TECHNISCHE
UNIVERSITÄT
DRESDEN

Dresden University of Technology
Institute for Theoretical Computer Science
Chair for Automata Theory

LTCS–Report

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

Franz Baader

LTCS-Report 03-01

Lehrstuhl für Automatentheorie
Institut für Theoretische Informatik
TU Dresden
http://lat.inf.tu-dresden.de

Hans-Grundig-Str. 25
01062 Dresden
Germany

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
Dresden University of Technology
D-01062 Dresden, Germany
e-mail: baader@inf.tu-dresden.de

Abstract

In two previous reports 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. Regarding standard inference problems, it was shown there that the subsumption problem remains polynomial for all three types of semantics usually considered for cyclic definitions in description logics, and that the instance problem remains polynomial for greatest fixpoint semantics. Regarding non-standard inference problems, it was shown that, w.r.t. greatest fixpoint semantics, the least common subsumer and the most specific concept always exist and can be computed in polynomial time, and that, w.r.t. descriptive semantics, the least common subsumer need not exist.

The present report 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 the instance problem is polynomial also 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.

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

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 [7, 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 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 in this context.

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 (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 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 [5] 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 report [1] 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 stark 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 [2, 3]. 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 [1] 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 [6].) For cyclic terminologies in $\mathcal{EL}$ with descriptive semantics, the lcs need not exist. In [4], 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 [1] and [4]. In the present report 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.

In the next section, we introduce $\mathcal{EL}$ with cyclic terminologies as well the msc. Then we recall the important definitions and results from [1] and [4]. Section 4 formulates and proves the new results for the instance problem, and Section 5 does the same for the msc.

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 report, 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 $A^\mathcal{I}$ 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

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	$A(a)$	$a^{\mathcal{I}} \in \mathcal{A}^{\mathcal{I}}$
role assertion	$r(a, b)$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$

Table 1: Syntax and semantics of $\mathcal{EL}$-concept descriptions, TBox definitions, and ABox assertions.

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 report we restrict our attention to descriptive semantics (see [1, 4] for definitions and results concerning cyclic terminologies in $\mathcal{EL}$ with fixpoint semantics).

We are now ready to define the subsumption and the instance problem w.r.t. descriptive 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 what kind of defined concepts 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 the proof).

Lemma 2 *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 $A \sqsubseteq_{\mathcal{T}_1} B$ iff $A \sqsubseteq_{\mathcal{T}_2} B$.*

Definition 3 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).

[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.

Proposition 4 *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox and $\mathcal{A}$ an $\mathcal{EL}$-ABox containing the individual name a. 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 an msc of a in $\mathcal{A}$ and $\mathcal{T}_1$;*
- *the defined concept E' in $\mathcal{T}_2'$ is an msc of a in $\mathcal{A}$ and $\mathcal{T}_1$;*
- *the sets of newly defined concepts in respectively $\mathcal{T}_2$ and $\mathcal{T}_2'$ are disjoint.*

Where $\mathcal{T}_3 := \mathcal{T}_2 \cup \mathcal{T}_2'$, we have $E \equiv_{\mathcal{T}_3} E'$.

3 Characterizing subsumption in $\mathcal{EL}$ with cyclic definitions

In this section, we recall the characterizations of subsumption w.r.t. descriptive semantics developed in [1]. To this purpose, we must represent TBoxes by description graphs, and introduce the notion of a simulation on description graphs.

3.1 Description graphs and simulations

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 [1], 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 5 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 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 6 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 $Z\colon \mathcal{G}_1 \rightrightarrows\!\!\!\!\sim \mathcal{G}_2$ 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 $\mathcal{G}_1$ to $\mathcal{G}_2$ is closed under arbitrary unions. Consequently, there always exists a greatest simulation from $\mathcal{G}_1$ to $\mathcal{G}_2$. If $\mathcal{G}_1, \mathcal{G}_2$ are finite, then this greatest simulation can be computed in polynomial time [9]. As an easy consequence of this fact, the following proposition is proved in [1].

Proposition 7 *Let $\mathcal{G}_1, \mathcal{G}_2$ be two finite $\mathcal{EL}$-description graphs, v_1 a node of $\mathcal{G}_1$ and v_2 a node of $\mathcal{G}_2$. Then we can decide in polynomial time whether there is a simulation $Z\colon \mathcal{G}_1 \rightrightarrows\!\!\!\!\sim \mathcal{G}_2$ such that $(v_1, v_2) \in Z$.*

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

Figure 1: A (B, A)-simulation chain.

