqsubseteq_{\mathcal{T}'} F.$$

If $rd(F) > 0$, $F$ may contain two kinds of conjuncts: concepts names and existential restrictions. The concepts names in $F$ must appear in $L$ as well by by the same argument as in case $rd(F) = 0$. Let $\exists r.F'$ be a top-level conjunct of $F$. Since $X \sqsubseteq_{\mathcal{T}'} F \sqsubseteq_{\mathcal{T}'} \exists r.F'$, completeness yields that there exists an $E \in S^{\top}(X, r)$ such that $F' \in S^{\top}(E)$, i.e. $E \sqsubseteq_{\mathcal{T}'} F'$. Since $rd(F') < rd(F)$, the induction hypothesis yields traversal-concept$(E, \mathsf{S}, k{-}1) \sqsubseteq_{\mathcal{T}'} F'$, and thus also $L \sqsubseteq_{\mathcal{T}'} \exists r.$traversal-concept$(E, \mathsf{S}, k{-}1) \sqsubseteq_{\mathcal{T}'} \exists r.F'$. Everything together we get $L \sqsubseteq_{\mathcal{T}'} F$. □

*Note.* In the previous lemmas, if $X$ is a nominal $\{a\}$, then $\{a\} \sqsubseteq_{\mathcal{T}'} L$ is equivalent to $\mathcal{T}' \models L(a)$.

The correctness of the k-msc algorithm for $\mathcal{ELOR}$-KBs follows directly from Lemmas 6 and 7:

**Proposition 3.** *Let $\mathcal{K}$ be an $\mathcal{ELOR}$-KB, $a$ be an individual occuring in $\mathcal{K}$ and $k$ be a natural number. Then $M =$ k-msc$(a, \mathcal{K}, k)$ is the $\mathcal{ELR}$-msc of $a$ w.r.t. $\mathcal{K}$ and the role-depth bound $k$.*