The instance problem and the most specific concept in the description logic $\mathcal{EL}$ w.r.t. terminological cycles with descriptive semantics

Franz Baader*

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

Abstract. Previously, we have investigated both standard and non-standard inferences in the presence of terminological cycles for the description logic $\mathcal{EL}$, which allows for conjunctions, existential restrictions, and the top concept. The present paper is concerned with two problems left open by this previous work, namely the instance problem and the problem of computing most specific concepts w.r.t. descriptive semantics, which is the usual first-order semantics for description logics. We will show that—like subsumption—the instance problem is polynomial in this context. Similar to the case of the least common subsumer, the most specific concept w.r.t. descriptive semantics need not exist, but we are able to characterize the cases in which it exists and give a decidable sufficient condition for the existence of the most specific concept. Under this condition, it can be computed in polynomial time.

1 Introduction

Early description logic (DL) systems allowed the use of value restrictions ($\forall r.C$), but not of existential restrictions ($\exists r.C$). Thus, one could express that all children are male using the value restriction $\forall \mathsf{child}.\mathsf{Male}$, but not that someone has a son using the existential restriction $\exists \mathsf{child}.\mathsf{Male}$. The main reason was that, when clarifying the logical status of property arcs in semantic networks and slots in frames, the decision was taken that arcs/slots should be read as value restrictions (see, e.g., [12]). Once one considers more expressive DLs allowing for full negation, existential restrictions come in as the dual of value restrictions [14]. Thus, for historical reasons, DLs that allow for existential, but not for value restrictions, were until recently mostly unexplored.

The recent interest in such DLs has at least two reasons. On the one hand, there are indeed applications where DLs without value restrictions appear to be sufficient. For example, SNOMED, the Systematized Nomenclature of Medicine [16, 15] employs the DL $\mathcal{EL}$, which allows for conjunctions, existential restrictions, and the top concept. On the other hand, non-standard inferences in DLs [11], like computing the least common subsumer, often make sense only for DLs

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

that do not allow for full negation. Thus, the decision of whether to use DLs with value restrictions or with existential restrictions becomes again relevant.

Non-standard inferences were introduced to support building and maintaining large DL knowledge bases. For example, 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 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 most specific concept (msc) of a given ABox individual need not exist in languages allowing for existential restrictions or number restrictions. For the DL $\mathcal{ALN}$ (which allows for conjunctions, value restrictions, and number restrictions), it was shown in [6] that the most specific concept always exists if one adds cyclic concept definitions with greatest fixpoint semantics. If one wants to use this approach for the bottom-up construction of knowledge bases, then one must also be able to solve the standard inferences (the subsumption and the instance problem) and to compute the least common subsumer in the presence of cyclic concept definitions. Thus, in order to adapt the approach also to the DL $\mathcal{EL}$, the impact on both standard and non-standard inferences of cyclic definitions in this DL had to be investigated first.

The paper [5] considers cyclic terminologies in $\mathcal{EL}$ w.r.t. the three types of semantics (greatest fixpoint, least fixpoint, and descriptive semantics) introduced by Nebel [13], and shows that the subsumption problem can be decided in polynomial time in all three cases. This is in strong contrast to the case of DLs with value restrictions. Even for the small DL $\mathcal{FL}_0$ (which allows for conjunctions and value restrictions only), adding cyclic terminologies increases the complexity of the subsumption problem from polynomial (for concept descriptions) to PSPACE [1]. The main tool in the investigation of cyclic definitions in $\mathcal{EL}$ is a characterization of subsumption through the existence of so-called simulation relations, which can be computed in polynomial time [9]. The results in [5] also show that cyclic definitions with least fixpoint semantics are not interesting in $\mathcal{EL}$. For this reason, all the extensions of these results mentioned below are concerned with greatest fixpoint (gfp) and descriptive semantics only.

The characterization of subsumption in $\mathcal{EL}$ w.r.t. gfp-semantics through the existence of certain simulation relations on the graph associated with the terminology is used in [4] to characterize the least common subsumer via the product of this graph with itself. This shows that, w.r.t. gfp semantics, the lcs always exists, and the binary lcs can be computed in polynomial time. (The n-ary lcs may grow exponentially even in $\mathcal{EL}$ without cyclic terminologies [7].) For cyclic terminologies in $\mathcal{EL}$ with descriptive semantics, the lcs need not exist. In [2], possible candidates P_k ($k \geq 0$) for the lcs are introduced, and it is shown that the lcs exists iff one of these candidates is the lcs. In addition, a sufficient condition for the existence of the lcs is given, and it is shown that, under this condition, the lcs can be computed in polynomial time.

In [4], the characterization of subsumption w.r.t. gfp-semantics is also extended to the instance problem in $\mathcal{EL}$. This is then used to show that, w.r.t. gfp-semantics, the instance problem in $\mathcal{EL}$ can be decided in polynomial time and that the msc in $\mathcal{EL}$ always exists, and can be computed in polynomial time.

Given the positive results for gfp-semantics regarding both standard inferences (subsumption and instance) and non-standard inferences (lcs and msc), one might be tempted to restrict the attention to gfp-semantics. However, existing DL systems like FaCT [10] and RACER [8] allow for terminological cycles (even more general inclusion axioms), but employ descriptive semantics. In some cases it may be desirable to use a semantics that is consistent with the one employed by these systems even if one works with a DL that is considerably less expressive than then one available in them. For example, non-standard inferences that support building DL knowledge bases are often restricted to rather inexpressive DLs (either because they do not make sense for more expressive DLs or because they can currently only be handled for such DLs). Nevertheless, it may be desirable that the result of these inferences (like the msc or the lcs) is again in a format that is accepted by systems like FaCT and RACER. This is not the case if the msc algorithm produces a cyclic terminology that must be interpreted with gfp-semantics.