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

Figure 2: A partial (B, A)-simulation chain.

3.2 Subsumption w.r.t. descriptive semantics

W.r.t. gfp-semantics, A is subsumed by B iff there is a simulation Z: $\mathcal{G}_\mathcal{T} \stackrel{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{T}$ such that $(B, A) \in Z$ (see [1]). 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.

Definition 8 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 6 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 9 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 10 Let A, B be defined concepts in $\mathcal{T}$, and $Z\colon \mathcal{G}_\mathcal{T} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{T}$ 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 proved in [1].

Theorem 11 *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 $Z\colon \mathcal{G}_\mathcal{T} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{T}$ such that $(B, A) \in Z$.*

In [1] it is also proved 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 $Z\colon \mathcal{G}_\mathcal{T} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{T}$ 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.

The proof of Theorem 11 in [1] depends on an appropriate characterization of when an individual in a model belongs to a defined concept in this model. This characterization will also be useful when proving our results for the instance problem in Section 4. Before we can formulate this characterization, we need to introduce some notation.

3.3 Primitive interpretations and $\mathcal{I}_0$-gfp-models

Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox containing the roles N_{role}, the primitive concepts N_{prim}, and the defined concepts $N_{def} := \{A_1, \ldots, A_k\}$, and let $\mathcal{A}$ be an $\mathcal{EL}$-ABox containing the individual names N_{ind}. A *primitive interpretations* $\mathcal{J}$ for $\mathcal{T}$ and $\mathcal{A}$ is given by a domain $\Delta^\mathcal{J}$, an interpretation of the roles $r \in N_{role}$ by binary relations $r^\mathcal{J}$ on $\Delta^\mathcal{J}$, an interpretation of the primitive concepts $P \in N_{prim}$ by subsets $P^\mathcal{J}$ of $\Delta^\mathcal{J}$, and interpretation of the individual. Obviously, a primitive interpretation differs from an interpretation in that it does not interpret the defined concepts in N_{def}. Any primitive interpretation can be translated into an $\mathcal{EL}$-description graph (where we do not represent the interpretation of individual names):

Definition 12 The primitive interpretation $\mathcal{J} = (\Delta^\mathcal{J}, \cdot^\mathcal{J})$ is translated into the $\mathcal{EL}$-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 $x \in \Delta^\mathcal{J}$.

We say that the interpretation $\mathcal{I}$ is *based on* the primitive interpretation $\mathcal{J}$ iff it has the same domain as $\mathcal{J}$ and coincides with $\mathcal{J}$ on N_{role}, N_{prim}, and N_{ind}. For a fixed primitive interpretation $\mathcal{J}$, the interpretations $\mathcal{I}$ 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 $\mathcal{J}$ can be compared by the following ordering, which realizes a pairwise inclusion test between the respective interpretations of the defined concepts: if $\mathcal{I}_1, \mathcal{I}_2 \in Int(\mathcal{J})$, then

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

It is easy to see that $\preceq_\mathcal{J}$ induces a *complete lattice* on $Int(\mathcal{J})$, i.e., every subset of $Int(\mathcal{J})$ has a least upper bound (lub) and a greatest lower bound (glb). Thus, *Tarski's fixpoint theorem* [17] applies to all monotonic functions from $Int(\mathcal{J})$ to $Int(\mathcal{J})$.

Definition 13 The $\mathcal{EL}$-TBox $\mathcal{T} := \{A_1 \equiv D_1, \ldots, A_k \equiv D_k\}$ induces the following function $O_{\mathcal{T},\mathcal{J}}$ on $Int(\mathcal{J})$: $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_1) = \mathcal{I}_2$ iff $A_i^{\mathcal{I}_2} = D_i^{\mathcal{I}_1}$ holds for all $i, 1 \le i \le k$.

Monotonicity of this function is an easy consequence of the fact that the concept constructors of $\mathcal{EL}$ are all monotonic (see [1] for details). It is also an immediate consequence of the definition of $O_{\mathcal{T},\mathcal{J}}$ that any interpretation $\mathcal{I}$ based on $\mathcal{J}$ is a fixpoint of $O_{\mathcal{T},\mathcal{J}}$ iff $\mathcal{I}$ is a model of $\mathcal{T}$. Greatest (least) fixpoint semantics restricts the attention to models that are greatest (least) fixpoints of $O_{\mathcal{T},\mathcal{J}}$ (which exist by Tarski's fixpoint theorem).

