Least Common Subsumers and Most Specific Concepts in a Description Logic with Existential Restrictions and Terminological Cycles*

Franz Baader
Theoretical Computer Science
TU Dresden
D-01062 Dresden, Germany
baader@inf.tu-dresden.dea

Abstract

Computing least common subsumers (lcs) and most specific concepts (msc) are inference tasks that can support the bottom-up construction of knowledge bases in description logics. In description logics with existential restrictions, the most specific concept need not exist if one restricts the attention to concept descriptions or acyclic TBoxes. In this paper, we extend the notions les and msc to cyclic TBoxes. For the description logic *EC* (which allows for conjunctions, existential restrictions, and the top-concept), we show that the les and msc always exist and can be computed in polynomial time if we interpret cyclic definitions with greatest fixpoint semantics.

1 Introduction

Computing the most specific concept of an individual and the least common subsumer of concepts can be used in the bottom-up construction of description logic (DL) knowledge bases. Instead of defining the relevant concepts of an application domain from scratch, this methodology allows the user to give typical examples of individuals belonging to the concept to be defined. These individuals are then generalized to a concept by first computing the most specific concept of each individual (i.e., the least concept description in the available description language that has this individual as an instance), and then computing the least common subsumer of these concepts (i.e., the least concept description in the available description language that subsumes all these concepts). The knowledge engineer can then use the computed concept as a starting point for the concept definition.

The least common subsumer (les) in DLs with existential restrictions was investigated in [Baader *et* a/., 1999]. In particular, it was shown there that the les in the small DL *EC* (which allows for conjunctions, existential restrictions, and the top-concept) always exists, and that the binary les can be computed in polynomial time. Unfortunately, the most specific concept (msc) of a given ABox individual need not exist in languages allowing for existential restrictions or number restrictions. As a possible solution to this problem, Kiisters and Molitor [2001] show how the most specific concept can be approximated in *EC* and some of its extensions. Here, we follow an alternative approach: we extend the language by cyclic terminologies with greatest fixpoint (gfp) semantics, and show that the msc always exists in this setting. Of course, then one must also be able to compute the les w.r.t. cyclic terminologies with gfp-semantics. For the DL *ACM* (which allows for conjunctions, value restrictions, and number restrictions) it was shown in [Baader and Kusters, 1998] that the most specific concept always exists if one adds cyclic concept definitions with gfp-semantics. One reason for Kusters and Molitor to choose an approximation approach rather than an exact characterization of the most specific concept using cyclic definitions was that the impact of cyclic definitions in description logics with existential restrictions was largely unexplored.

The paper [Baader, 2003a] is a first step toward overcoming this deficit. It considers cyclic terminologies in *EC* w.r.t. the three types of semantics (greatest fixpoint, least fixpoint, and descriptive semantics) introduced by Nebel [1991], and shows that the subsumption problem can be decided in polynomial time in all three cases. This is in stark contrast to the case of DLs with value restrictions. Even for the small DL F_{Lo} (which allows conjunctions and value restrictions only), adding cyclic terminologies increases the complexity of the subsumption problem from polynomial (for concept descriptions) to PSPACE. The main tool in the investigation of cyclic definitions in *EC* is a characterization of subsumption through the existence of so-called simulation relations on the graph associated with an EL-terminology, which can be computed in polynomial time [Henzinger *et al,* 1995].

This characterization of subsumption can be used to characterize the les w.r.t. gfp-semantics via the product of this graph with itself (Section 4). This shows that, w.r.t. gfp semantics, the les always exists, and that the binary les can be computed in polynomial time. (The n-ary les may grow exponentially even in *EC* without cyclic terminologies [Baader *et al. ,* 1999].)

The characterization of subsumption w.r.t. gfp-semantics can be extended to the instance problem in *EC.* This allows us to show that the msc in *EC* with cyclic terminologies interpreted with gfp semantics always exists, and can be computed in polynomial time (Section 5).

In the next section, we introduce *E L* with cyclic terminolo-

* Partially supported by the DFG under grant BA 1122/4-3.

Name	Syntax	Semantics
concept name	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
role name	r	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
top-concept	$\top$	$\Delta^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
exist. restriction	$\exists r.C$	$\{x \mid \exists y : (x,y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
concept definition	$A \equiv D$	$A^{\mathcal{I}} = D^{\mathcal{I}}$
individual name	a	$a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$
concept assertion	$A(a)$	$a^{\mathcal{I}} \in A^{\mathcal{I}}$
role assertion	$r(a,b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$

Table 1: Syntax and semantics of $\mathcal{EL}$.

gics as well as the lcs and the msc. Then we recall the important definitions and results from [Baader, 2003a]. Section 4 formulates and proves the new results for the lcs, and Section 5 does the same for the msc.

2 Cyclic terminologies, least common subsumers, and most specific concepts

Concept descriptions arc inductively defined with the help of a set of *constructors,* starting with a set N_C of *concept names* and a set *NR* of *role names.* The constructors determine the expressive power of the DL. In this paper, wc restrict the attention to the DL $\mathcal{EL}$, whose concept descriptions arc formed using the constructors top-concept (T), conjunction $(C \sqcap D)$, and existential restriction $(\exists r.C)$.. The semantics of E L-concept descriptions is defined in terms of an *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$. The d o m$\Delta^{\mathcal{I}}$ of $\mathcal{I}$ s a non-empty set of individuals and the interpretation function I maps each concept name $A \in N_C$ to a subset $A^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$ and each role $r \in N_R$ to a binary relation $r^{\mathcal{I}}$ on $\Delta^{\mathcal{I}}$. The extension of $\cdot^{\mathcal{I}}$ to arbitrary concept descriptions is inductively defined, as shown in the third column of Table 1.