The subsumption problem and the problem of computing least common subsumers in $\mathcal{EL}$ w.r.t cyclic terminologies with descriptive semantics have already been tackled in [5] and [2], respectively. In the present paper we address the instance problem and the problem of computing the most specific concept in this setting. We will show that the instance problem is polynomial also in this context. Unfortunately, the most specific concept w.r.t descriptive semantics need not exist, but—similar to the case of the least common subsumer—we are able to characterize the cases in which it exists and give a decidable sufficient condition for the existence of the most specific concept. Under this condition, it can be computed in polynomial time.

2 Cyclic terminologies and most specific concepts in $\mathcal{EL}$

Concept descriptions are inductively defined with the help of a set of *constructors*, starting with a set N_C of *concept names* and a set N_R of *role names*. The constructors determine the expressive power of the DL. In this paper, we restrict the attention to the DL $\mathcal{EL}$, whose concept descriptions are formed using the constructors top-concept ($\top$), conjunction ($C \sqcap D$), and existential restriction ($\exists r.C$). The semantics of $\mathcal{EL}$-concept descriptions is defined in terms of an *interpretation* $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$. The domain $\Delta^\mathcal{I}$ of $\mathcal{I}$ is a non-empty set of individuals and the interpretation function $\cdot^\mathcal{I}$ maps each concept name $A \in N_C$ to a subset

name of constructor	Syntax	Semantics
concept name $A \in N_C$	A	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
role name $r \in N_R$	r	$r^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
top-concept	$\top$	$\Delta^{\mathcal{I}}$
conjunction	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
existential restriction	$\exists r.C$	$\{x \in \Delta^{\mathcal{I}} \mid \exists y : (x, y) \in r^{\mathcal{I}} \wedge y \in C^{\mathcal{I}}\}$
concept definition	$A \equiv D$	$A^{\mathcal{I}} = D^{\mathcal{I}}$
individual name $a \in N_I$	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 $\mathcal{EL}$

$A^{\mathcal{I}}$ 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 TBoxes 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 $\mathcal{I}$ is a model of the TBox $\mathcal{T}$ iff it satisfies all its concept definitions, i.e., $A^{\mathcal{I}} = D^{\mathcal{I}}$ for all definitions $A \equiv D$ in $\mathcal{T}$.

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 $\mathcal{I}$ is a model of the ABox $\mathcal{A}$ iff it satisfies all its assertions, i.e., $a^{\mathcal{I}} \in A^{\mathcal{I}}$ for all concept assertions $A(a)$ in $\mathcal{A}$ and $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ for all role assertions $r(a, b)$ in $\mathcal{A}$. The interpretation $\mathcal{I}$ is a model of the ABox $\mathcal{A}$ together with the TBox $\mathcal{T}$ iff it is a model of both $\mathcal{T}$ and $\mathcal{A}$.

The semantics of (possibly cyclic) $\mathcal{EL}$-TBoxes we have defined above is called *descriptive semantic* by Nebel [13]. For some applications, it is more appropriate to interpret cyclic concept definitions with the help of an appropriate fixpoint semantics. However, in this paper we restrict our attention to descriptive semantics (see [5,4] for definitions and results concerning cyclic terminologies in $\mathcal{EL}$ with fixpoint semantics).

Definition 1. *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and $\mathcal{A}$ an $\mathcal{EL}$-ABox, let C, D be concept descriptions (possibly containing defined concepts of $\mathcal{T}$), and a an individual name occurring in $\mathcal{A}$. Then,*

- *C is subsumed by D w.r.t. descriptive semantics ($C \sqsubseteq_{\mathcal{T}} D$) iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ holds for all models $\mathcal{I}$ of $\mathcal{T}$.*

– *a is an instance of C w.r.t. descriptive semantics ($\mathcal{A} \models_{\mathcal{T}} C(a)$) iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$ holds for all models $\mathcal{I}$ of $\mathcal{T}$ together with $\mathcal{A}$.*

On the level of concept descriptions, the most specific concept of a given ABox individual a is the least concept description E (of the DL under consideration) that has a as an instance. An extensions of this definition to the level of (possibly cyclic) TBoxes is not completely trivial. In fact, assume that a is an individual in the ABox $\mathcal{A}$ and that $\mathcal{T}$ is a TBox. It should be obvious that taking as the msc of a the least defined concept A in $\mathcal{T}$ such that $\mathcal{A} \models_{\mathcal{T}} A(a)$ is too weak since the lcs would then strongly depend on the defined concepts that are already present in $\mathcal{T}$. However, a second approach (which might look like the obvious generalization of the definition of the msc in the case of concept descriptions) is also not quite satisfactory. We could say that the msc of a is the least concept description C (possibly using defined concepts of $\mathcal{T}$) such that $\mathcal{A} \models_{\mathcal{T}} C(a)$. The problem is that this definition does not allow us to use the expressive power of cyclic definitions when constructing the msc.

To avoid this problem, we allow the original TBox to be extended by new definitions when constructing the msc. 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 [4] for details).

Definition 2. *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.*[1] *Then E in $\mathcal{T}_2$ is a* most specific concept *of a in $\mathcal{A}$ and $\mathcal{T}_1$* w.r.t. descriptive semantics *(msc) iff the following two conditions are satisfied:*

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

In the case of concept descriptions, the msc is unique up to equivalence. In the presence of (possibly cyclic) TBoxes, this uniqueness property also holds, though its formulation is more complicated (see [4] for details).