In the context of descriptive semantics, it is sometimes convenient to consider models of $\mathcal{T}$ that are the greatest models below a given interpretation $\mathcal{I}_0$.

Definition 14 Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{J}$ a primitive interpretation, and $\mathcal{I}_0$ an interpretation based on $\mathcal{J}$. The model $\mathcal{I}$ of $\mathcal{T}$ is called $\mathcal{I}_0$-*model of* $\mathcal{T}$ iff it is based on $\mathcal{J}$ and satisfies $\mathcal{I} \preceq_\mathcal{J} \mathcal{I}_0$. The greatest $\mathcal{I}_0$-model of $\mathcal{T}$ (if it exists) is called $\mathcal{I}_0$-*gfp-model of* $\mathcal{T}$.

If $\mathcal{I}_0$ is itself a model of $\mathcal{T}$, then it is also the $\mathcal{I}_0$-gfp-model of $\mathcal{T}$. The following proposition (whose proof can be found in [1]) describes a more general sufficient condition for the greatest $\mathcal{I}_0$-model of $\mathcal{T}$ to exist.

Proposition 15 *If $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_{\mathcal{J}} \mathcal{I}_0$, then $\mathcal{T}$ has an $\mathcal{I}_0$-gfp-model based on $\mathcal{J}$.*

We are now ready to give the announced characterization of when an individual of a model belongs to a defined concept in this model (see [1] for the proof). Since any model $\mathcal{I}$ of $\mathcal{T}$ is itself an $\mathcal{I}$-gfp-model of $\mathcal{T}$, it is sufficient to formulate the condition for $\mathcal{I}$-gfp-models of $\mathcal{T}$.

Proposition 16 *Let $\mathcal{J}$ be a primitive interpretation, $\mathcal{I}_0$ an interpretation based on $\mathcal{J}$ such that $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_{\mathcal{J}} \mathcal{I}_0$, and $\mathcal{I}$ the $\mathcal{I}_0$-gfp-model of $\mathcal{T}$. 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: $\mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}}$ such that*
   (a) $(A, x) \in Z$; and
   (b) if $(B, y) \in Z$ then $y \in B^{\mathcal{I}_0}$.

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 11: the individual a is an instance of A in $\mathcal{A}$ and $\mathcal{T}$ iff there is an (A, a)-synchronized simulation Z: $\mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{A}}$ 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 17 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:

[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.

- 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:

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

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

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

  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.

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

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 model of $\mathcal{T}$ is a model of $\mathcal{A}$. Before we can formulate this characterization we must introduce one more notation.

Definition 18 Let $\mathcal{J}$ be a primitive interpretation and $\mathcal{G}_\mathcal{J}$ the $\mathcal{EL}$-description graph associated with $\mathcal{J}$. We say that the simulation $Z\colon \mathcal{G}_\mathcal{A} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{J}$ *respects ABox individuals* iff

$$\{x \mid (a, x) \in Z\} = \{a^\mathcal{J}\}$$

for all individual names a occurring in $\mathcal{A}$.

Since the primitive interpretation $\mathcal{J}$ interprets the primitive concepts and roles as well as the individual names, the question whether an interpretation $\mathcal{I}$ based on $\mathcal{J}$ satisfies a role assertion $r(a, b)$ (a concept assertion $P(a)$ for a primitive concept P) or not depends only on $\mathcal{J}$. Thus, it makes sense to say that a primitive interpretation satisfies a role assertion or a concept assertion for a primitive concept.

Lemma 19 *If there exists a simulation $Z\colon \mathcal{G}_\mathcal{A} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{J}$ that respects ABox individuals, then $\mathcal{J}$ satisfies all role assertions $r(a, b) \in \mathcal{A}$ and all concept assertions $P(a) \in \mathcal{A}$ where P is a primitive concept.*

A proof of this lemma is included in the proof of ($2 \Rightarrow 1$) of Proposition 45 in [4].

Proposition 20 *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{A}$ an ABox, $\mathcal{J}$ a primitive interpretation, $\mathcal{I}_0$ an interpretation based on $\mathcal{J}$ such that*

- $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_{\mathcal{J}} \mathcal{I}_0$ *and*
- $B(b) \in \mathcal{A}$ *implies* $b^{\mathcal{J}} \in B^{\mathcal{I}_0}$,

and $\mathcal{I}$ the $\mathcal{I}_0$-gfp-model of $\mathcal{T}$. Then the following are equivalent:

1. *$\mathcal{I}$ is a model of $\mathcal{A}$.*
2. *There is a simulation $Z\colon \mathcal{G}_{\mathcal{A}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}}$ that respects ABox individuals and satisfies*
$$(B, y) \in Z \Rightarrow y \in B^{\mathcal{I}_0}$$
for all defined concepts B in $\mathcal{T}$ and all elements $y \in \Delta^{\mathcal{J}}$.