A *terminology* (or *TBox* for short) is a finite set of concept definitions of the form $A \equiv D$, where A is a concept name and *D* a concept description. In addition, we require that TBoxcs do not contain *multiple definitions,* i.e., there cannot be two distinct concept descriptions D_1 and D_2 such that both $A \equiv D_1$ and $A \equiv D_2$ belongs to the TBox. Concept names occurring on the left-hand side of a definition are called *defined concepts.* All other concept names occurring in the TBox are called *primitive concepts.* Note that we allow for cyclic dependencies between the defined concepts, i.e., the definition of *A* may refer (directly or indirectly) to *A* itself. An interpretation I is a model of the TBox T iff it satisfies all its concept definitions, i.e., $A^{\mathcal{I}} = D^{\mathcal{I}}$ for all definitions $A \equiv D$ in T.

An *ABox* is a finite set of assertions of the form *A(a)* and r(a,b), where *A* is a concept name, *r* is a role name, and a, *b* are individual names from a set N_I. Interpretations of ABoxes must additionally map each individual name $a \in N_I$ to an element $a^{\mathcal{I}}$ of $\Delta^{\mathcal{I}}$. An interpretation I is a model of the ABox *A* iff it satisfies all its a s:ions, i.e., $a^{\mathcal{I}} \in A^{\mathcal{I}}$ o r all concept assertions *A(a)* in *A* and $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ for all role assertions $r(a,b)$ in *A*. The interpretation I is a model of the A Box *A* together with the TBox $\mathcal{T}$ iff it is a model of both *T and A.*

The semantics of (possibly cyclic) $\mathcal{EL}$-TBoxes we have defined above is called *descriptive semantic* by Nebel [1991]. For some applications, it is more appropriate to interpret cyclic concept definitions with the help of a fixpoint semantics.

Example 1 To illustrate this, let us recall an example from LBaader, 2003a]: $\mathsf{Inode} \equiv \mathsf{Node} \sqcap \exists \mathsf{edge}.\mathsf{Inode}$

Here the intended interpretations arc graphs where we have nodes (elements of the concept Node) and edges (represented by the role edge), and we want to define the concept I node of all nodes lying on an infinite (possibly cyclic) path of the graph. In order to capture this intuition, the above definition must be interpreted with greatest fixpoint semantics.

Before we can define greatest fixpoint semantics (gfp-semantics), we must introduce some notation. Let *T* be an E L-TBox containing the r o N_{role}, the primitive concepts N_{prim}, and the defined concepts $N_{def} = \{A_1, \ldots, A_k\}$. A *primitive interpretations J* for T is given by a domain , an interpretation of the roles $r \in N_{role}$ by binary relations $r^{\mathcal{J}}$ on $\Delta^{\mathcal{J}}$, and an interpretation of the primitive concepts $P \in N_{prim}$ by subsets $P^{\mathcal{J}}$ of $\Delta^{\mathcal{J}}$. Obviously, a primitive interpretation differs from an interpretation in that it does not interpret the defined concepts in N_{def}. We say that the interpretation *X* is *based on* the primitive interpretation *J* iff it has the same domain as *J* and coincides with *J* on N_{role} and N_{prim}. For a fixed primitive interpretation J, the interpretations *X* based on it are uniquely determined by the tuple $(A_1^{\mathcal{I}}, \ldots, A_k^{\mathcal{I}})$ of the interpretations of the defined concepts in N_{def}. We define

$$Int(\mathcal{J}) := \{\mathcal{I} \mid \mathcal{I} \text{ is an interpretation based on } \mathcal{J}\}.$$

Interpretations based on *J* can be compared by the following ordering, which realizes a pairwise inclusion test between the respective interpretations of the defined concepts: if $\mathcal{I}_1, \mathcal{I}_2 \in$ *Int(J),* then

$$\mathcal{I}_1 \preceq_{\mathcal{J}} \mathcal{I}_2 \text{ iff } A_i^{\mathcal{I}_1} \subseteq A_i^{\mathcal{I}_2} \text{ for all } i, 1 \leq i \leq k.$$

It is easy to see that $\preceq_{\mathcal{J}}$ induces a *complete lattice* on Int *(J),* i.e., every subset *of Jnt(J)* has a least upper bound (lub) and a greatest lower bound (gib). Using *Tarski's fixpoint theorem* LTarski, 1955] for complete lattices, it is not hard to show LNebel, 1991] that, for a given primitive interpretation *J,* there always is a greatest (w.r.t. $\preceq_{\mathcal{J}}$) model of *T* based on *J.* We call this models the *greatest fixpoint model (gfp-model)* of T. *Greatest fixpoint semantics* considers only gfp-models as admissible models.

Definition 2 Let *T* be an $\mathcal{EL}$-TBox and $\mathcal{A}$ an $\mathcal{EL}$-ABox, let *A, B* be defined concepts occurring in T> and *a* an individual name occurring in *A.* Then,

- *A* is subsumed by *B* w.r.t. gfp-semantics ($A \sqsubseteq_{gfp,\mathcal{T}} B$) iff $A^{\mathcal{I}} \subseteq B^{\mathcal{I}}$ holds for all gfp-models $\mathcal{I}$ of $\mathcal{T}$.
- *a* is an instance of *A* w.r.t. gfp-semantics ($\mathcal{A} \models_{gfp,\mathcal{T}} A(a)$) iff $a^{\mathcal{I}} \in A^{\mathcal{I}}$ holds for all models *X* of *A* that are gfp-models of T.

On the level of concept descriptions, the least common subsumer of two concept descriptions C, *D* is the least concept

description E that subsumes both C and D. An extensions of this definition to the level of (possibly cyclic) TBoxes is not completely trivial. In fact, assume that A_1, A_2 are concepts defined in the TBox T. It should be obvious that taking as the lcs of A_1, A_2 the least defined concept B in $\mathcal{T}$ such that $A_1 \sqsubseteq_{gfp,\mathcal{T}} B$ and $A_2 \sqsubseteq_{gfp,\mathcal{T}} B$ is too weak since the lcs would then strongly depend on what other defined concepts are already present in $\mathcal{T}$. However, a second approach (which might look like the obvious generalization of the definition of the lcs in the case of concept descriptions) is also not quite satisfactory. We could say that the lcs of A_1, A_2 is the least concept description C (possibly using defined concepts of $\mathcal{T}$) such that $A_1 \sqsubseteq_{gfp,\mathcal{T}} C$ and $A_2 \sqsubseteq_{gfp,\mathcal{T}} C$. The problem is that this definition does not allow us to use the expressive power of cyclic definitions (with gfp-semantics) when constructing the lcs. For example, consider the TBox $\mathcal{T}$ consisting of the following concept definitions:

$$\mathsf{BlueInode} \equiv \mathsf{Blue} \sqcap \mathsf{Node} \sqcap \exists \mathsf{edge}.\mathsf{BlueInode},$$
$$\mathsf{RedInode} \equiv \mathsf{Red} \sqcap \mathsf{Node} \sqcap \exists \mathsf{edge}.\mathsf{RedInode}.$$

The intended interpretation is similar to the one in Example 1, with the only difference that now nodes may have colors, and we are interested in blue (red) nodes lying on an infinite path consisting of blue (red) nodes. Intuitively, the lcs of BlueInode and RedInode describes nodes lying on an infinite path (without any restriction on their color), i.e., the concept I node from Example 1 should be a definition of this lcs. However, this cannot be expressed by a simple concept description. It requires a new cyclic definition.

Consequently, to obtain the lcs we must allow the original TBox to be extended by new definitions. We say that the TBox $\mathcal{T}_2$ is a *conservative extension* of the TBox $\mathcal{T}_1$ iff $\mathcal{T}_1 \subseteq \mathcal{T}_2$ and $\mathcal{T}_1$ and $\mathcal{T}_2$ have the same primitive concepts and roles. Thus, $\mathcal{T}_2$ may contain new definitions $A \equiv D$, but then D does not introduce new primitive concepts and roles (i.e., all of them already occur in $\mathcal{T}_1$), and A is a new concept name (i.e., A does not occur in $\mathcal{T}_1$). The name "conservative extension" is justified by the fact that the new definitions in $\mathcal{T}_2$ do not influence the subsumption relationships between defined concepts in $\mathcal{T}_1$ (see [Baader, 2002] for a proof).

Lemma 3 *Let $\mathcal{T}_1, \mathcal{T}_2$ be $\mathcal{EL}$-TBoxes such that $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$, and let A, B be defined concepts in $\mathcal{T}_1$ (and thus also in $\mathcal{T}_2$). Then $A \sqsubseteq_{gfp,\mathcal{T}_1} B$ iff $A \sqsubseteq_{gfp,\mathcal{T}_2} B$.*

Definition 4 Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox containing the defined concepts A, B, and let $\mathcal{T}_2$ be a conservative extension of $\mathcal{T}_1$ containing the new defined concept E. Then E in $\mathcal{T}_2$ is a *least common subsumer* of A, B in $\mathcal{T}_1$ *w.r.t. gfp-semantics* (gfp-lcs) iff the following two conditions are satisfied:

1. $A \sqsubseteq_{gfp,\mathcal{T}_2} E$ and $B \sqsubseteq_{gfp,\mathcal{T}_2} E$.
2. If $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_2$ and F a defined concept in $\mathcal{T}_3$ such that $A \sqsubseteq_{gfp,\mathcal{T}_3} F$ and $B \sqsubseteq_{gfp,\mathcal{T}_3} F$, then $E \sqsubseteq_{gfp,\mathcal{T}_3} F$.

In the case of concept descriptions, the lcs is unique up to equivalence, i.e., if E_1 and E_2 are both least common subsumers of the descriptions C, D, then $E_1 \equiv E_2$ (i.e., $E_1 \sqsubseteq E_2$ and $E_2 \sqsubseteq E_1$). In the presence of (possibly cyclic) TBoxes, this uniqueness property also holds (though its formulation is more complicated).

Proposition 5 *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox containing the defined concepts A, B. Assume that $\mathcal{T}_2$ and $\mathcal{T}_2$ are conservative extensions of $\mathcal{T}_1$ such that*

- *the defined concept E in $\mathcal{T}_2$ is a gfp-lcs of A, B in $\mathcal{T}_1$;*
- *the defined concept E' in $\mathcal{T}_2$ is a gfp-lcs of A, B in $\mathcal{T}_1$;*
- *the sets of newly defined concepts in respectively $\mathcal{T}_2$ and $\mathcal{T}_2$ are disjoint.*

For $\mathcal{T}_3 := \mathcal{T}_2 \cup \mathcal{T}_2'$, we have $E \equiv_{gfp,\mathcal{T}_3} E'$ (i.e., $E \sqsubseteq_{gfp,\mathcal{T}_3} E'$ and $E' \sqsubseteq_{gfp,\mathcal{T}_3} E$).

The notion "most specific concept" can be extended in a similar way from concept descriptions to concepts defined in a TBox.

Definition 6 Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox and $\mathcal{A}$ an $\mathcal{EL}$-ABox containing the individual name a, and let $\mathcal{T}_2$ be a conservative extension of $\mathcal{T}_1$ containing the defined concept E. Then E in $\mathcal{T}_2$ is a *most specific concept* of a in $\mathcal{A}$ and $\mathcal{T}_1$ *w.r.t. gfp-semantics* (gfp-msc) iff the following two conditions are satisfied:

1. $\mathcal{A} \models_{gfp,\mathcal{T}_2} E(a)$.
2. If $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_2$ and F a defined concept in $\mathcal{T}_3$ such that $\mathcal{A} \models_{gfp,\mathcal{T}_3} F(a)$, then $E \sqsubseteq_{gfp,\mathcal{T}_3} F$.

Uniqueness up to equivalence of the most specific concept can be formulated and shown like uniqueness of the least common subsumer.

3 Characterizing subsumption

In this section, we recall the characterizations of subsumption w.r.t. gfp-semantics developed in [.Baader, 2003a]. To this purpose, we must represent TBoxes and primitive interpretations by description graphs, and introduce the notion of a simulation on description graphs.

Before we can translate $\mathcal{EL}$-TBoxes into description graphs, we must normalize the TBoxes. In the following, let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, N_{def} the defined concepts of $\mathcal{T}$, N_{prim} the primitive concepts of $\mathcal{T}$, and N_{role} the roles of $\mathcal{T}$.

We say that the $\mathcal{EL}$-TBox $\mathcal{T}$ is *normalized* iff $A \equiv D \in \mathcal{T}$ implies that D is of the form

$$P_1 \sqcap \ldots \sqcap P_m \sqcap \exists r_1.B_1 \sqcap \ldots \sqcap \exists r_\ell.B_\ell,$$

for $m, \ell \geq 0$, $P_1, \ldots, P_m \in N_{prim}$, $r_1, \ldots, r_\ell \in N_{role}$, and $B_1, \ldots, B_\ell \in N_{def}$. If $m = \ell = 0$, then $D = \top$.

As shown in [Baader, 2003a], one can (without loss of generality) restrict the attention to normalized TBox. In the following, we thus assume that all TBoxes are normalized. Normalized £L-TBoxes can be viewed as graphs whose nodes are the defined concepts, which are labeled by sets of primitive concepts, and whose edges are given by the existential restrictions. For the rest of this section, we fix a normalized $\mathcal{EL}$-TBox $\mathcal{T}$ with primitive concepts N_{prim}, defined concepts N_{def}, and roles N_{role}.

Definition 7 An *$\mathcal{EL}$-description graph* is a graph $\mathcal{G} = (V, E, L)$ where

- V is a set of nodes;

- $E \subseteq V \times N_{role} \times V$ is a set of edges labeled by role names;
- $L: V \to 2^{N_{prim}}$ is a function that labels nodes with sets of primitive concepts.

The normalized TBox $\mathcal{T}$ can be translated into the following $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{T} = (N_{def}, E_\mathcal{T}, L_\mathcal{T})$:

- the nodes of $\mathcal{G}_\mathcal{T}$ are the defined concepts of $\mathcal{T}$;
- if A is a defined concept and

$$A \equiv P_1 \sqcap \ldots \sqcap P_m \sqcap \exists r_1.B_1 \sqcap \ldots \sqcap \exists r_\ell.B_\ell$$

its definition in $\mathcal{T}$, then

  - $L_\mathcal{T}(A) = \{P_1, \ldots, P_m\}$, and
  - A is the source of the edges $(A, r_1, B_1), \ldots, (A, r_\ell, B_\ell) \in E_\mathcal{T}$.

Any primitive interpretation $\mathcal{J} = (\Delta^\mathcal{J}, \cdot^\mathcal{J})$ can be ranslated into the following $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{J} = \Delta^\mathcal{J}, E_\mathcal{J}, L_\mathcal{J})$:

- the nodes of $\mathcal{G}_\mathcal{J}$ are the elements of $\Delta^\mathcal{J}$;
- $E_\mathcal{J} := \{(x, r, y) \mid (x, y) \in r^\mathcal{J}\}$;
- $L_\mathcal{J}(x) = \{P \in N_{prim} \mid x \in P^\mathcal{J}\}$ for all $x \in \Delta^\mathcal{J}$.

Conversely, every £L-description graph can be viewed as representing either an *£C-T Box* or a primitive interpretation.

Simulations are binary relations between nodes of two *£L*-description graphs that respect labels and edges in the sense defined below.

Definition 8 Let $\mathcal{G}_i = (V_i, E_i, L_i)$ $(i = 1, 2)$ be two $\mathcal{EL}$-description graphs. The binary relation $Z \subseteq V_1 \times V_2$ is a *simulation* from $\mathcal{G}_1$ to $\mathcal{G}_2$ iff

(S1) $(v_1, v_2) \in Z$ implies $L_1(v_1) \subseteq L_2(v_2)$; and

(S2) if $(v_1, v_2) \in Z$ and $(v_1, r, v_1') \in E_1$, then there exists a node $v_2' \in V_2$ such that $(v_1', v_2') \in Z$ and $(v_2, r, v_2') \in E_2$.

We write $Z: \mathcal{G}_1 \overrightarrow{\sim} \mathcal{G}_2$ to express that Z is a simulation from $\mathcal{G}_1$ to $\mathcal{G}_2$.

It is easy to see that the set of all simulations from *G* to G_2 is closed under arbitrary unions. Consequently, there always exists a greatest simulation from G_1 to G_2- If G_1, $G2$ are finite, then this greatest simulation can be computed in polynomial time [Henzinger *et al,* 1995]. As an easy consequence of this fact, the following proposition is proved in [Baader, 2003a].

Proposition 9 *Let G_1, G_2 be two finite EL-description graphs, v_1 a node of G\ and V_2 a node of G_2- Then we can decide in polynomial time whether there is a simulation $Z: \mathcal{G}_1 \overrightarrow{\sim} \mathcal{G}_2$ such that $(v_1, v_2) \in Z$.*

Subsumption w.r.t. gfp-semantics corresponds to the existence of a simulation relation such that the subsumee simulates the subsumer:

Theorem 10 *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and A, B defined concepts in $\mathcal{T}$. Then the following are equivalent:*

1. $A \sqsubseteq_{gfp,\mathcal{T}} B$.
2. *There is a simulation Z: Gr ~ GT such that* (B, *A)* £ Z.

The theorem together with Proposition 9 shows that subsumption w.r.t. gfp-semantics in £C is tractable. The proof of the theorem given in [Baader, 2003a] depends on a characterization of when an individual of a gfp-model belongs to a defined concept in this model.

Proposition 11 *Let $\mathcal{J}$ be a primitive interpretation and $\mathcal{I}$ the gfp-model of $\mathcal{T}$ based on $\mathcal{J}$. Then the following are equivalent for any $A \in N_{def}$ and $x \in \Delta^\mathcal{J}$:*

1. $x \in A^\mathcal{I}$.
2. *There is a simulation $Z: \mathcal{G}_\mathcal{T} \overrightarrow{\sim} \mathcal{G}_\mathcal{J}$ such that $(A, x) \in Z$.*

This proposition is also important in the proof of correctness of our characterization of the instance problem (Theorem 17).

4 Computing the lcs

We will show how the characterization of subsumption w.r.t. gfp-semantics given in Theorem 10 can be used to characterize the gfp-lcs (see [Baader, 2002] for more details and proofs). Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox, let $\mathcal{G}_{\mathcal{T}_1} = (N_{def}, E_{\mathcal{T}_1}, L_{\mathcal{T}_1})$ be the corresponding description graph, and let A, B be defined concepts in $\mathcal{T}_1$ (i.e., elements of N_{def}). In principle, the lcs of A, B in $\mathcal{T}_1$ is defined in a TBox whose description graph is the product of $\mathcal{G}_{\mathcal{T}_1}$ with itself.

Definition 12 Let $\mathcal{G}_1 = (V_1, E_1, L_1)$ and $\mathcal{G}_2 = (V_2, E_2, L_2)$ be two description graphs. Their *product* is the description graph $\mathcal{G}_1 \times \mathcal{G}_2 := (V, E, L)$ where

- $V := V_1 \times V_2$;
- $E := \{((v_1, v_2), r, (v_1', v_2')) \mid (v_1, r, v_1') \in E_1 \wedge (v_2, r, v_2') \in E_2\}$;
- $L(v_1, v_2) := L_1(v_1) \sqcap L_2(v_2)$.

The description graph $\mathcal{G}_{\mathcal{T}_1} \times \mathcal{G}_{\mathcal{T}_1}$ yields a TBox $\mathcal{T}$ such that $\mathcal{G}_\mathcal{T} = \mathcal{G}_{\mathcal{T}_1} \times \mathcal{G}_{\mathcal{T}_1}$. Now, $\mathcal{T}_2 := \mathcal{T}_1 \cup \mathcal{T}$ is a conservative extension of $\mathcal{T}_1$. In fact, $\mathcal{G}_{\mathcal{T}_1} \times \mathcal{G}_{\mathcal{T}_1}$ is based on the same primitive concepts and roles as $\mathcal{G}_{\mathcal{T}_1}$, and the set of defined concepts in $\mathcal{T}$ is $N_{def} \times N_{def}$, which is disjoint from N_{def}.

Lemma 13 *(A, B) in $\mathcal{T}_2$ is the gfp-lcs of A and B in $\mathcal{T}_1$.*

Computing the (binary) product of two $\mathcal{EL}$-description graphs can obviously be done in polynomial time, and thus the gfp-lcs can be computed in polynomial time.

Theorem 14 *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox, and let A, B be defined concepts in $\mathcal{T}_1$. Then the gfp-lcs of A, B in $\mathcal{T}_1$ always exists, and it can be computed in polynomial time.*

5 Instance and most specific concepts

One motivation for considering cyclic terminologies in £L is the fact that the most specific concept of an ABox individual need not exist in £C (without cyclic terminologies). An example is the simple cyclic ABox $A := \{r(b,b)\}$, where b has no most specific concept, i.e., there is no least £L-concept description D such that b is an instance of D w.r.t. A [Kiisters and Molitor, 2001]. However, if one allows for cyclic TBoxes with gfp-semantics, then the defined concept B with $B \equiv \exists r.B$ is such a most specific concept.

5.1 The instance problem w.r.t. gfp-semantics

Let $\mathcal{T}$ be a normalized $\mathcal{EL}$-TBox and $\mathcal{A}$ an $\mathcal{EL}$-ABox. In the following, we assume that $\mathcal{T}$ is fixed and that all instance problems for $\mathcal{A}$ are considered w.r.t. this TBox.

In this setting, $\mathcal{A}$ can be translated into an $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{A}$ by viewing $\mathcal{A}$ as a graph and extending it appropriately by the graph $\mathcal{G}_\mathcal{T}$ associated with $\mathcal{T}$. The idea is then that the characterization of the instance problem should be similar to the statement of Proposition 11: the individual a is an instance of A in $\mathcal{A}$ iff there is a simulation $Z: \mathcal{G}_\mathcal{T} \overrightarrow{\sim} \mathcal{G}_\mathcal{A}$ such that $(A, a) \in Z$.

Before giving an exact definition of $\mathcal{G}_\mathcal{A}$, we consider an example that demonstrates that a too simple-minded realization of this idea does not work. Let

$$\begin{aligned} \mathcal{A} &:= \{A(a), P(a)\} \text{ and} \\ \mathcal{T} &:= \{A \equiv \exists r.A,\ B \equiv P \sqcap \exists r.B\}. \end{aligned}$$

The ABox $\mathcal{A}$ itself can be viewed as an $\mathcal{EL}$-description graph consisting of a single node a with label $\{P\}$. Since $A \equiv \exists r.A$ is in $\mathcal{T}$ and $A(a)$ is in $\mathcal{A}$, we extend this graph by an r-loop from a to a. Figure 1 shows the graph $\mathcal{G}$ obtained this way as well as the $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{T}$ corresponding to $\mathcal{T}$.

Obviously, there is a simulation $Z: \mathcal{G}_\mathcal{T} \overrightarrow{\sim} \mathcal{G}$ such that $(B, a) \in Z$. However, it is easy to see that a is not an instance of B. The reason for this problem is that node labels and edges in $\mathcal{G}_\mathcal{T}$ state facts that must hold for *all* individuals that are instances of the defined concept labeling a given node whereas assertions of the ABox make statements about properties of *particular* named individuals. The construction of $\mathcal{G}$ in the above example mixes these different things, and thus leads to unfounded conclusions.

In order to separate edges and labels coming from ABox assertions from the ones coming from TBox definitions, we do not "identify" the node a with the node A if $A(a)$ belongs to $\mathcal{A}$ (as done in the construction of $\mathcal{G}$ above). Instead, we do a "one-step expansion" of the definition of A. The right-most graph in Figure 1 shows the graph $\mathcal{G}_\mathcal{A}$ obtained this way in our example. Obviously, there is no simulation $Z: \mathcal{G}_\mathcal{T} \overrightarrow{\sim} \mathcal{G}_\mathcal{A}$ such that $(B, a) \in Z$.

Below, we give a formal definition of the $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{A}$ associated with the ABox $\mathcal{A}$ and the TBox $\mathcal{T}$ in the general case.

Definition 15 Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{A}$ an $\mathcal{EL}$-ABox, and $\mathcal{G}_\mathcal{T} = (V, E, L)$ be the $\mathcal{EL}$-description graph associated with $\mathcal{T}$. The $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{A} = (V_\mathcal{A}, E_\mathcal{A}, L_\mathcal{A})$ associated with $\mathcal{A}$ and $\mathcal{T}$ is defined as follows:

- The nodes of $\mathcal{G}_\mathcal{A}$ are the individual names occurring in $\mathcal{A}$ together with the defined concepts of $\mathcal{T}$, i.e., $V_\mathcal{A} := V \cup \{a \mid a \text{ is an individual name in } \mathcal{A}\}$.
- The edges of $\mathcal{G}_\mathcal{A}$ are the edges of $\mathcal{G}$, the role assertions of $\mathcal{A}$, and additional edges linking the ABox individuals with defined concepts:

$$\begin{aligned} E_\mathcal{A} := E \cup \{(a, r, b) \mid r(a, b) \in \mathcal{A}\} \cup \\ \{(a, r, B) \mid A(a) \in \mathcal{A} \text{ and } (A, r, B) \in E\}. \end{aligned}$$

- if $u \in V_\mathcal{A}$ is a defined concept, then it inherits its label from $\mathcal{G}_\mathcal{T}$, i.e., $L_\mathcal{A}(u) := L(u)$ if $u \in V$. Otherwise, $u \in V_\mathcal{A} \setminus V$ is an ABox individual, and then its label is derived from the concept assertions for u in $\mathcal{A}$:

$$L_\mathcal{A}(u) := \{P \mid P(u) \in \mathcal{A}\} \cup \bigcup_{A(u) \in \mathcal{A}} L(A).$$

Here P denotes primitive and A denotes defined concepts.

Before we can characterize the instance problem via the existence of certain simulation relations from $\mathcal{G}_\mathcal{T}$ to $\mathcal{G}_\mathcal{A}$, we must characterize under what conditions a gfp-model of $\mathcal{T}$ is a model of $\mathcal{A}$. In the following we assume that primitive interpretations $\mathcal{J}$ also interpret ABox individuals. We say that the simulation $Z: \mathcal{G}_\mathcal{A} \overrightarrow{\sim} \mathcal{G}_\mathcal{J}$ *respects ABox individuals* iff $\{x \mid (a, x) \in Z\} = \{a^\mathcal{J}\}$ for all individual names a occurring in $\mathcal{A}$.

Proposition 16 *Let $\mathcal{J}$ be a primitive interpretation and $\mathcal{I}$ the gfp-model of $\mathcal{T}$ based on $\mathcal{J}$. Then the following are equivalent:*

1. *$\mathcal{I}$ is a model of $\mathcal{A}$.*
2. *There is a simulation $Z: \mathcal{G}_\mathcal{A} \overrightarrow{\sim} \mathcal{G}_\mathcal{J}$ that respects ABox individuals.*

The following characterization of the instance problem is an easy consequence of this proposition and Proposition 11.

Theorem 17 *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{A}$ an $\mathcal{EL}$-ABox, A a defined concept in $\mathcal{T}$ and 'a' an individual name occurring in $\mathcal{A}$. Then the following are equivalent:*

1. *$\mathcal{A} \models_{gfp,\mathcal{T}} A(a)$.*
2. *There is a simulation $Z: \mathcal{G}_\mathcal{T} \overrightarrow{\sim} \mathcal{G}_\mathcal{A}$ such that $(A, a) \in Z$.*

The theorem together with Proposition 9 shows that the instance problem w.r.t. gfp-semantics in $\mathcal{EL}$ is tractable.

Corollary 18 *The instance problem w.r.t. gfp-semantics in $\mathcal{EL}$ can be decided in polynomial time.*

5.2 Computing the gfp-msc

Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox and $\mathcal{A}$ an $\mathcal{EL}$-ABox containing the individual name a. Let $\mathcal{G}_\mathcal{A} = (V_\mathcal{A}, E_\mathcal{A}, L_\mathcal{A})$ be the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}_1$, as introduced in Definition 15. In order to obtain the gfp-msc of a, we view $\mathcal{G}_\mathcal{A}$ as the $\mathcal{EL}$-description graph of an $\mathcal{EL}$-TBox $\mathcal{T}_2$, i.e., let $\mathcal{T}_2$ be the TBox such that $\mathcal{G}_\mathcal{A} = \mathcal{G}_{\mathcal{T}_2}$. By the definition of $\mathcal{G}_\mathcal{A}$, the defined concepts of $\mathcal{T}_2$ are the defined concepts of $\mathcal{T}_1$ together with the individual names occurring in $\mathcal{A}$. It is easy to show that $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$. To avoid confusion we will denote the defined concept in $\mathcal{T}_2$ corresponding to the individual name b in $\mathcal{A}$ by C_b.

Using the results of the previous subsection, we can show [Baader, 2002] that C_a is the gfp-msc of a.

Lemma 19 *The defined concept C_a in $\mathcal{T}_2$ is the gfp-msc of 'a' in $\mathcal{A}$ and $\mathcal{T}_1$.*

Given $\mathcal{T}_1$ and $\mathcal{A}$, the graph $\mathcal{G}_\mathcal{A}$ can obviously be computed in polynomial time, and thus the gfp-msc can be computed in polynomial time.

Theorem 20 *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox and $\mathcal{A}$ an $\mathcal{EL}$-ABox containing the individual name 'a'. Then the gfp-msc of 'a' in $\mathcal{T}_1$ and $\mathcal{A}$ always exists, and it can be computed in polynomial time.*

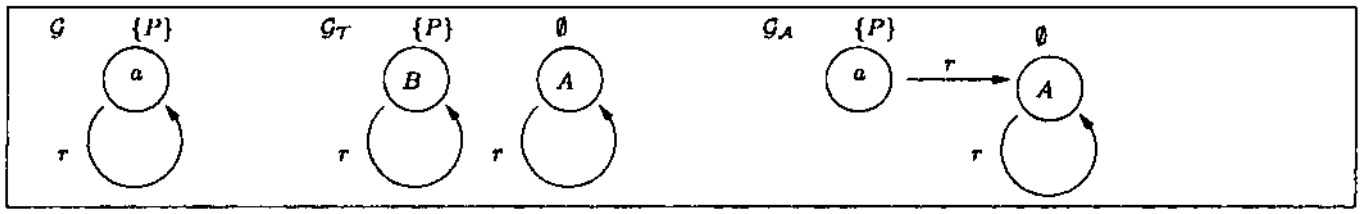

Figure 1: The $\mathcal{EL}$-description graphs $\mathcal{G}$, $\mathcal{G}_\mathcal{T}$, and $\mathcal{G}_\mathcal{A}$ of our example.

6 Conclusion

In [Baader, 2003a] we have shown that subsumption in $\mathcal{EL}$ remains polynomial if one allows for cyclic terminologies with greatest fixpoint (gfp) semantics. In Section 5.1 of this paper we have complemented this result by showing that the instance problem in $\mathcal{EL}$ with cyclic terminologies interpreted with gfp-semantics is also polynomial. Thus, all the standard inferences in $\mathcal{EL}$ remain polynomial if one allows for cyclic terminologies with gfp-semantics. Our main motivation for considering cyclic terminologies with gfp-semantics in $\mathcal{EL}$ was that the most specific concept of an ABox-individual then always exists. In fact, we have shown in this paper that both the least common subsumer (les) and the most specific concept (msc) can be computed in polynomial time in $\mathcal{EL}$ with cyclic terminologies interpreted with gfp-semantics. Thus, also two of the most important non-standard inferences in DLs [Kusters, 2001] remain polynomial in this context.

It should be noted that there are indeed applications where the expressive power of the small DL $\mathcal{EL}$ appears to be sufficient. In fact, SNOMED, the Systematized Nomenclature of Medicine [Cote *et al,* 1993] uses $\mathcal{EL}$ [Spackman, 2000; 2001].

Subsumption [Baader, 2003a] and the instance problem [Baader, 2003b] are also polynomial w.r.t. descriptive semantics. For the les, descriptive semantics is not that well-behaved: in [Baader, 2003] we have shown that w.r.t. descriptive semantics the les need not exist in $\mathcal{EL}$ with cyclic terminologies. In addition, we could only give a sufficient condition for the existence of the les. If this condition applies, then the les can be computed in polynomial time. In [Baader, 2003b] similar results are shown for the msc w.r.t. descriptive semantics.

One problem left for future research is the question of how to obtain a decidable characterization of the cases in which the les (msc) exists w.r.t. descriptive semantics, and to determine whether in these cases it can always be computed in polynomial time.

References

[Baader, 2002] F. Baader. Least common subsumers, most specific concepts, and role-value-maps in a description logic with existential restrictions and terminological cycles. LTCS-Report 02-07, TU Dresden, 2002. See http://lat.inf.tu-dresden.de/research/reports.html.

[Baader, 2003] F. Baader. Computing the least common subsumer in the description logic *£L* w.r.t. terminological cycles with descriptive semantics. In *Proc. ICCS'03,* Springer LNAI, 2003.

[Baader, 2003a] F. Baader. Terminological cycles in a description logic with existential restrictions. In *Proc. IJCAI'03,2003.*

[Baader, 2003b] Franz Baader. The instance problem and the most specific concept in the description logic *£C* w.r.t. terminological cycles with descriptive semantics. LTCS-Report 03-01, TU Dresden, 2003. See http://lat.inf.tu-dresden.de/research/reports.html.

[Baader and Kusters, 1998] F. Baader and R. Kiisters. Computing the least common subsumer and the most specific concept in the presence of cyclic *ALN-concept* descriptions. In *Proc. KI'98,* Springer LNAI 1504, 1998.

[Baader *et al,* 1999] F. Baader, R. Kusters, and R. Molitor. Computing least common subsumers in description logics with existential restrictions. *In Proc. IJCAI'99,* 1999.

[Cote *et al,* 1993] R. Cote, D. Rothwell, J. Palotay, R. Beckett, and L. Brochu. The systematized nomenclature of human and veterinary medicine. Technical report, SNOMED International, Northfield, IL: College of American Pathologists, 1993.

[Henzinger et a/., 1995] M. R. Henzinger, T. A. Henzinger, and P. W. Kopke. Computing simulations on finite and infinite graphs. In *36th Annual Symposium on Foundations of Computer Science,* 1995. IEEE Computer Society Press.

[Kusters, 2001] R. Kusters. *Non-standard Inferences in Description Logics,* Springer LNAI 2100, 2001.

[Kusters and Molitor, 2001] R. Kusters and R. Molitor. Approximating most specific concepts in description logics with existential restrictions. In *Proc. KI 2001,* Springer LNAI 2174,2001.

[Nebel, 1991] B. Nebel. Terminological cycles: Semantics and computational properties. In J. F. Sowa, editor, *Principles of Semantic Networks.* Morgan Kaufmann, 1991.

[Spackman, 2000] K. Spackman. Managing clinical terminology hierarchies using algorithmic calculation of subsumption: Experience with SNOMED-RT. *J. of the American Medical Informatics Association,* 2000. Fall Symposium Special Issue.

[Spackman, 2001] K. Spackman. Normal forms for description logic expressions of clinical concepts in SNOMED RT. *J. of the American Medical Informatics Association,* 2001. Symposium Supplement.

[Tarski, 1955] A. Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics,* 5, 1955.