3 Characterizing subsumption in cyclic $\mathcal{EL}$-TBoxes

In this section, we recall the characterizations of subsumption w.r.t. descriptive semantics developed in [5]. To this purpose, we must represent TBoxes by description graphs, and introduce the notion of a simulation on description graphs.

[1] Without loss of generality we assume that the msc is given by a defined concept rather than a concept description since one can always introduce an appropriate definition for the description. For the same reason, we can in the following restrict the instance problem and the subsumption problem to defined concepts.

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 [5], one can (without loss of generality) restrict the attention to normalized TBox. In the following, we thus assume that all TBoxes are normalized. Normalized $\mathcal{EL}$-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 3. *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;*
- *$L\colon V \to 2^{N_{prim}}$ 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 $(A, r_1, B_1), \ldots, (A, r_\ell, B_\ell) \in E_\mathcal{T}$.*

Simulations are binary relations between nodes of two $\mathcal{EL}$-description graphs that respect labels and edges in the sense defined below.

Definition 4. *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 $Z\colon \mathcal{G}_1 \vec{\sim} \mathcal{G}_2$ to express that Z is a simulation from $\mathcal{G}_1$ to $\mathcal{G}_2$.

W.r.t. gfp-semantics, A is subsumed by B iff there is a simulation $Z\colon \mathcal{G}_\mathcal{T} \vec{\sim} \mathcal{G}_\mathcal{T}$ such that $(B, A) \in Z$ (see [5]). W.r.t. descriptive semantics, the simulation Z must satisfy some additional properties for this equivalence to hold. To define these properties, we must introduce some notation.

$$
\begin{array}{c}
B = B_0 \xrightarrow{r_1} B_1 \xrightarrow{r_2} B_2 \xrightarrow{r_3} B_3 \xrightarrow{r_4} \cdots \\
Z\downarrow \quad Z\downarrow \quad Z\downarrow \quad Z\downarrow \\
A = A_0 \xrightarrow{r_1} A_1 \xrightarrow{r_2} A_2 \xrightarrow{r_3} A_3 \xrightarrow{r_4} \cdots
\end{array}
$$

Fig. 1. A (B, A)-simulation chain.

$$
\begin{array}{c}
B = B_0 \xrightarrow{r_1} B_1 \xrightarrow{r_2} \cdots \xrightarrow{r_{n-1}} B_{n-1} \xrightarrow{r_n} B_n \\
Z\downarrow \quad Z\downarrow \qquad\qquad Z\downarrow \\
A = A_0 \xrightarrow{r_1} A_1 \xrightarrow{r_2} \cdots \xrightarrow{r_{n-1}} A_{n-1}
\end{array}
$$

Fig. 2. A partial (B, A)-simulation chain.

Definition 5. *The path p_1: $B = B_0 \xrightarrow{r_1} B_1 \xrightarrow{r_2} B_2 \xrightarrow{r_3} B_3 \xrightarrow{r_4} \cdots$ in $\mathcal{G}_\mathcal{T}$ is Z-*simulated *by the path p_2: $A = A_0 \xrightarrow{r_1} A_1 \xrightarrow{r_2} A_2 \xrightarrow{r_3} A_3 \xrightarrow{r_4} \cdots$ in $\mathcal{G}_\mathcal{T}$ iff $(B_i, A_i) \in Z$ for all $i \geq 0$. In this case we say that the pair (p_1, p_2) is a* (B, A)-simulation chain w.r.t. *Z (see Figure 1).*

If $(B, A) \in Z$, then (S2) of Definition 4 implies that, for every infinite path p_1 starting with $B_0 := B$, there is an infinite path p_2 starting with $A_0 := A$ such that p_1 is Z-simulated by p_2. In the following we construct such a simulating path step by step. The main point is, however, that the decision which concept A_n to take in step n should depend only on the partial (B, A)-simulation chain already constructed, and *not* on the parts of the path p_1 not yet considered.

Definition 6. *A* partial (B, A)-simulation chain *is of the form depicted in Figure 2. A* selection function *S for A, B and Z assigns to each partial (B, A)-simulation chain of this form a defined concept A_n such that (A_{n-1}, r_n, A_n) is an edge in $\mathcal{G}_\mathcal{T}$ and $(B_n, A_n) \in Z$. Given a path $B = B_0 \xrightarrow{r_1} B_1 \xrightarrow{r_2} B_2 \xrightarrow{r_3} B_3 \xrightarrow{r_4} \cdots$ and a defined concept A such that $(B, A) \in Z$, one can use a selection function S for A, B and Z to construct a Z-simulating path. In this case we say that the resulting (B, A)-simulation chain is* S-selected.

Definition 7. *Let A, B be defined concepts in $\mathcal{T}$, and Z: $\mathcal{G}_\mathcal{T} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{T}$ a simulation with $(B, A) \in Z$. Then Z is called* (B, A)-synchronized *iff there exists a selection function S for A, B and Z such that the following holds: for every infinite S-selected (B, A)-simulation chain of the form depicted in Figure 1 there exists an $i \geq 0$ such that $A_i = B_i$.*

We are now ready to state the characterization of subsumption w.r.t. descriptive semantics from [5].

Theorem 1. *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, and A, B defined concepts in $\mathcal{T}$. Then the following are equivalent:*

1. *$A \sqsubseteq_\mathcal{T} B$.*

2. There is a (B,A)-synchronized simulation $Z\colon \mathcal{G}_\mathcal{T} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{T}$ such that $(B,A) \in Z$.