Proof. ($2 \Rightarrow 1$) Assume that a simulation $Z\colon \mathcal{G}_{\mathcal{A}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}}$ satisfying the properties stated in (2) of the proposition is given. We must show that $\mathcal{I}$ satisfies all the assertions in $\mathcal{A}$.

For role assertions and concept assertions involving primitive concepts, this is the case by Lemma 19.

Thus, assume that $A(a)$ is a concept assertion in $\mathcal{A}$ where A is a defined concept. We use Proposition 16 to show that $a^{\mathcal{J}} = a^{\mathcal{I}} \in A^{\mathcal{I}}$. Thus, we need to find a simulation $Y\colon \mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}}$ such that $(A, a^{\mathcal{J}}) \in Y$ and $(B, y) \in Y$ implies $y \in B^{\mathcal{I}_0}$ for all defined concepts B in $\mathcal{T}$. We define the relation Y as follows:

$$Y := \{(A, a^{\mathcal{J}})\} \cup \{(B, x) \mid (B, x) \in Z \text{ where } B \text{ is a defined concept in } \mathcal{T}\}.$$

Thus, Y is the restriction of Z to the nodes of $\mathcal{G}_{\mathcal{T}}$, extended by the tuple $(A, a^{\mathcal{J}})$. The fact that Y is a simulation relation (i.e., satisfies (S1) and (S2) of Definition 6) can be shown as in the proof of ($2 \Rightarrow 1$) of Proposition 45 in [4]. In addition, we have $(A, a^{\mathcal{J}}) \in Y$ by definition of Y. Finally, assume that $(B, y) \in Y$. If $(B, y) \in Z$, then $y \in B^{\mathcal{I}_0}$ follows from our assumption on Z. It remains to be shown that $(A, a^{\mathcal{J}}) \in Y$ implies $a^{\mathcal{J}} \in A^{\mathcal{I}_0}$. Since $A(a) \in \mathcal{A}$, this is the case by our assumption on $\mathcal{I}_0$.

($1 \Rightarrow 2$) Assume that $\mathcal{I}$ is a model of $\mathcal{A}$. In particular, this implies that $a^{\mathcal{J}} = a^{\mathcal{I}} \in A^{\mathcal{I}}$ holds for all concept assertions $A(a) \in \mathcal{A}$. If A is a defined concept, then Proposition 16 implies the existence of simulation relations $Z_{A(a)}\colon \mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}}$ such that

- $(A, a^{\mathcal{J}}) \in Z_{A(a)}$ and

- $(B, y) \in Z_{A(a)} \Rightarrow y \in B^{\mathcal{I}_0}$.

Let Y be the union of these simulations, i.e.,

$$Y := \bigcup_{A(a) \in \mathcal{A}} Z_{A(a)}.$$

Then Y is a simulation relation that satisfies $(A, a^{\mathcal{J}}) \in Y$ for all concept assertions $A(a) \in \mathcal{A}$ where A is a defined concept. In addition, if B is a defined concept in $\mathcal{T}$ and $y \in \Delta^{\mathcal{J}}$, then $(B, y) \in Y$ implies that $(B, y) \in Z_{A(a)}$ for some concept assertion $A(a) \in \mathcal{A}$, and thus $y \in B^{\mathcal{I}_0}$.

We define the relation Z as follows:

$$Z := Y \cup \{(a, a^{\mathcal{J}}) \mid a \text{ is an individual name occurring in } \mathcal{A}\}.$$

By definition, Z respects ABox individuals, and it satisfies $(B, y) \in Z \Rightarrow y \in B^{\mathcal{I}_0}$ since Y satisfies this property. Thus, it remains to be shown that Z is a simulation from $\mathcal{G}_{\mathcal{A}}$ to $\mathcal{G}_{\mathcal{J}}$. This can be shown as in the proof of $(1 \Rightarrow 2)$ of Proposition 45 in [4]. □

In the remainder of this section, we will us this proposition to prove correctness of the following characterization of the instance problem.

Theorem 21 *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 $Z\colon \mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{A}}$ such that*
   - *$(A, a) \in Z$.*
   - *Z is (B, u)-synchronized for all $(B, u) \in Z$.*

Proof of $(2 \Rightarrow 1)$ of Theorem 21

Assume that there is a simulation $Z\colon \mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{A}}$ satisfying the two properties stated in (2) of the theorem. We must show $\mathcal{A} \models_{\mathcal{T}} A(a)$, i.e., if $\mathcal{I}$ is a model of $\mathcal{T}$ and $\mathcal{A}$, then $a^{\mathcal{I}} \in A^{\mathcal{I}}$.

Thus, let $\mathcal{I}$ be a model of $\mathcal{T}$ and $\mathcal{A}$, and let $\mathcal{J}$ be the primitive interpretation on which $\mathcal{I}$ is based. If we define $\mathcal{I}_0 := \mathcal{I}$, then

- $\mathcal{I}_0$ is a model of $\mathcal{A}$, and thus $B(b) \in \mathcal{A}$ implies $b^{\mathcal{J}} \in B^{\mathcal{I}_0}$;
- $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_{\mathcal{J}} \mathcal{I}_0$ (in fact, $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) = \mathcal{I}_0$ since $\mathcal{I}_0$ is a model of $\mathcal{T}$);

- $\mathcal{I}$ is the $\mathcal{I}_0$-gfp-model of $\mathcal{T}$.

Thus, the prerequisites for Proposition 20 are satisfied, and the fact that $\mathcal{I}$ is a model of $\mathcal{A}$ yields a simulation $Y\colon \mathcal{G}_\mathcal{A} \rightleftharpoons \mathcal{G}_\mathcal{J}$ that respects ABox individuals and satisfies $(B, y) \in Y \Rightarrow y \in B^{\mathcal{I}_0}$ for all defined concepts B and elements y of $\Delta^\mathcal{J}$. The composition $X := Z \circ Y$ is a simulation from $\mathcal{G}_\mathcal{T}$ to $\mathcal{G}_\mathcal{J}$ (see Lemma 17 in [1]). By Proposition 16, to show $a^\mathcal{I} \in A^\mathcal{I}$, it is sufficient to show that

(a) $(A, a^\mathcal{I}) \in X$; and

(b) $(B, y) \in X$ implies $y \in B^{\mathcal{I}_0} = B^\mathcal{I}$.

Property (a) holds since we know that $(A, a) \in Z$ and the fact that Y respects ABox individuals implies that $(a, a^\mathcal{I}) \in Y$.

The proof of property (b) is similar to the proof of $(2 \Rightarrow 1)$ of Theorem 29 in [1]. Since there are, however, subtle differences, we include a detailed proof of property (b) for the sake of completeness. Thus, assume that $(B, y) \in X$, i.e., there is a node u in $\mathcal{G}_\mathcal{A}$ such that $(B, u) \in Z$ and $(u, y) \in Y$. We must show that this implies $y \in B^\mathcal{I}$.

Assume to the contrary that $y \notin B^\mathcal{I}$. Where

$$B \equiv P_1 \sqcap \ldots \sqcap P_m \sqcap \exists s_1.C_1 \sqcap \ldots \sqcap \exists s_\ell.C_\ell$$

is the definition of B in $\mathcal{T}$, this implies that there is an index $i, 1 \leq i \leq m$, such that $y \notin P_i^\mathcal{I} = P_i^\mathcal{J}$ or an index $j, 1 \leq j \leq \ell$ such that $y \notin (\exists s_j.C_j)^\mathcal{I}$.

The facts that $(B, u) \in Z$ and $(u, y) \in Y$ obviously imply that $P_j \in L_\mathcal{T}(B) \subseteq L_\mathcal{A}(u) \subseteq L_\mathcal{J}(y)$, and thus the first alternative cannot occur. Consequently, there is an index $j, 1 \leq j \leq \ell$ such that $y \notin (\exists s_j.C_j)^\mathcal{I}$.

Since $(B, u) \in Z$, we know that Z is (B, u)-synchronized. Let S be the corresponding selection function. Consider the partial (B, u)-simulation chain

$$\begin{array}{ccccc} B & = & B_0 & \xrightarrow{r_1} & B_1 \\ & & Z\downarrow & & \\ u & = & u_0 & & \end{array}$$

where $B_1 := C_j$ and $r_1 := s_j$. Then S yields a node u_1 in $\mathcal{G}_\mathcal{A}$ such that $(B_1, u_1) \in Z$ and (u_0, r_1, u_1) is an edge in $\mathcal{G}_\mathcal{A}$. Since Y is a simulation with $(u_0, y) \in Y$, this implies the existence of an individual $y_1 \in \Delta^\mathcal{J}$ such that (y, r_1, y_1) is an edge in $\mathcal{G}_\mathcal{J}$ (i.e., $(y, y_1) \in r_1^\mathcal{J}$) and $(u_1, y_1) \in Y$. Thus, we have the following situation:

$$\begin{array}{ccccc} B & = & B_0 & \xrightarrow{r_1} & B_1 \\ & & Z\downarrow & & Z\downarrow \\ u & = & u_0 & \xrightarrow{r_1} & u_1 \\ & & Y\downarrow & & Y\downarrow \\ y & = & y_0 & \xrightarrow{r_1} & y_1 \end{array}$$

where $y_0 := y$. Thus, $y = y_0 \notin (\exists s_j.C_j)^{\mathcal{I}} = (\exists r_1.B_1)^{\mathcal{I}}$ and $(y, y_1) \in r_1^{\mathcal{J}} = r_1^{\mathcal{I}}$ imply that $y_1 \notin {B_1}^{\mathcal{I}}$.

This shows that we can now continue with B_1, u_1, y_1 in place of B_0, u_0, y_0, etc. This yields the following pair of infinite simulation chains:

$$\begin{array}{ccccccccccc} B & = & B_0 & \stackrel{r_1}{\to} & B_1 & \stackrel{r_2}{\to} & B_2 & \stackrel{r_3}{\to} & B_3 & \stackrel{r_4}{\to} & \cdots \\ & & Z\!\downarrow & & Z\!\downarrow & & Z\!\downarrow & & Z\!\downarrow & & \\ u & = & u_0 & \stackrel{r_1}{\to} & u_1 & \stackrel{r_2}{\to} & u_2 & \stackrel{r_3}{\to} & u_3 & \stackrel{r_4}{\to} & \cdots \\ & & Y\!\downarrow & & Y\!\downarrow & & Y\!\downarrow & & Y\!\downarrow & & \\ & & y_0 & \stackrel{r_1}{\to} & y_1 & \stackrel{r_2}{\to} & y_2 & \stackrel{r_3}{\to} & y_3 & \stackrel{r_4}{\to} & \cdots \end{array}$$

where $y_n \notin B_n^{\mathcal{I}}$ for all $n \geq 0$. However, the upper chain was constructed using the selection function S (i.e., it is S-selected), and thus there exists an index $n \geq 0$ such that $B_n = u_n$. But then we have $(B_n, y_n) \in Y$, which implies $y_n \in B_n^{\mathcal{I}}$ by our assumptions on Y. Thus, our original assumption $y \notin B^{\mathcal{I}}$ is refuted, which completes the proof that property (b) holds, and thus the proof of $(2 \to 1)$ of Theorem 11.

Proof of $(1 \Rightarrow 2)$ of Theorem 21

Assume that $\mathcal{A} \models_{\mathcal{T}} A(a)$. The $\mathcal{EL}$-description graph $\mathcal{G}_{\mathcal{A}}$ can be viewed as the graph of a primitive interpretation. Thus, let $\mathcal{J}$ be this primitive interpretation (i.e., $\mathcal{G}_{\mathcal{A}} = \mathcal{G}_{\mathcal{J}}$) where each individual a in $\mathcal{A}$ interprets itself (i.e., $a^{\mathcal{J}} = a$). First, we construct an interpretation $\mathcal{I}_0$ based on $\mathcal{J}$ that satisfies the prerequisites of Proposition 20, i.e.,

- $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_{\mathcal{J}} \mathcal{I}_0$ and
- $B(b) \in \mathcal{A}$ implies $b^{\mathcal{J}} \in B^{\mathcal{I}_0}$.

The construction of $\mathcal{I}_0$ is similar to the one done in the proof of $(1 \Rightarrow 2)$ of Theorem 29 in [1]. In order to define $\mathcal{I}_0$, we introduce an appropriate simulation $Y\colon \mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{A}} = \mathcal{G}_{\mathcal{J}}$, and then define for all defined concepts B of $\mathcal{T}$:

$$(*) \qquad B^{\mathcal{I}_0} := \{u \mid (B, u) \in Y\}.$$

We define $Y := \bigcup_{n \geq 0} Y_n$, where the relations Y_n are defined by induction on n:

$$Y_0 := \{(B, B) \mid B \text{ is a defined concept in } \mathcal{T}\} \cup \{(B, b) \mid B(b) \in \mathcal{A}\}.$$

If Y_n is already defined, then

$$\begin{array}{rl} Y_{n+1} := Y_n \cup \{(C, u) \mid & (1)\ L_{\mathcal{T}}(C) \subseteq L_{\mathcal{A}}(u), \\ & (2)\ (C, r_1, C_1), \ldots, (C, r_\ell, C_\ell) \text{ are all the edges in } \mathcal{G}_{\mathcal{T}} \\ & \quad\ \text{with source } C, \text{ and} \\ & (3)\ \text{there are edges } (u, r_1, u_1), \ldots, (u, r_\ell, u_\ell) \text{ in } \mathcal{G}_{\mathcal{A}} \\ & \quad\ \text{such that } (C_1, u_1) \in Y_n, \ldots, (C_\ell, u_\ell) \in Y_n \qquad \}. \end{array}$$

Lemma 22 *Y is a simulation.*

Proof. We show by induction on n that all the relations Y_n are simulations. Since the set of all simulations from $\mathcal{G}_\mathcal{T}$ to $\mathcal{G}_\mathcal{A}$ is closed under arbitrary unions, this implies that Y is a simulation.

($n = 0$) For $(B, B) \in Y_0$, properties (S1) and (S2) of Definition 6 are satisfied since $\mathcal{G}_\mathcal{T}$ is a subgraph of $\mathcal{G}_\mathcal{A}$. Now, assume that $(B, b) \in Y_0$, i.e., $B(b) \in \mathcal{A}$.

(S1) If $P \in L_\mathcal{T}(B)$, then $B(b) \in \mathcal{A}$ yields $P \in L_\mathcal{A}(b)$.

(S2) If $(B, r, B') \in E_\mathcal{T}$, then $B(b) \in \mathcal{A}$ yields $(b, r, B') \in E_\mathcal{A}$. In addition, we have $(B', B') \in Y_0$.

($n \to n+1$) The induction step is identical to the corresponding step in the proof of Lemma 32 in [1]. □

Now, let $\mathcal{I}_0$ be the interpretation based on $\mathcal{J}$ defined by the identity $(*)$ above.

Lemma 23 $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_\mathcal{J} \mathcal{I}_0$.

The proof of this lemma is identical to the proof of Lemma 33 in [1].

Lemma 24 *$B(b) \in \mathcal{A}$ implies $b^\mathcal{J} \in B^{\mathcal{I}_0}$.*

Proof. By definition of Y, $B(b) \in \mathcal{A}$ yields $(B, b) \in Y$, and thus $b^\mathcal{J} = b \in B^{\mathcal{I}_0}$ by definition of $\mathcal{I}_0$. □

In the following, let $\mathcal{I}$ be the $\mathcal{I}_0$-gfp-model of $\mathcal{T}$.

Lemma 25 *$\mathcal{I}$ is a model of $\mathcal{A}$.*

Proof. Because of the previous two lemmas, $\mathcal{I}_0$ satisfies the prerequisites of Proposition 20. Thus, it is sufficient to show that there exists a simulation $Z\colon \mathcal{G}_\mathcal{A} \stackrel{\sim}{\rightarrow} \mathcal{G}_\mathcal{J} = \mathcal{G}_\mathcal{A}$ that respects ABox individuals and satisfies

$$(B, u) \in Z \Rightarrow u \in B^{\mathcal{I}_0}$$

for all defined concepts B in $\mathcal{T}$ and all elements $u \in \Delta^\mathcal{J}$. We define Z as follows:

$$Z := Y \cup \{(a, a) \mid a \text{ is an individual in } \mathcal{A}\}.$$

The relation Z is a simulation. In fact, for tuples from Y, (S1) and (S2) are satisfied since Y is a simulation. For tuples $(a, a) \in Z$ (where a is an individual in $\mathcal{A}$), (S1) is trivially satisfied. To show (S2), assume that $(a, r, u) \in E_{\mathcal{A}}$. Then it is enough to show that $(u, u) \in Z$. If u is an individual in $\mathcal{A}$, then $(u, u) \in Z$ by definition of Z. If u is a defined concept in $\mathcal{T}$, then $(u, u) \in Y_0 \subseteq Y \subseteq Z$.

By its definition, Z respects ABox individuals. In addition, if B is a defined concept in $\mathcal{T}$, then $(B, u) \in Z$ implies $(B, u) \in Y$, and thus $u \in B^{\mathcal{I}_0}$. □

Lemma 26 $a = a^{\mathcal{J}} \in A^{\mathcal{I}}$.

Proof. Since we have assumed that $\mathcal{A} \models_{\mathcal{T}} A(a)$, this is an immediate consequence of the fact that $\mathcal{I}$ is a model of $\mathcal{A}$. □

The next lemma completes the proof of $(1 \Rightarrow 2)$ of Theorem 21.

Lemma 27 *The simulation* $Y \colon \mathcal{G}_{\mathcal{T}} \vec{\approx} \mathcal{G}_{\mathcal{A}}$ *satisfies*

(i) $(A, a) \in Y$.

(ii) Y *is* (B, u)*-synchronized for all* $(B, u) \in Y$.

Proof. (i) By Proposition 16, $a = a^{\mathcal{J}} \in A^{\mathcal{I}}$ implies that there exists a simulation $X \colon \mathcal{G}_{\mathcal{T}} \vec{\approx} \mathcal{G}_{\mathcal{J}} = \mathcal{G}_{\mathcal{A}}$ such that

- $(A, a) \in X$;
- if $(B, y) \in X$ then $y \in B^{\mathcal{I}_0}$.

In particular, the two properties of X imply that $a \in A^{\mathcal{I}_0}$, and thus $(A, a) \in Y$ by the definition of $\mathcal{I}_0$.

(ii) Assume that $(B, u) \in Y$. To show that Y is (B, u)-synchronized, we define an appropriate selection function S. Thus, consider the following partial (B, u)-simulation chain:

$$
\begin{array}{ccccccccccc}
B & = & B_0 & \stackrel{r_1}{\to} & B_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & B_{n-1} & \stackrel{r_n}{\to} & B_n \\
 & & Y\downarrow & & Y\downarrow & & & & Y\downarrow & & \\
u & = & u_0 & \stackrel{r_1}{\to} & u_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & u_{n-1} & &
\end{array}
$$

Let k be minimal with $(B_{n-1}, u_{n-1}) \in Y_k$.

Case 1: $k = 0$. If $B_{n-1} = u_{n-1}$, then the selection function S chooses $u_n := B_n$. Otherwise, $B_{n-1}(u_{n-1}) \in \mathcal{A}$, and thus $(B_{n-1}, B_n) \in E_{\mathcal{T}}$ implies $(u_{n-1}, B_n) \in E_{\mathcal{A}}$. Consequently, the selection function can again choose $u_n := B_n$. In both cases we have $(B_n, B_n) \in Y_0 \subseteq Y$.

Case 2: $k > 0$. The minimality of k implies that $(B_{n-1}, u_{n-1}) \in Y_k \setminus Y_{k-1}$. By definition of Y_k, the existence of the edge $(B_{n-1}, r_n, B_n) \in E_{\mathcal{T}}$ thus implies that there is an u_n such that $(u_{n-1}, r_n, u_n) \in E_{\mathcal{A}}$ and $(B_n, u_n) \in Y_{k-1}$. The selection function S chooses such an u_n.

It remains to be shown that the selection function S really satisfies the condition stated in Definition 10.[3] Thus, consider the following infinite S-selected (B, u)-simulation chain:

$$
\begin{array}{ccccccccccc}
B & = & B_0 & \stackrel{r_1}{\to} & B_1 & \stackrel{r_2}{\to} & B_2 & \stackrel{r_3}{\to} & B_3 & \stackrel{r_4}{\to} & \cdots \\
 & & Y\!\downarrow & & Y\!\downarrow & & Y\!\downarrow & & Y\!\downarrow & & \\
u & = & u_0 & \stackrel{r_1}{\to} & u_1 & \stackrel{r_2}{\to} & u_2 & \stackrel{r_3}{\to} & u_3 & \stackrel{r_4}{\to} & \cdots
\end{array}
$$

Let k_0 be minimal with $(B_0, u_0) \in Y_{k_0}$. If $k_0 = 0$, then we are done (see below). Otherwise, $k_0 > 0$ and then we know that $(B_1, u_1) \in Y_{k_0 - 1}$. Thus, if k_1 is minimal with $(B_1, u_1) \in Y_{k_1}$, then $k_0 > k_1$. If we continue this argument, then we obtain indices $k_0, k_1, k_2, \ldots$ where either $k_i > k_{i+1}$ or $k_i = 0$. This shows that there exists an n such that $k_n = 0$. