Most Specific Generalizations w.r.t. General $\mathcal{EL}$-TBoxes

Benjamin Zarrieß and **Anni-Yasmin Turhan***
Institute for Theoretical Computer Science, Technische Universität Dresden, Germany
{zarriess,turhan}@tcs.inf.tu-dresden.de

Abstract

In the area of Description Logics the least common subsumer (lcs) and the most specific concept (msc) are inferences that generalize a set of concepts or an individual, respectively, into a single concept. If computed w.r.t. a general $\mathcal{EL}$-TBox neither the lcs nor the msc need to exist. So far in this setting no exact conditions for the existence of lcs- or msc-concepts are known. This paper provides necessary and sufficient conditions for the existence of these two kinds of concepts. For the lcs of a fixed number of concepts and the msc we show decidability of the existence in PTime and polynomial bounds on the maximal role-depth of the lcs- and msc-concepts. This bound allows to compute the lcs and the msc, respectively.

1 Introduction

Description Logics (DL) allow to model application domains in a structured and well-understood way. Due to their formal semantics, DLs can offer powerful reasoning services.

In recent years the lightweight DL $\mathcal{EL}$ became popular as an ontology language for large-scale ontologies. $\mathcal{EL}$ provides the logical underpinning of the OWL 2 EL profile of the W3C web ontology language [W3C OWL Working Group, 2009], which is used in important life science ontologies, as for instance, SNOMED CT [Spackman, 2000] and the thesaurus of the US national cancer institute (NCI) [Sioutos *et al.*, 2007], which contain ten thousands of concepts. The reason for the success of $\mathcal{EL}$ is that it offers limited, but sufficient expressive power, while reasoning can still be done in polynomial time [Baader *et al.*, 2005].

In DLs basic categories from an application domain can be captured by *concepts* and binary relations by *roles*. Implications between concepts can be specified in the so-called *TBox*. A *general TBox* allows complex concepts on both sides of implications. Facts from the application domain can be captured by *individuals* and their relations in the *ABox*.

Classical inferences for DLs are *subsumption*, which computes the sub- and super-concept relationships of named concepts and *instance checking*, which determines for a given individual whether it belongs to a given concept. Reasoning support for the design and maintenance of large ontologies can be provided by the *bottom-up approach*, which allows to derive a new concept from a set of example individuals, see [Baader *et al.*, 1999]. For this kind of task the generalization inferences *least common subsumer* (lcs) and *most specific concept* (msc) are investigated for lightweight DLs like $\mathcal{EL}$. The lcs of a collection of concepts is a complex concept that captures all commonalities of these concepts. The msc generalizes an individual into a complex concept, that is the most specific one (w.r.t. subsumption) of which the individual is an instance of.

Unfortunately, neither the lcs nor the msc need to exist, if computed w.r.t. general $\mathcal{EL}$-TBoxes [Baader, 2003] or cyclic ABoxes written in $\mathcal{EL}$ [Küsters and Molitor, 2002]. Let's consider the TBox statements:

$$\begin{aligned}
\mathsf{Penicillin} &\sqsubseteq \mathsf{Antibiotic} \sqcap \exists \mathsf{kills.S\text{-}aureus},\\
\mathsf{Carbapenem} &\sqsubseteq \mathsf{Antibiotic} \sqcap \exists \mathsf{kills.E\text{-}coli},\\
\mathsf{S\text{-}aureus} &\sqsubseteq \mathsf{Bacterium} \sqcap \exists \mathsf{resistantMutant.Penicillin},\\
\mathsf{E\text{-}coli} &\sqsubseteq \mathsf{Bacterium} \sqcap \exists \mathsf{resistantMutant.Carbapenem}
\end{aligned}$$

We want to compute the lcs of Penicillin and Carbapenem. Now, both concepts are defined by the type of bacterium they kill. These, in turn, are defined by the substance a mutant of theirs is resistant to. This leads to a cyclic definition and thus the common subsumer cannot be captured by a finite $\mathcal{EL}$-concept, since this would need to express the cycle. If computed w.r.t. a TBox that extends the above one by the axioms:

$$\begin{aligned}
\mathsf{Antibiotic} &\sqsubseteq \exists \mathsf{kills.Bacterium},\\
\mathsf{Bacterium} &\sqsubseteq \exists \mathsf{resistantMutant.Antibiotic},
\end{aligned}$$

then the lcs of Penicillin and Carbapenem is just Antibiotic. We can observe that the existence of the lcs does not merely depend on whether the TBox is cyclic. In fact, for cyclic $\mathcal{EL}$-TBoxes exact conditions for the existence of the lcs have been devised [Baader, 2004]. However, for the case of general $\mathcal{EL}$-TBoxes such conditions are unknown.

There are several approaches to compute generalizations even in this setting. In [Lutz *et al.*, 2010] an extension of

*Partially supported by the German Research Foundation (DFG) in the Collaborative Research Center 912 "Highly Adaptive Energy-Efficient Computing".

$\mathcal{EL}$ with greatest fixpoints was introduced, where the generalization concepts always exist. Computation algorithms for approximative solutions for the lcs were devised in [Baader *et al.*, 2007; Peñaloza and Turhan, 2011b] and for the msc in [Küsters and Molitor, 2002]. The last two methods simply compute the generalization concept up to a given k, a bound on the maximal nestings of quantifiers. If the lcs or msc exists and a large enough k was given, then these methods yield the exact solutions. However, to obtain the *least* common subsumer and the *most* specific concept by these methods in practice, a decision procedure for the existence of the lcs or msc, resp., and a method for computing a sufficient k are still needed. This paper provides these methods for the lcs and the msc.

In this paper we first introduce basic notions for the DL $\mathcal{EL}$ and its canonical models, which serve as a basis for the characterization of the lcs introduced in the subsequent section. There we show that the characterization can be used to verify whether a given generalization is the most specific one and that the size of the lcs, if it exists, is polynomially bounded in the size of the input, which yields a decision procedure for the existence problem. In Section 4 we show the corresponding results for the msc. We end with some conclusions.

2 Preliminaries

2.1 The Description Logic $\mathcal{EL}$

Let N_C, N_R and N_I be disjoint sets of *concept*, *role* and *individual names*. Let $A \in N_C$ and $r \in N_R$. $\mathcal{EL}$*-concepts* are built according to the syntax rule

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

An *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ consists of a non-empty domain $\Delta^{\mathcal{I}}$ and a function $\cdot^{\mathcal{I}}$ that assigns subsets of $\Delta^{\mathcal{I}}$ to concept names, binary relations on $\Delta^{\mathcal{I}}$ to role names and elements of $\Delta^{\mathcal{I}}$ to individual names. The function is extended to complex concepts in the usual way. For a detailed description of the semantic of DLs see [Baader *et al.*, 2003].

Let C, D denote $\mathcal{EL}$-concepts. A *general concept inclusion* (GCI) is an expression of the form $C \sqsubseteq D$. A *(general) TBox* $\mathcal{T}$ is a finite set of GCIs. A GCI $C \sqsubseteq D$ is satisfied in an interpretation $\mathcal{I}$ if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. An interpretation $\mathcal{I}$ is a *model* of a TBox $\mathcal{T}$ if it satisfies all GCIs in $\mathcal{T}$.

Let $a, b \in N_I$, $r \in N_R$ and C a concept, then $C(a)$ is a *concept assertion* and $r(a, b)$ a *role assertion*. An interpretation $\mathcal{I}$ satisfies an assertion $C(a)$ if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ and $r(a, b)$ if $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ holds. An *ABox* $\mathcal{A}$ is a finite set of assertions. An interpretation $\mathcal{I}$ is a *model* of an ABox $\mathcal{A}$ if it satisfies all assertions in $\mathcal{A}$. A *knowledge base* (KB) $\mathcal{K}$ consists of a TBox and an ABox ($\mathcal{K} = (\mathcal{T}, \mathcal{A})$). An interpretation is a model of $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ if it is a model of $\mathcal{T}$ and $\mathcal{A}$.[1]

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. A reasoning service dealing with a KB is instance checking. An individual a *is instance of* the 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 $\mathcal{K}$. These two reasoning problems can be decided for $\mathcal{EL}$ in polynomial time [Baader *et al.*, 2005].

Based on subsumption and instance checking our two inferences of interest *least common subsumer* (lcs) and *most specific concept* (msc) are defined.

Definition 1. Let C, D be concepts and $\mathcal{T}$ a TBox. The concept E is the *lcs of* C, D *w.r.t.* $\mathcal{T}$ ($\text{lcs}_{\mathcal{T}}(C, D)$) if the properties

1. $C \sqsubseteq_{\mathcal{T}} E$ and $D \sqsubseteq_{\mathcal{T}} E$, and
2. $C \sqsubseteq_{\mathcal{T}} F$ and $D \sqsubseteq_{\mathcal{T}} F$ implies $E \sqsubseteq_{\mathcal{T}} F$.

are satisfied. If a concept E satisfies Property 1 it is a *common subsumer of* C *and* D *w.r.t.* $\mathcal{T}$.

The lcs is unique up to equivalence, while common subsumers are not unique, thus we write $G \in \text{cs}_{\mathcal{T}}(C, D)$.

The *role depth* $rd(C)$ of a concept C denotes the maximal nesting depth of $\exists$ in C. If, in Definition 1 the concepts E and F have a role-depth up to k, then E is the *role-depth bounded lcs* (k-$\text{lcs}_{\mathcal{T}}(C, D)$) of C and D w.r.t. $\mathcal{T}$.

$N_{I,\mathcal{A}}$ is the set of individual names used in an ABox $\mathcal{A}$.

Definition 2. Let $a \in N_{I,\mathcal{A}}$ and $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ a KB. A concept C is the *most specific concept of* a *w.r.t.* $\mathcal{K}$ ($\text{msc}_{\mathcal{K}}(a)$) if it satisfies:

1. $\mathcal{K} \models C(a)$, and 2. $\mathcal{K} \models D(a)$ implies $C \sqsubseteq_{\mathcal{T}} D$.

If in the last definition the concepts C and D have a role-depth limited to k, then C is the *role depth bounded msc* of a w.r.t. $\mathcal{K}$ (k-$\text{msc}_{\mathcal{K}}(a)$). The msc and the k-msc are unique up to equivalence in $\mathcal{EL}$.

2.2 Canonical Models and Simulation Relations

The correctness proof of the computation algorithms for the lcs and msc depends on the characterization of subsumption and instance checking, respectively. In case of an empty TBox, homomorphisms between syntax trees of concepts [Baader *et al.*, 1999] were used. A characterization w.r.t. general TBoxes using *canonical models* and *simulations* was given in [Lutz and Wolter, 2010], which we want to use in the following.

Let X be a concept, TBox, ABox or KB, then $N_{C,X}$ ($N_{R,X}$) denotes the set of concept names (role names) occurring in X and $\mathsf{sub}(X)$ denotes the subconcepts in X.

Definition 3. Let C be a concept and $\mathcal{T}$ a TBox.
The *canonical model* $\mathcal{I}_{C,\mathcal{T}}$ *of* C *and* $\mathcal{T}$ is defined as follows:

$$\begin{aligned}
\Delta^{\mathcal{I}_{C,\mathcal{T}}} &:= \{d_C\} \cup \{d_D \mid \exists r.D \in \mathsf{sub}(C) \cup \mathsf{sub}(\mathcal{T})\}; \\
A^{\mathcal{I}_{C,\mathcal{T}}} &:= \{d_D \mid D \sqsubseteq_{\mathcal{T}} A\}, \text{ for all } A \in N_{C,\mathcal{T}} \\
r^{\mathcal{I}_{C,\mathcal{T}}} &:= \{(d_D, d_E) \mid D \sqsubseteq_{\mathcal{T}} \exists r.E \text{ for } \exists r.E \in \mathsf{sub}(\mathcal{T}) \\
&\qquad \text{or } \exists r.E \text{ is conjunct in } D\}, \text{ for all } r \in N_{R,\mathcal{T}}.
\end{aligned}$$

The notion of a canonical model can be extended to a KB.

[1] Since we only use the DL $\mathcal{EL}$, we write 'concept' instead of '$\mathcal{EL}$-concept' and assume all TBoxes, ABoxes and KBs to be written in $\mathcal{EL}$ in the following.

Definition 4. Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a KB. The *canonical model* $\mathcal{I}_\mathcal{K}$ *w.r.t.* $\mathcal{K}$ is defined as follows:

$$\begin{aligned}
\Delta^{\mathcal{I}_\mathcal{K}} &:= \{d_a \mid a \in N_{I,\mathcal{A}}\} \cup \{d_C \mid \exists r.C \in \mathsf{sub}(\mathcal{K})\}\\
A^{\mathcal{I}_\mathcal{K}} &:= \{d_a \mid \mathcal{K} \models A(a)\} \cup \{d_C \mid C \sqsubseteq_\mathcal{T} A\},\\
&\quad\ \text{for all } A \in N_{C,\mathcal{K}};\\
r^{\mathcal{I}_\mathcal{K}} &:= \{(d_C, d_D) \mid C \sqsubseteq_\mathcal{T} \exists r.D, \exists r.D \in \mathsf{sub}(\mathcal{T})\}\\
&\quad\ \cup\{(d_a, d_b) \mid r(a,b) \in \mathcal{A}\},\ \text{for all } r \in N_{R,\mathcal{K}};\\
a^{\mathcal{I}_\mathcal{K}} &:= d_a, \text{ for all } a \in N_{I,\mathcal{A}}.
\end{aligned}$$

To identify some properties of canonical models we use *simulation relations* between interpretations.

Definition 5. Let $\mathcal{I}_1, \mathcal{I}_2$ be interpretations. $\mathcal{S} \subseteq \Delta^{\mathcal{I}_1} \times \Delta^{\mathcal{I}_2}$ is a *simulation between* $\mathcal{I}_1$ *and* $\mathcal{I}_2$ if the following conditions are satisfied for all $A \in N_C$ and for all $r \in N_R$:

$(S1)$ If $(e_1, e_2) \in \mathcal{S}$ and $e_1 \in A^{\mathcal{I}_1}$, then $e_2 \in A^{\mathcal{I}_2}$.

$(S2)$ If $(e_1, e_2) \in \mathcal{S}$ and $(e_1, e_1') \in r^{\mathcal{I}_1}$, then there exists $e_2' \in \Delta^{\mathcal{I}_2}$ s.t. $(e_2, e_2') \in r^{\mathcal{I}_2}$ and $(e_1', e_2') \in \mathcal{S}$.

The tuple $(\mathcal{I}, d)$ denotes an interpretation $\mathcal{I}$ with $d \in \Delta^\mathcal{I}$. If there exists a simulation $\mathcal{S} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{J}$ with $(d, e) \in \mathcal{S}$, we write $(\mathcal{I}, d) \precsim (\mathcal{J}, e)$ and say $(\mathcal{J}, e)$ *simulates* $(\mathcal{I}, d)$. We write $(\mathcal{I}, d) \simeq (\mathcal{J}, e)$ if $(\mathcal{I}, d) \precsim (\mathcal{J}, e)$ and $(\mathcal{J}, e) \precsim (\mathcal{I}, d)$ holds. We summarize some important properties of canonical models.

Lemma 6. *[Lutz and Wolter, 2010] Let C be a concept and $\mathcal{T}$ a TBox.*

1. *$\mathcal{I}_{C,\mathcal{T}}$ is a model of $\mathcal{T}$.*
2. *For all models $\mathcal{I}$ of $\mathcal{T}$ and all $d \in \Delta^\mathcal{I}$ holds: $d \in C^\mathcal{I}$ iff $(\mathcal{I}_{C,\mathcal{T}}, d_C) \precsim (\mathcal{I}, d)$.*
3. *$C \sqsubseteq_\mathcal{T} D$ iff $d_C \in D^{\mathcal{I}_{C,\mathcal{T}}}$ iff $(\mathcal{I}_{D,\mathcal{T}}, d_D) \precsim (\mathcal{I}_{C,\mathcal{T}}, d_C)$.*

This Lemma gives us a characterization of subsumption. A similar Lemma was shown for the instance relationship.

Lemma 7. *[Lutz and Wolter, 2010] Let $\mathcal{K}$ be a KB. $\mathcal{I}_\mathcal{K}$ satisfies: 1. $\mathcal{I}_\mathcal{K}$ is a model of $\mathcal{K}$. 2. $\mathcal{K} \models C(a)$ iff $d_a \in C^{\mathcal{I}_\mathcal{K}}$.*

Next, we recall some known operations on interpretations. Taking an element d of the domain of an interpretation as the root, the interpretation can be unraveled into a possibly infinite tree. The nodes of the tree are words that correspond to paths starting in d. We have that $\pi = dr_1d_1r_2d_2r_3...$ is a *path* in an interpretation $\mathcal{I}$, if the domain elements d_i and d_{i+1} are connected via $r_{i+1}^\mathcal{I}$ for all i.

Definition 8. Let $\mathcal{I}$ be an interpretation with $d \in \Delta^\mathcal{I}$. The *tree unraveling* $\mathcal{I}_d$ of $\mathcal{I}$ in d is defined as follows:

$$\begin{aligned}
\Delta^{\mathcal{I}_d} &:= \{dr_1d_1r_2...r_nd_n \mid (d_i, d_{i+1}) \in r_{i+1}^\mathcal{I}, i \geq 0, d_0 = d\}\\
A^{\mathcal{I}_d} &:= \{\sigma d' \mid \sigma d' \in \Delta^{\mathcal{I}_d} \wedge d' \in A^\mathcal{I}\}\\
r^{\mathcal{I}_d} &:= \{(\sigma, \sigma r d') \mid (\sigma, \sigma r d') \in \Delta^{\mathcal{I}_d} \times \Delta^{\mathcal{I}_d}\}.
\end{aligned}$$

The *length* of an element $\sigma \in \Delta^{\mathcal{I}_d}$, denoted by $|\sigma|$, is the number of role names occurring in σ. If σ is of the form $dr_1d_1r_2...r_md_m$, then d_m is the *tail* of σ denoted by $\mathsf{tail}(\sigma) = d_m$. The interpretation $\mathcal{I}_d^\ell$ denotes the finite subtree of the tree unraveling $\mathcal{I}_d$ up to depth ℓ. Such a tree can be translated into an *ℓ-characteristic concept* of an interpretation $(\mathcal{I}, d)$.

Definition 9. Let $(\mathcal{I}, d)$ be an interpretation. The ℓ-characteristic concept $X^\ell(\mathcal{I}, d)$ is defined as follows:

- $X^0(\mathcal{I}, d) := \bigsqcap\{A \in N_C \mid d \in A^\mathcal{I}\}$
- $X^\ell(\mathcal{I}, d) :=$
$$X^0(\mathcal{I}, d) \sqcap \bigsqcap_{r \in N_R} \bigsqcap\{\exists r.X^{\ell-1}(\mathcal{I}, d') \mid (d, d') \in r^\mathcal{I}\}$$

3 Existence of Least Common Subsumers

In this section we develop a decision procedure for the problem whether for two given concepts and a given TBox the least common subsumer of these two concepts exists w.r.t. the given TBox. If not stated otherwise, the two input concepts are denoted by C and D and the TBox by $\mathcal{T}$.

Similar to the approach used in [Baader, 2004] we proceed by the following steps:

1. Devise a method to identify lcs-candidates for the lcs. The set of lcs-candidates is a possibly infinite set of common subsumers of C and D w.r.t. $\mathcal{T}$, such that if the lcs exists then one of these lcs-candidates actually is the lcs.

2. Characterize the existence of the lcs. Find a condition such that the problem whether a given common subsumer of C and D w.r.t. $\mathcal{T}$ is least (w.r.t. $\sqsubseteq_\mathcal{T}$), can be decided by testing this condition.

3. Establish an upper bound on the role-depth of the lcs. We give a bound ℓ such that *if the lcs exists*, then it has a role-depth less or equal ℓ. By such an upper bound one needs to check only for finitely many of the lcs-candidates if they are least (w.r.t. $\sqsubseteq_\mathcal{T}$).

The next subsection addresses the first two problems, afterwards we show that such a desired upper bound exists.

3.1 Characterizing the Existence of the lcs

The characterization presented here is based on the product of canonical models. This product construction is adopted from [Baader, 2003; Lutz *et al.*, 2010] where it was used to compute the lcs in $\mathcal{EL}$ with gfp-semantics and in the DL $\mathcal{EL}^\nu$, respectively.

To obtain the k-lcs$_\mathcal{T}(C, D)$ we build the product of the canonical models $(\mathcal{I}_{C,\mathcal{T}}, d_C)$ and $(\mathcal{I}_{D,\mathcal{T}}, d_D)$ and then take the k-characteristic concept of this product model.

Lemma 10. *Let $k \in \mathbb{N}$.*

1. *$X^k(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D)) \in cs_\mathcal{T}(C, D)$.*
2. *Let $E \in cs_\mathcal{T}(C, D)$ with $rd(E) \leq k$. It holds that $X^k(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D)) \sqsubseteq_\mathcal{T} E$.*

This and all the proofs omitted in this paper due to space constraints can be found in [Zarrieß and Turhan, 2013].

In the following we take $X^k(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ as a representation of the k-lcs$_\mathcal{T}(C, D)$. It is implied by Lemma 10 that the set of k-characteristic concepts of the product model $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ for all k is the set of lcs-candidates for the lcs$_\mathcal{T}(C, D)$, which can be stated as follows.

Corollary 11. *The lcs$_\mathcal{T}(C, D)$ exists iff there exists a $k \in \mathbb{N}$ such that for all $\ell \in \mathbb{N}$: k-lcs$_\mathcal{T}(C, D) \sqsubseteq_\mathcal{T} \ell$-lcs$_\mathcal{T}(C, D)$.*

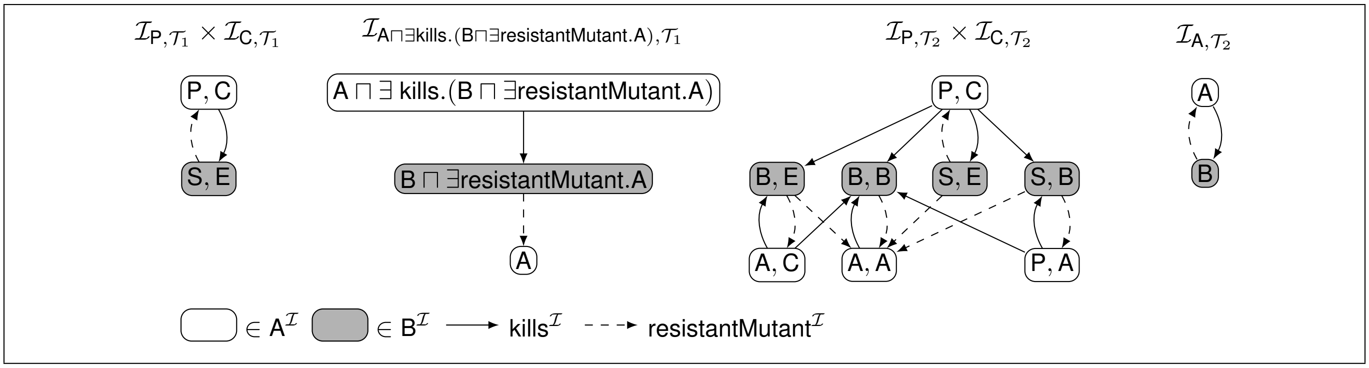

Figure 1: Product of canonical models of $\mathcal{T}_1$ and $\mathcal{T}_2$

Obviously, this doesn't yield a decision procedure for the problem whether the k-$\text{lcs}_\mathcal{T}(C, D)$ is the lcs, since subsumption cannot be checked for infinitely many ℓ in finite time.

Next, we address step 2 and show a condition on the common subsumers that decides whether a common subsumer is least or not. The main idea is that the product model captures all commonalities of the input concepts by means of canonical models. Thus we compare the canonical models of the common subsumers and the product model using simulation-equivalence $\simeq$.

Lemma 12. *Let E be a concept. $E \equiv_\mathcal{T} lcs_\mathcal{T}(C, D)$ iff $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D)) \simeq (\mathcal{I}_{E,\mathcal{T}}, d_E)$.*

Proof sketch. For any $F \in \text{cs}_\mathcal{T}(C, D)$ it holds by Lemma 6, Claim 3 that $(\mathcal{I}_{F,\mathcal{T}}, d_F)$ is simulated by $(\mathcal{I}_{C,\mathcal{T}}, d_C)$ and $(\mathcal{I}_{D,\mathcal{T}}, d_D)$ and therefore also by $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$.

Assume $(\mathcal{I}_{E,\mathcal{T}}, d_E)$ is simulation-equivalent to the product model. We need to show that $E \equiv_\mathcal{T} \text{lcs}_\mathcal{T}(C, D)$. By transitivity of $\lesssim$ it is implied that $(\mathcal{I}_{F,\mathcal{T}}, d_F) \lesssim (\mathcal{I}_{E,\mathcal{T}}, d_E)$ and $E \sqsubseteq_\mathcal{T} F$ by Lemma 6. Therefore $E \equiv_\mathcal{T} \text{lcs}_\mathcal{T}(C, D)$.

For the other direction assume $E \equiv_\mathcal{T} \text{lcs}_\mathcal{T}(C, D)$. It has to be shown that $(\mathcal{I}_{E,\mathcal{T}}, d_E)$ simulates the product model. Let $\mathcal{J}_{(d_C,d_D)}$ be the tree unraveling of the product model. Since E is more specific than the k-characteristic concepts of the product model for all k (by Corollary 11), $(\mathcal{I}_{E,\mathcal{T}}, d_E)$ simulates the subtree $\mathcal{J}^k_{(d_C,d_D)}$ of $\mathcal{J}_{(d_C,d_D)}$ limited to elements up to depth k, for all k. For each k we consider the maximal simulation from $\mathcal{J}^k_{(d_C,d_D)}$ to $(\mathcal{I}_{E,\mathcal{T}}, d_E)$. Note that $((d_C, d_D), d_E)$ is contained in any of these simulations. Let σ be an element of $\Delta^{\mathcal{J}_{(d_C,d_D)}}$ at an arbitrary depth ℓ. We show how to determine the elements of $\Delta^{\mathcal{I}_{E,\mathcal{T}}}$, that simulate this fixed element σ. Let $\mathcal{S}_n(\sigma)$ be the maximal set of elements from $\Delta^{\mathcal{I}_{E,\mathcal{T}}}$ that simulate σ in each of the trees $\mathcal{J}^n_{(d_C,d_D)}$ with $n \geq \ell$. We can observe that the infinite sequence $(\mathcal{S}_{\ell+i}(\sigma))_{i=0,1,2,\ldots}$ is decreasing (w.r.t. $\supseteq$). Therefore at a certain depth we reach a fixpoint set. This fixpoint set exists for any σ. It can be shown that the union of all these fixpoint sets yields a simulation from the product model to $(\mathcal{I}_{E,\mathcal{T}}, d_E)$. □

By the use of this Lemma it can be verified whether a given common subsumer is the least one or not, which we illustrate by an example.

Example 13. Consider again the TBox from the introduction (now displayed with abbreviated concept names)

$$\mathcal{T}_1 = \{\mathsf{P} \sqsubseteq \mathsf{A} \sqcap \exists\mathsf{kills.S},\quad \mathsf{S} \sqsubseteq \mathsf{B} \sqcap \exists\mathsf{resistantMutant.P},\quad \mathsf{C} \sqsubseteq \mathsf{A} \sqcap \exists\mathsf{kills.E},\quad \mathsf{E} \sqsubseteq \mathsf{B} \sqcap \exists\mathsf{resistantMutant.C}\}$$

and the following extended TBox

$$\mathcal{T}_2 = \mathcal{T}_1 \cup \{\mathsf{A} \sqsubseteq \exists\mathsf{kills.B},\ \mathsf{B} \sqsubseteq \exists\mathsf{resistantMutant.A}\}.$$

In Figure 1 we can see that

$$\mathsf{A} \sqcap \exists\mathsf{kills.}(\mathsf{B} \sqcap \exists\mathsf{resistantMutant.A}) \in \text{cs}_{\mathcal{T}_1}(\mathsf{P}, \mathsf{C}),$$

but it is not the lcs, because its canonical model cannot simulate the product model $(\mathcal{I}_{\mathsf{P},\mathcal{T}_1} \times \mathcal{I}_{\mathsf{C},\mathcal{T}_1}, (d_\mathsf{P}, d_\mathsf{C}))$. The concept A, however, is the lcs of P and C w.r.t. $\mathcal{T}_2$. We have $(\mathcal{I}_{\mathsf{P},\mathcal{T}_2} \times \mathcal{I}_{\mathsf{C},\mathcal{T}_2}, (d_\mathsf{P}, d_\mathsf{C})) \lesssim (\mathcal{I}_{\mathsf{A},\mathcal{T}_2}, d_\mathsf{A})$ since any element from $\Delta^{\mathcal{I}_{\mathsf{P},\mathcal{T}_2} \times \mathcal{I}_{\mathsf{C},\mathcal{T}_2}}$ in $\mathsf{A}^{\mathcal{I}_{\mathsf{P},\mathcal{T}_2} \times \mathcal{I}_{\mathsf{C},\mathcal{T}_2}}$ or $\mathsf{B}^{\mathcal{I}_{\mathsf{P},\mathcal{T}_2} \times \mathcal{I}_{\mathsf{C},\mathcal{T}_2}}$ is simulated by Ⓐ or Ⓑ, respectively.

The characterization of the existence of the lcs given in Corollary 11 can be reformulated using Lemma 12.

Corollary 14. *The $lcs_\mathcal{T}(C, D)$ exists iff there exists a k such that the canonical model of $X^k(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ w.r.t. $\mathcal{T}$ simulates $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$.*

This corollary still doesn't yield a decision procedure for the existence problem, since the depth k is still unrestricted. Such a restriction will be developed in the next section.

3.2 A Polynomial Upper Bound on the Role-depth of the lcs

In this section we show that, if the lcs exists, its role-depth is bounded by the size of the product model. First, consider again the TBox $\mathcal{T}_2$ from Example 13, where $\mathsf{A} \sqsubseteq_{\mathcal{T}_2} \exists\mathsf{kills.}(\mathsf{B} \sqcap \exists\mathsf{resistantMutant.A})$ holds, which results in a loop in the product model through the elements (A, A) and (B, B). Furthermore, the cycles in the product model involving the roles kills and resistantMutant are captured by the canonical model $\mathcal{I}_{\mathsf{A},\mathcal{T}_2}$. Therefore $\mathsf{A} \equiv_{\mathcal{T}_2} \text{lcs}_{\mathcal{T}_2}(\mathsf{P}, \mathsf{C})$. On this observation we build our general method.

We call elements $(d_F, d_{F'}) \in \Delta^{\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}}$ *synchronous* if $F = F'$ and *asynchronous* otherwise. The structure of $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ can now be simplified by considering only synchronous successors of synchronous elements.

In order to find a number k, such that the product model is simulated by the canonical model of

$K = X^k(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$, we first represent the model $(\mathcal{I}_{K,\mathcal{T}}, d_K)$ as a subtree of the tree unraveling of the product model $\mathcal{J}_{(d_C,d_D)}$ with root (d_C, d_D). We construct this representation by extending the subtree $\mathcal{J}^k_{(d_C,d_D)}$ by new tree models at depth k. We need to ensure that the resulting interpretation, denoted by $\widehat{\mathcal{J}}^k_{(d_C,d_D)}$, is a model of $\mathcal{T}$, that is simulation-equivalent to $(\mathcal{I}_{K,\mathcal{T}}, d_K)$. The elements $\sigma \in \Delta^{\mathcal{J}^k_{(d_C,d_D)}}$ with $|\sigma| = k$ that we extend and the corresponding trees we append to them are selected as follows: Let M be a conjunction of concept names and $\exists r.F \in \mathsf{sub}(\mathcal{T})$. If $\sigma \in M^{\mathcal{J}^k_{(d_C,d_D)}}$ and $M \sqsubseteq_{\mathcal{T}} \exists r.F$, then we append the tree unraveling of the canonical model $\mathcal{I}_{\exists r.F,\mathcal{T}}$. Furthermore, we consider elements that have a tail that is a synchronous element. If $\mathsf{tail}(\sigma) = (d_F, d_F)$, then F is called *tail concept* of σ. To select the elements with a synchronous tail, that we extend by the canonical model of their tail concept, we use embeddings of $\mathcal{J}^k_{(d_C,d_D)}$ into $(\mathcal{I}_{K,\mathcal{T}}, d_K)$. Let $\mathcal{H} = \{Z_1, ..., Z_n\}$ be the set of all functional simulations Z_i from $\mathcal{J}^k_{(d_C,d_D)}$ to $(\mathcal{I}_{K,\mathcal{T}}, d_K)$ with $Z_i((d_C, d_D)) = d_K$. We say that σ with tail concept F is *matched* by Z_i if $Z_i(\sigma) \in F^{\mathcal{I}_{K,\mathcal{T}}}$. The set of elements $\sigma \in \Delta^{\mathcal{J}^k_{(d_C,d_D)}}$ with $|\sigma| = k$, that are matched by a functional simulation Z_i is called *matching set*, denoted by $\mathcal{M}(Z_i)$. Now consider the set $\mathcal{M}(\mathcal{H}) := \{\mathcal{M}(Z_1), ..., \mathcal{M}(Z_n)\}$. If σ is contained in *all* maximal matching sets from $\mathcal{M}(\mathcal{H})$, then we extend σ by the tree unraveling of the canonical model of its tail concept w.r.t. $\mathcal{T}$.

We can show that the resulting interpretation $\widehat{\mathcal{J}}^k_{(d_C,d_D)}$ has the desired properties.

Lemma 15. *Let $K = X^k(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$. $\widehat{\mathcal{J}}^k_{(d_C,d_D)}$ is a model of $\mathcal{T}$ and $\widehat{\mathcal{J}}^k_{(d_C,d_D)} \simeq (\mathcal{I}_{K,\mathcal{T}}, d_K)$.*

Having this representation of the canonical model of the k-lcs$_{\mathcal{T}}(C, D)$ we first show a sufficient condition for the existence of the lcs.

Corollary 16. *If all cycles in $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$, that are reachable from (d_C, d_D) consist of synchronous elements, then the lcs$_{\mathcal{T}}(C, D)$ exists.*

Proof sketch. There exists an $\ell \in \mathbb{N}$ such that all paths in the tree unraveling $\mathcal{J}_{(d_C,d_D)}$ of $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ starting in (d_C, d_D) have a maximal asynchronous prefix up to length ℓ, i.e., if there exists an element at depth $\geq \ell + 1$, then it is a synchronous element. Consider the number

$$m := \mathsf{max}(\{rd(F) \mid F \in \mathsf{sub}(\mathcal{T}) \cup \{C, D\}\}).$$

We unravel $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ up to depth $\ell + m + 1$ such that we get $\mathcal{J}^{\ell+m+1}_{(d_C,d_D)}$. Now it is ensured that the corresponding model $\widehat{\mathcal{J}}^{\ell+m+1}_{(d_C,d_D)}$ contains all paths with a maximal asynchronous prefix up to length ℓ. It is implied that $\widehat{\mathcal{J}}^{\ell+m+1}_{(d_C,d_D)} = \mathcal{J}_{(d_C,d_D)}$. From Lemma 15 and Corollary 14 it follows that $X^{\ell+m+1}(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ is the lcs. □

As seen in Example 13 for $\mathcal{T}_2$, this is not a necessary condition for the existence of the lcs.

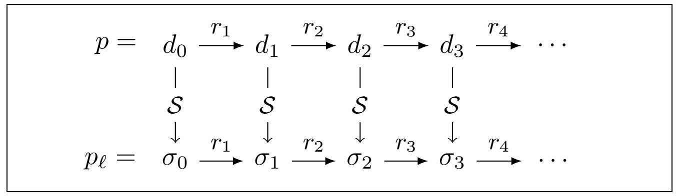

Figure 2: simulation chain of p and p_ℓ

Another consequence of Lemma 15 is, that if the product model $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ has only asynchronous cycles reachable from (d_C, d_D), then the lcs$_{\mathcal{T}}(C, D)$ does not exist. Since in this case $\mathcal{J}_{(d_C,d_D)}$ is infinite but $\widehat{\mathcal{J}}^k_{(d_C,d_D)}$ is finite for all $k \in \mathbb{N}$, a simulation from $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ to $\widehat{\mathcal{J}}^k_{(d_C,d_D)}$ never exists for all k. For instance, this case applies to Example 13 w.r.t. to $\mathcal{T}_1$.

The interesting case is where we have both asynchronous and synchronous cycles reachable from (d_C, d_D) in the product model. In this case we choose a k that is large enough and then check whether the canonical model of $X^k(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ w.r.t. $\mathcal{T}$ simulates the product model.

We show in the next Lemma that the role-depth of the lcs$_{\mathcal{T}}(C, D)$, if it exists, can be bounded by a polynomial, that is quadratic in the size of the product model.

Lemma 17. *Let $n := |\Delta^{\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}}|$ and $m := \mathsf{max}(\{rd(F) \mid F \in \mathsf{sub}(\mathcal{T}) \cup \{C, D\}\})$. If lcs$_{\mathcal{T}}(C, D)$ exists then $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D)) \lesssim \widehat{\mathcal{J}}^{n^2+m+1}_{(d_C,d_D)}$.*

Proof sketch. Assume lcs$_{\mathcal{T}}(C, D)$ exists. From Corollary 14 and Lemma 15 it follows that there exists a number ℓ such that

$$(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D)) \lesssim \widehat{\mathcal{J}}^{\ell}_{(d_C,d_D)}. \quad (1)$$

Every path in $\widehat{\mathcal{J}}^{\ell}_{(d_C,d_D)}$ has a maximal asynchronous prefix of length $\leq \ell$. From depth $\ell + 1$ on there are only synchronous elements in the tree $\widehat{\mathcal{J}}^{\ell}_{(d_C,d_D)}$. From (1) it follows that every path p in $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ starting in (d_C, d_D), is simulated by a corresponding path p_ℓ in $\widehat{\mathcal{J}}^{\ell}_{(d_C,d_D)}$ also starting in (d_C, d_D). The *simulation chain* of p and p_ℓ is depicted in Figure 2. The idea is to use the simulating path p_ℓ to construct a simulating path in $\widehat{\mathcal{J}}^{\ell}_{(d_C,d_D)}$ (also starting in (d_C, d_D)) with a maximal asynchronous prefix of length $\leq n^2$, where n^2 is the number of pairs of elements from $\Delta^{\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}}$. Intuitively, if p_ℓ has a maximal asynchronous prefix that is longer than n^2, then there are pairs in the simulation chain that occur more than once. This is used to construct a simulating path with a shorter maximal asynchronous prefix step-wise. After a finite number of steps the result is a simulating path, such that all pairs consisting of asynchronous elements in the corresponding simulation chain are pairwise distinct. Therefore we need only asynchronous elements from $\widehat{\mathcal{J}}^{\ell}_{(d_C,d_D)}$ up to depth n^2 to simulate the product model. Then we add $m+1$ to n^2 to ensure that $\widehat{\mathcal{J}}^{n^2+m+1}_{(d_C,d_D)}$ contains *all* paths from $\mathcal{J}_{(d_C,d_D)}$ starting in (d_C, d_D), that have a maximal asynchronous prefix of length $\leq n^2$. As argued above $\widehat{\mathcal{J}}^{n^2+m+1}_{(d_C,d_D)}$ simulates $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$. □

Using Lemma 12 and Lemma 17 we can now show the main result of this paper.

Theorem 18. *Let C, D be concepts and $\mathcal{T}$ a general TBox. It is decidable in polynomial time whether the lcs$_\mathcal{T}(C, D)$ exists. If the lcs$_\mathcal{T}(C, D)$ exists it can be computed in polynomial time.*

Proof. First we compute the bound k as given in Lemma 17 and then the k-characteristic concept K of $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$. The canonical model of K can be build according to Definition 3 in polynomial time [Baader *et al.*, 2005]. Next we check whether $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D)) \lesssim (\mathcal{I}_{K,\mathcal{T}}, d_K)$ holds, which can be done in polynomial time. If yes, K is the lcs by Lemma 12 and if no, the lcs doesn't exist by Lemma 17. □

The results from this section can be easily generalized to the lcs of an arbitrary set of concepts $M = \{C_1, ..., C_m\}$ w.r.t. a TBox $\mathcal{T}$. But in this case the size of the lcs is already exponential w.r.t. an empty TBox [Baader *et al.*, 1999]. In this general case we have to take the product model

$$(\mathcal{I}_{C_1,\mathcal{T}} \times \cdots \times \mathcal{I}_{C_m,\mathcal{T}}, (d_{C_1}, \cdots, d_{C_m})),$$

whose size is exponential in the size of M and $\mathcal{T}$, as input for the methods introduced in this section. Then the same steps as for the binary version can be applied.

4 Existence of Most Specific Concepts

We show now that the results obtained for the lcs, can be easily applied to the existence problem of the msc.

Example 19 (From [Küsters and Molitor, 2002]). The msc of the individual a w.r.t. the following KB

$$\mathcal{K}_1 = (\emptyset, \mathcal{A}_1), \text{ with } \mathcal{A}_1 = \{r(a, a)\}$$

doesn't exist, whereas w.r.t. the modified KB

$$\mathcal{K}_2 = (\{C \sqsubseteq \exists r.C\}, \mathcal{A}_2), \text{with } \mathcal{A}_2 = \mathcal{A}_1 \cup \{C(a)\}$$

C is the msc of a.

To decide existence of the msc of an individual a w.r.t. a KB $\mathcal{K} = (\mathcal{T}, \mathcal{A})$, we again start with defining the set of msc-candidates for the msc by taking the k-characteristic concept of the canonical model $(\mathcal{I}_\mathcal{K}, d_a)$ of $\mathcal{K}$.

Lemma 20. *Let $k \in \mathbb{N}$. It holds that $\mathcal{K} \models X^k(\mathcal{I}_\mathcal{K}, d_a)(a)$ and for a concept E with $rd(E) \leq k$, $\mathcal{K} \models E(a)$ implies $X^k(\mathcal{I}_\mathcal{K}, d_a) \sqsubseteq_\mathcal{T} E$.*

Therefore $X^k(\mathcal{I}_\mathcal{K}, d_a) \equiv_\mathcal{T} k$-msc$_\mathcal{K}(a)$. Now we use the canonical model of $X^k(\mathcal{I}_\mathcal{K}, d_a)$ w.r.t. the TBox component $\mathcal{T}$ of $\mathcal{K}$ and the model $(\mathcal{I}_\mathcal{K}, d_a)$ to check whether $X^k(\mathcal{I}_\mathcal{K}, d_a)$ is the *most* specific concept.

Lemma 21. *For a concept C it holds that $C \equiv_\mathcal{T}$ msc$_\mathcal{K}(a)$ iff $(\mathcal{I}_\mathcal{K}, d_a) \simeq (\mathcal{I}_{C,\mathcal{T}}, d_C)$.*

By this Lemma the existence of the msc can be characterized as follows.

Corollary 22. *The msc$_\mathcal{K}(a)$ exists iff there exists a k such that the canonical model of $X^k(\mathcal{I}_\mathcal{K}, d_a)$ w.r.t. $\mathcal{T}$ simulates $(\mathcal{I}_\mathcal{K}, d_a)$.*

To decide whether an appropriate k exists such that $X^k(\mathcal{I}_\mathcal{K}, d_a)$ simulates $(\mathcal{I}_\mathcal{K}, d_a)$, we further examine the structure of $(\mathcal{I}_\mathcal{K}, d_a)$. In Example 19 d_a has a self-loop in the model $(\mathcal{I}_{\mathcal{K}_1}, d_a)$, but the canonical models of $X^k(\mathcal{I}_{\mathcal{K}_1}, d_a)$ are finite for all $k \in \mathbb{N}$, because the TBox is empty. Therefore a simulation never exists. In comparison, the model $(\mathcal{I}_{\mathcal{K}_2}, d_a)$ has additionally a self-loop at d_C and the canonical models of $X^k(\mathcal{I}_{\mathcal{K}_2}, d_a)$ w.r.t. $\mathcal{T}_2$ also contain this loop.

Intuitively, in the general case, the elements in $\Delta^{\mathcal{I}_\mathcal{K}}$, that are elements in $b^{\mathcal{I}_\mathcal{K}}$ (for $b \in N_{I,\mathcal{A}}$), correspond to the asynchronous elements of the product of canonical models and the elements $d_C \in \Delta^{\mathcal{I}_\mathcal{K}}$ for some concept C, correspond to the synchronous elements. The model $(\mathcal{I}_\mathcal{K}, d_a)$ also has an analogous structure compared to the product model $(\mathcal{I}_{C,\mathcal{T}} \times \mathcal{I}_{D,\mathcal{T}}, (d_C, d_D))$ in the sense that elements in $\Delta^{\mathcal{I}_\mathcal{K}}$, that belong to concepts only have successor elements that belong to concepts. Therefore similar arguments as presented in Section 3.2 can be used to show, that a representation of the canonical model of $X^k(\mathcal{I}_\mathcal{K}, d_a)$ as a subtree of the tree unraveling of $(\mathcal{I}_\mathcal{K}, d_a)$ can be obtained. This representation is denoted by $\widehat{\mathcal{J}}^k_{d_a}$. This model is used to show an upper bound on the role-depth k of the msc.

Lemma 23. *Let $m := \mathsf{max}(\{rd(F) \mid F \in \mathsf{sub}(\mathcal{K})\})$ and $n := |N_{I,\mathcal{A}}|$. If the msc$_\mathcal{K}(a)$ exists, then $(\mathcal{I}_\mathcal{K}, d_a) \lesssim \widehat{\mathcal{J}}^{n^2+m+1}_{d_a}$.*

The results of this section can be summarized in the following theorem.

Theorem 24. *Let $\mathcal{K} = (\mathcal{T}, \mathcal{A})$ be a KB and $a \in N_{I,\mathcal{A}}$. It is decidable in polynomial time whether the msc$_\mathcal{K}(a)$ exists. If the msc$_\mathcal{K}(a)$ exists, it can be computed in polynomial time.*

Proof sketch. First we compute the bound k as given in Lemma 23 and then the k-characteristic concept $X^k(\mathcal{I}_\mathcal{K}, d_a)$. The canonical model of $\mathcal{K}$ can be build according to Definition 4 in polynomial time [Baader *et al.*, 2005]. Then we check whether $(\mathcal{I}_\mathcal{K}, d_a) \lesssim (\mathcal{I}_{C,\mathcal{T}}, d_C)$ holds, which can be done in polynomial time. If yes, C is the msc and if no, the msc doesn't exist by Corollary 22. □

All the proofs omitted here due to space constraints are given in [Zarrieß and Turhan, 2013].

5 Conclusions

In this paper we have studied the conditions for the existence of the lcs and of the msc, if computed w.r.t. general TBoxes or cyclic ABoxes, respectively, written in the DL $\mathcal{EL}$. In this setting neither the lcs nor the msc need to exist. It was an open problem to give necessary and sufficient conditions for their existence. We showed that the existence problem of the msc and the lcs of two concepts is decidable in polynomial time. Furthermore, we showed that the role-depth of these most specific generalizations can be bounded by a polynomial. This upper bound k can be used to compute the msc or lcs, if it exists. Otherwise the computed concept can still serve as an approximation [Peñaloza and Turhan, 2011b].

References

[Baader *et al.*, 1999] 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.

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

[Baader *et al.*, 2005] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *Proceedings of the Nineteenth International Joint Conference on Artificial Intelligence IJCAI-05*. Morgan-Kaufmann Publishers, 2005.

[Baader *et al.*, 2007] F. Baader, B. Sertkaya, and A.-Y. Turhan. Computing the least common subsumer w.r.t. a background terminology. *Journal of Applied Logic*, 5(3):392–420, 2007.

[Baader, 2003] F. Baader. Least common subsumers and most specific concepts in a description logic with existential restrictions and terminological cycles. In Georg Gottlob and Toby Walsh, editors, *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, pages 319–324. Morgan Kaufman, 2003.

[Baader, 2004] F. Baader. A graph-theoretic generalization of the least common subsumer and the most specific concept in the description logic $\mathcal{EL}$. In J. Hromkovic and M. Nagl, editors, *Proceedings of the 30th International Workshop on Graph-Theoretic Concepts in Computer Science (WG 2004)*, volume 3353 of *Lecture Notes in Computer Science*, pages 177–188. Springer-Verlag, 2004.

[Küsters and Molitor, 2002] R. Küsters and R. Molitor. Approximating most specific concepts in description logics with existential restrictions. *AI Communications*, 15(1):47–59, 2002.

[Lutz and Wolter, 2010] C. Lutz and F. Wolter. Deciding inseparability and conservative extensions in the description logic $\mathcal{EL}$. *Journal of Symbolic Computation*, 45(2):194–228, 2010.

[Lutz *et al.*, 2010] C. Lutz, R. Piro, and F. Wolter. Enriching $\mathcal{EL}$-concepts with greatest fixpoints. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI'10)*. IOS Press, 2010.

[Peñaloza and Turhan, 2011b] 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.

[Sioutos *et al.*, 2007] N. Sioutos, S. de Coronado, M. W. Haber, F. W. Hartel, Wen-Ling Shaiu, and Lawrence W. Wright. NCI thesaurus: A semantic model integrating cancer-related clinical and molecular information. *J. of Biomedical Informatics*, 40(1):30–43, 2007.

[Spackman, 2000] 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.

[W3C OWL Working Group, 2009] 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/`.

[Zarrieß and Turhan, 2013] B. Zarrieß and A.-Y. Turhan. Most specific generalizations w.r.t. general $\mathcal{EL}$-TBoxes. LTCS-Report 13-06, Chair of Automata Theory, Institute of Theoretical Computer Science, Technische Universität Dresden, Dresden, Germany, 2013. See `http://lat.inf.tu-dresden.de/research/reports.html`.