In [5] it is also shown that, for a given $\mathcal{EL}$-TBox $\mathcal{T}$ and defined concepts A, B in $\mathcal{T}$, the existence of a (B,A)-synchronized simulation $Z\colon \mathcal{G}_\mathcal{T} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{T}$ with $(B,A) \in Z$ can be decided in polynomial time, which shows that the subsumption w.r.t. descriptive semantics in $\mathcal{EL}$ is tractable.

4 The instance problem

Assume that $\mathcal{T}$ is an $\mathcal{EL}$-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 Theorem 1: the individual a is an instance of A in $\mathcal{A}$ and $\mathcal{T}$ iff there is an (A,a)-synchronized simulation $Z\colon \mathcal{G}_\mathcal{T} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{A}$ such that $(A,a) \in Z$.[2] The formal definition of the $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{A}$ associated with the ABox $\mathcal{A}$ and the TBox $\mathcal{T}$ given below was also used in [4] to characterize the instance problem in $\mathcal{EL}$ w.r.t. gfp-semantics.

Definition 8. *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{A}$ an $\mathcal{EL}$-ABox, and $\mathcal{G}_\mathcal{T} = (V_\mathcal{T}, E_\mathcal{T}, L_\mathcal{T})$ 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.,*

$$V_\mathcal{A} := V_\mathcal{T} \cup \{a \mid a \text{ is an individual name occurring in } \mathcal{A}\};$$

- *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:*

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

- *if $u \in V_\mathcal{A}$ is a defined concept, then it inherits its label from $\mathcal{G}_\mathcal{T}$, i.e.,*

$$L_\mathcal{A}(u) := L_\mathcal{T}(u) \quad \text{if } u \in V_\mathcal{T};$$

  otherwise, u is an ABox individual, and then its label is derived from the concept assertions for u in $\mathcal{A}$. In the following, let P denote primitive and A denote defined concepts.

$$L_\mathcal{A}(u) := \{P \mid P(u) \in \mathcal{A}\} \cup \bigcup_{A(u) \in \mathcal{A}} L_\mathcal{T}(A) \quad \text{if } u \in V_\mathcal{A} \setminus V_\mathcal{T}.$$

[2] The actual characterization of the instance problem turns out to be somewhat more complex, but for the moment the above is sufficient to gives the right intuition.

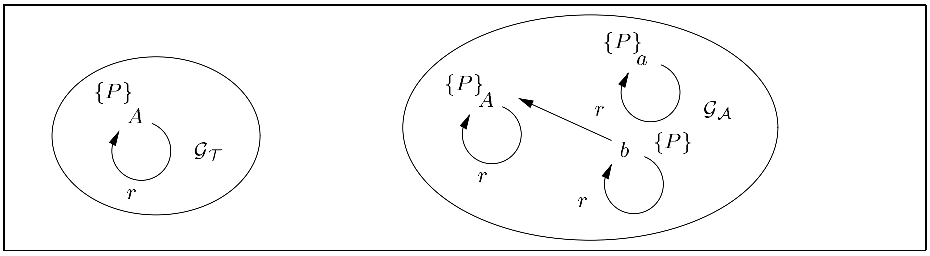

Fig. 3. The $\mathcal{EL}$-description graphs $\mathcal{G}_\mathcal{T}$ and $\mathcal{G}_\mathcal{A}$ of the example.

We are now ready to formulate our characterization of the instance problem w.r.t. descriptive semantics (see [3] for the proof).

Theorem 2. *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. $\mathcal{A} \models_\mathcal{T} A(a)$.
2. *There is a simulation Z: $\mathcal{G}_\mathcal{T} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{A}$ such that*
   - $(A, a) \in Z$.
   - *Z is (B, u)-synchronized for all $(B, u) \in Z$.*

As an example, we consider the following TBox and ABox:

$$\mathcal{T} := \{A \equiv P \sqcap \exists r.A\} \quad \text{and} \quad \mathcal{A} := \{P(a), r(a,a), A(b), r(b,b)\}.$$

It is easy to see that there is no simulation satisfy the conditions of Theorem 2 for A and a. In contrast, the simulation $Z := \{(A, A), (A, b)\}$ satisfies these conditions for A and b (see also Figure 3).

Since the existence of a synchronized simulation relation satisfying the conditions stated in (2) of Theorem 2 can be decided in polynomial time (see [3]), the instance problem w.r.t. descriptive semantics is tractable.

Corollary 1. *The instance problem w.r.t. descriptive semantics in $\mathcal{EL}$ can be decided in polynomial time.*

5 The most specific concept

In this section, we will first show that the most specific concept w.r.t. descriptive semantics need not exist. Then, we will show that the most specific concept w.r.t. gfp-semantics (see [4]) coincides with the most specific concept w.r.t. descriptive semantics iff the ABox satisfies a certain acyclicity condition. This yields a sufficient condition for the existence of the msc, which is, however, not a necessary one. We will then characterize the cases in which the msc exists. Unfortunately, it is not yet clear how to turn this characterization into a decision procedure for the existence of the msc.

5.1 The msc need not exist

Theorem 3. *Let $\mathcal{T}_1 = \emptyset$ and $\mathcal{A} = \{r(b,b)\}$. Then b does not have an msc in $\mathcal{A}$ and $\mathcal{T}_1$.*

Proof. Assume to the contrary that $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$ such that the defined concept E in $\mathcal{T}_2$ is an msc of b. Let $\mathcal{G}_\mathcal{A}$ be the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}_2$, as introduced in Definition 8. Since b is an instance of E, there is a simulation $Z\colon \mathcal{G}_{\mathcal{T}_2} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{A}$ such that $(E, b) \in Z$ and Z is (B,u)-synchronized for all $(B,u) \in Z$.

Since $\mathcal{T}_1 = \emptyset$, there is no edge in $\mathcal{G}_\mathcal{A}$ from b to a defined concept in $\mathcal{T}_2$. Thus, the fact that Z is (E,b)-synchronized implies that there cannot be an infinite path in $G_{\mathcal{T}_2}$ (and thus $\mathcal{G}_\mathcal{A}$) starting with E. Consequently, there is an upper-bound n_0 on the length of the paths in $G_{\mathcal{T}_2}$ (and thus $\mathcal{G}_\mathcal{A}$) starting with E. Now, consider the TBox $\mathcal{T}_3 = \{F_n \equiv \exists r.F_{n-1}, \ldots, F_1 \equiv \exists r.F_0, F_0 \equiv \top\}$. It is easy to see that $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_2$ (where we assume without loss of generality that $F_0, \ldots, F_n$ are concept names not occurring in $\mathcal{T}_2$) and that $\mathcal{A} \models_{\mathcal{T}_3} F_n(b)$. Since E is an msc of b, this implies that $E \sqsubseteq_{\mathcal{T}_3} F_n$. Thus, there is an (F_n, E)-synchronized simulation $Y\colon \mathcal{G}_{\mathcal{T}_3} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{T}_3}$ such that $(F_n, E) \in Y$. However, for $n > n_0$, the path

$$F_n \xrightarrow{r} F_{n-1} \xrightarrow{r} \cdots \xrightarrow{r} F_0$$

cannot be simulated by a path starting from E. □

5.2 A sufficient condition for the existence of the 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 8. We can 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 $\mathcal{G}_\mathcal{A} = \mathcal{G}_{\mathcal{T}_2}$. It is easy to see that $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$. 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 $\mathcal{A}$. 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.

In [4] it is shown that, w.r.t. gfp-semantics, the defined concept C_a in $\mathcal{T}_2$ is the most specific concept of a in $\mathcal{A}$ and $\mathcal{T}_1$. W.r.t. descriptive semantics, this is only true if $\mathcal{A}$ does not contain a cycle that is reachable from a.

Definition 9. *The ABox $\mathcal{A}$ is called a-acyclic iff there are no $n \geq 1$ and individuals $a_0, a_1, \ldots, a_n$ and roles $r_1, \ldots, r_n$ such that*

- $a = a_0$,
- $r_i(a_{i-1}, a_i) \in \mathcal{A}$ *for* $1 \leq i \leq n$,
- *there is a* $j, 0 \leq j < n$ *such that* $a_j = a_n$.

Theorem 4. *Let $\mathcal{T}_1$, $\mathcal{A}$, a, and $\mathcal{T}_2$ be defined as above. Then the following are equivalent:*

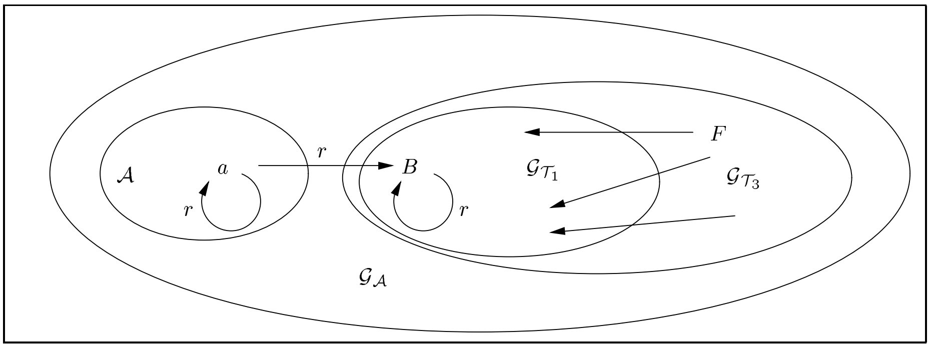

Fig. 4. The $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{A}$ in the proof of Proposition 1.

1. *The defined concept* C_a *in* $\mathcal{T}_2$ *is the msc of* a *in* $\mathcal{A}$ *and* $\mathcal{T}_1$.
2. $\mathcal{A} \models_{\mathcal{T}_2} C_a(a)$.
3. $\mathcal{A}$ *is* a*-acyclic.*

A proof of this theorem can be found in [3]. Given $\mathcal{T}$ and an a-acyclic ABox $\mathcal{A}$, the graph $\mathcal{G}_\mathcal{A}$ can obviously be computed in polynomial time, and thus the msc can in this case be computed in polynomial time.

Corollary 2. *Let* $\mathcal{T}_1$ *be an* $\mathcal{EL}$*-TBox and* $\mathcal{A}$ *an* $\mathcal{EL}$*-ABox containing the individual name* a *such that* $\mathcal{A}$ *is* a*-acyclic. Then the msc of* a *in* $\mathcal{T}_1$ *and* $\mathcal{A}$ *always exists, and it can be computed in polynomial time.*

The a-acyclicity of $\mathcal{A}$ is thus a sufficient condition for the existence of the msc. The following proposition states that this is not a necessary condition.

Proposition 1. *There exists an* $\mathcal{EL}$*-TBox* $\mathcal{T}_1$ *and an* $\mathcal{EL}$*-ABox* $\mathcal{A}$ *containing the individual name* a *such that the msc of* a *in* $\mathcal{T}_1$ *and* $\mathcal{A}$ *exists, even though* $\mathcal{A}$ *is not* a*-acyclic.*

Proof. Let $\mathcal{T}_1 = \{B \equiv \exists r.B\}$ and $\mathcal{A} = \{r(a,a),\, B(a)\}$. We show that B in $\mathcal{T}_1$ is the msc of a in $\mathcal{A}$ and $\mathcal{T}_1$. Since $\mathcal{A}$ is obviously not a-acyclic, this shows that a-acyclicity of $\mathcal{A}$ is not a necessary condition for the existence of the msc.

The instance relationship $\mathcal{A} \models_{\mathcal{T}_1} B(a)$ is trivially true since $B(a) \in \mathcal{A}$. Now, assume that $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_1$, and that the defined concept F in $\mathcal{T}_3$ satisfies $\mathcal{A} \models_{\mathcal{T}_3} F(a)$. Let $\mathcal{G}_\mathcal{A}$ be the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}_3$, as introduced in Definition 8 (see Figure 4). Since $\mathcal{A} \models_{\mathcal{T}_3} F(a)$, there is a simulation $Z\colon \mathcal{G}_{\mathcal{T}_3} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{A}$ such that $(F, a) \in Z$ and Z is (C, u)-synchronized for all $(C, u) \in Z$.

We must show that $B \sqsubseteq_{\mathcal{T}_3} F$, i.e., there is an (F, B)-synchronized simulation $Y\colon \mathcal{G}_{\mathcal{T}_3} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{T}_3}$ such that $(F, B) \in Y$. We define Y as follows:

$$\begin{aligned} Y := \{(u,v) \mid (u,v) \in Z \text{ and } v \text{ is a defined concept in } \mathcal{T}_3\} \;\cup \\ \{(u,B) \mid (u,a) \in Z\}. \end{aligned}$$

Since $(F, a) \in Z$ we have $(F, B) \in Y$. Next, we show that Y is a simulation.

(S1) is trivially satisfied since $\mathcal{T}_1$ (and thus also $\mathcal{T}_3$) does not contain primitive concepts. Consequently, all node labels are empty.

(S2) Let $(u, v) \in Y$ and (u, r, v) be an edge in $\mathcal{G}_{\mathcal{T}_3}$.[3]

First, assume that v is a defined concept in $\mathcal{T}_3$ and $(u, v) \in Z$. Since Z is a simulation, there exists a node v' in $\mathcal{G}_\mathcal{A}$ such that (v, r, v') is an edge in $\mathcal{G}_\mathcal{A}$ and $(u', v') \in Z$. By the definition of $\mathcal{G}_\mathcal{A}$, this implies that also v' is a defined concept in $\mathcal{T}_3$, and thus (v, r, v') is an edge in $\mathcal{G}_{\mathcal{T}_3}$ and $(u', v') \in Y$.

Second, assume that $v = B$ and $(u, a) \in Z$. Since Z is a simulation, there exists a node v' in $\mathcal{G}_\mathcal{A}$ such that (a, r, v') is an edge in $\mathcal{G}_\mathcal{A}$ and $(u', v') \in Z$. Since there are only two edges with source a in $\mathcal{G}_\mathcal{A}$, we know that $v' = a$ or $v' = B$. If $v' = B$, then v' is a defined concept in $\mathcal{T}_3$, and thus (v, r, v') is an edge in $\mathcal{G}_{\mathcal{T}_3}$ and $(u', v') \in Y$. If $v' = a$, then (B, r, B) is an edge in $\mathcal{G}_{\mathcal{T}_3}$ and $(u', a) \in Z$ yields $(u', B) \in Y$.

Thus, we have shown that Y is indeed a simulation from $\mathcal{G}_{\mathcal{T}_3}$ to $\mathcal{G}_{\mathcal{T}_3}$. It remains to be shown that it is (F, B)-synchronized. Since (B, r, B) is the only edge in $\mathcal{G}_{\mathcal{T}_3}$ with source B, the selection function always chooses B. Thus, it is enough to show that any infinite path starting with F in $\mathcal{G}_{\mathcal{T}_3}$ eventually leads to B. This is an easy consequence of the fact that Z is (F, a)-synchronized and that the only node in $\mathcal{G}_{\mathcal{T}_3}$ reachable in $\mathcal{G}_\mathcal{A}$ from a is B. □

5.3 Characterizing when the msc exists

The example that demonstrates the non-existence of the msc given above (see Theorem 3) shows that cycles in the ABox are problematic. However, Proposition 1 shows that not all cycles cause problems. Intuitively, the reason for some cycles being harmless is that they can be simulated by cycles in the TBox. For this reason, it is not really necessary to have them in $\mathcal{G}_\mathcal{A}$. In order to make this more precise, we will introduce acyclic versions $\mathcal{G}_\mathcal{A}^{(k)}$ of $\mathcal{G}_\mathcal{A}$, where cycles are unraveled into paths up to depth k starting with a (see Definition 10 below). When viewed as the $\mathcal{EL}$-description graph of an $\mathcal{EL}$-TBox, this graph contains a defined concept that corresponds to the individual a. Let us call this concept P_k. We will see below that the msc of a exists iff there is a k such that P_k is the msc.[4] Unfortunately, it is not clear how this condition can be decided in an effective way.

Definition 10. *Let $\mathcal{T}_1$ be a fixed $\mathcal{EL}$-TBox with associated $\mathcal{EL}$-description graph $\mathcal{G}_{\mathcal{T}_1} = (V_{\mathcal{T}_1}, E_{\mathcal{T}_1}, L_{\mathcal{T}_1})$, $\mathcal{A}$ an $\mathcal{EL}$-ABox, a a fixed individual in $\mathcal{A}$, and $k \geq 0$. Then the graph $\mathcal{G}_\mathcal{A}^{(k)} := (V_k, E_k, L_k)$ is defined as follows:*

$$V_k := V_{\mathcal{T}_1} \cup \{a^0\} \cup \{b^n \mid b \text{ is an individual in } \mathcal{A} \text{ and } 1 \leq n \leq k\},$$

[3] Since r is the only role occurring in $\mathcal{T}_1$, it is also the only role occurring in the conservative extension $\mathcal{T}_3$ of $\mathcal{T}_1$.

[4] This result is similar to the characterization of the existence of the lcs w.r.t. descriptive semantics given in [2].

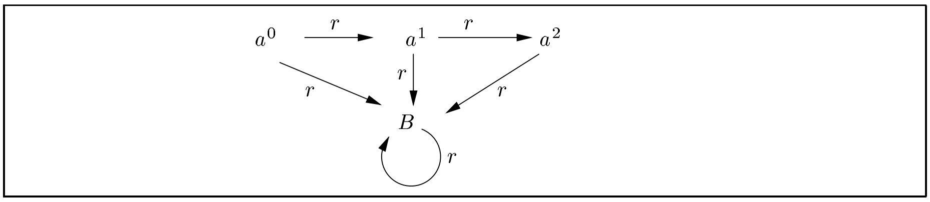

Fig. 5. The $\mathcal{EL}$-description graph $\mathcal{G}_{\mathcal{A}}^{(2)}$ of the example in the proof of Proposition 1.

where a^0 and b^n are new individual names;

$$\begin{aligned} E_k := E_{\mathcal{T}_1} \;\cup \\ &\{(b^i, r, c^{i+1}) \mid r(b,c) \in \mathcal{A}, b^i, c^{i+1} \in V_k \setminus V_{\mathcal{T}_1}\} \;\cup \\ &\{(b^i, r, B) \mid A(b) \in \mathcal{A}, b^i \in V_k \setminus V_{\mathcal{T}_1}, (A, r, B) \in E_{\mathcal{T}_1}\}; \end{aligned}$$

If u is a node in $V_{\mathcal{T}_1}$, then

$$L_k(u) := L_{\mathcal{T}_1}(u);$$

and if $u = b^i \in V_k \setminus V_{\mathcal{T}_1}$, then

$$L_k(u) := \{P \mid P(b) \in \mathcal{A}\} \cup \bigcup_{A(b) \in \mathcal{A}} L_{\mathcal{T}_1}(A),$$

where P denotes primitive and A denotes defined concepts.

As an example, consider the TBox $\mathcal{T}_1$ and the ABox $\mathcal{A}$ introduced in the proof of Proposition 1. The corresponding graph $\mathcal{G}_{\mathcal{A}}^{(2)}$ is depicted in Figure 5 (where the empty node labels are omitted).

Let $\mathcal{T}_2^{(k)}$ be the $\mathcal{EL}$-TBox corresponding to $\mathcal{G}_{\mathcal{A}}^{(k)}$. In this TBox, a^0 is a defined concept, which we denote by P_k. For example, the TBox corresponding to the graph $\mathcal{G}_{\mathcal{A}}^{(2)}$ depicted in Figure 5 consists of the following definitions (where nodes corresponding to individuals have been renamed[5]):

$$P_2 \equiv \exists r.A_1 \sqcap \exists r.B, \quad A_1 \equiv \exists r.A_2 \sqcap \exists r.B, \quad A_2 \equiv \exists r.B, \quad B \equiv \exists r.B.$$

Any msc of a must be equivalent to one of the concepts P_k:

Theorem 5. *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox, $\mathcal{A}$ an $\mathcal{EL}$-ABox, and a an individual in $\mathcal{A}$. Then there exists an msc of a in $\mathcal{A}$ and $\mathcal{T}_1$ iff there is a $k \geq 0$ such that P_k in $\mathcal{T}_2^{(k)}$ is the msc of a in $\mathcal{A}$ and $\mathcal{T}_1$.*

This theorem, whose proof can be found in [3], is an easy consequence of the following two lemmas. The first lemma states that a is an instance of the concepts P_k.

[5] This renaming is admissible since these nodes cannot occur on cycles

Lemma 1. *$\mathcal{A} \models_{\mathcal{T}_2^{(k)}} P_k(a)$ for all $k \geq 0$.*

The second lemma says that every concept that has a as an instance also subsumes P_k for an appropriate k. To make this more precise, assume that $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$, and that F is a defined concept in $\mathcal{T}_2$ such that $\mathcal{A} \models_{\mathcal{T}_2} F(a)$. Let $k := n \cdot (n+m)$ where n is the number of defined concepts in $\mathcal{T}_2$ and m is the number of individuals in $\mathcal{A}$. In order to have a subsumption relationship between P_k and F, both must "live" in the same TBox. For this, we simply take the union $\mathcal{T}_3$ of $\mathcal{T}_2^{(k)}$ and $\mathcal{T}_2$. Note that we may assume without loss of generality that the only defined concepts that $\mathcal{T}_2^{(k)}$ and $\mathcal{T}_2$ have in common are the ones from $\mathcal{T}_1$. In fact, none of the new defined concepts in $\mathcal{T}_2^{(k)}$ (i.e., the elements of $V_k \setminus V_{\mathcal{T}_1}$) lies on a cycle, and thus we can rename them without changing the meaning of these concepts. (Note that the characterization of subsumption given in Theorem 1 implies that only for defined concepts occurring on cycles their actual names are relevant.) Thus, $\mathcal{T}_3$ is a conservative extension of both $\mathcal{T}_2^{(k)}$ and $\mathcal{T}_2$.

Lemma 2. *If $k := n \cdot (n+m)$ where n is the number of defined concepts in $\mathcal{T}_2$ and m is the number of individuals in $\mathcal{A}$, then $P_k \sqsubseteq_{\mathcal{T}_3} F$.*

In the following, we assume without loss of generality that the TBoxes $\mathcal{T}_2^{(k)}$ ($k \geq 0$) are renamed such that they share only the defined concepts of $\mathcal{T}_1$.

Lemma 3. *Let $\mathcal{T} := \mathcal{T}_2^{(k)} \cup \mathcal{T}_2^{(k+1)}$. Then $P_{k+1} \sqsubseteq_{\mathcal{T}} P_k$.*

Thus, the concepts P_k form a decreasing chain w.r.t. subsumption. The individual a has an msc iff this chain becomes stable.

Corollary 3. *P_k is the msc of a iff it is equivalent to P_{k+i} for all $i \geq 1$.*

As an example, consider the TBox $\mathcal{T}_1$ and the ABox $\mathcal{A}$ introduced in the proof of Proposition 1 (see also Figure 5). It is easy to see that in this case P_0 is equivalent to P_k for all $k \geq 1$, and thus P_0 is the msc of a in $\mathcal{T}_1$ and $\mathcal{A}$.

6 Conclusion

The impact of cyclic definitions in $\mathcal{EL}$ on both standard and non-standard inferences in now well-investigated. The only two questions left open are how to give a decidable characterization of the cases in which the lcs/msc exists w.r.t. descriptive semantics, and to determine whether it can then be computed in polynomial time.

Though the characterizations of the existence of the lcs/msc given in [2] and in this paper do not provide us with such a decision procedure, they can be seen as a first step in this direction. In addition, these characterizations can be used to compute approximations of the lcs/msc.

References

1. Franz Baader. Using automata theory for characterizing the semantics of terminological cycles. *Ann. of Mathematics and Artificial Intelligence*, 18:175–219, 1996.
2. Franz Baader. Computing the least common subsumer in the description logic $\mathcal{EL}$ w.r.t. terminological cycles with descriptive semantics. In *Proceedings of the 11th International Conference on Conceptual Structures*, Lecture Notes in Artificial Intelligence. Springer-Verlag, 2003. To appear.
3. Franz Baader. The instance problem and the most specific concept in the description logic $\mathcal{EL}$ w.r.t. terminological cycles with descriptive semantics. LTCS-Report LTCS-03-01, Chair for Automata Theory, Institute for Theoretical Computer Science, Dresden University of Technology, Germany, 2003. See http://lat.inf.tu-dresden.de/research/reports.html.
4. Franz Baader. Least common subsumers and most specific concepts in a description logic with existential restrictions and terminological cycles. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 2003. To appear.
5. Franz Baader. Terminological cycles in a description logic with existential restrictions. In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 2003. To appear.
6. Franz Baader and Ralf Küsters. Computing the least common subsumer and the most specific concept in the presence of cyclic $\mathcal{ALN}$-concept descriptions. In *Proc. of the 22nd German Annual Conf. on Artificial Intelligence (KI'98)*, volume 1504 of *Lecture Notes in Computer Science*, pages 129–140. Springer-Verlag, 1998.
7. Franz Baader, Ralf Küsters, and Ralf Molitor. Computing least common subsumers in description logics with existential restrictions. In *Proc. of the 16th Int. Joint Conf. on Artificial Intelligence (IJCAI'99)*, pages 96–101, 1999.
8. Volker Haarslev and Ralf Möller. High performance reasoning with very large knowledge bases: A practical case study. In *Proc. of the 17th Int. Joint Conf. on Artificial Intelligence (IJCAI 2001)*, 2001.
9. Monika R. Henzinger, Thomas A. Henzinger, and Peter W. Kopke. Computing simulations on finite and infinite graphs. In *36th Annual Symposium on Foundations of Computer Science*, pages 453–462, Milwaukee, Wisconsin, 1995. IEEE Computer Society Press.
10. Ian Horrocks. Using an expressive description logic: FaCT or fiction? In *Proc. of the 6th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'98)*, pages 636–647, 1998.
11. Ralf Küsters. *Non-standard Inferences in Description Logics*, volume 2100 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 2001.
12. Bernhard Nebel. *Reasoning and Revision in Hybrid Representation Systems*, volume 422 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 1990.
13. Bernhard Nebel. Terminological cycles: Semantics and computational properties. In John F. Sowa, editor, *Principles of Semantic Networks*, pages 331–361. Morgan Kaufmann, Los Altos, 1991.
14. Manfred Schmidt-Schauß and Gert Smolka. Attributive concept descriptions with complements. *Artificial Intelligence*, 48(1):1–26, 1991.
15. K.A. Spackman. Normal forms for description logic expressions of clinical concepts in SNOMED RT. *J. of the American Medical Informatics Association*, pages 627–631, 2001. Symposium Supplement.
16. K.A. Spackman, K.E. Campbell, and R.A. Cote. SNOMED RT: A reference terminology for health care. *J. of the American Medical Informatics Association*, pages 640–644, 1997. Fall Symposium Supplement.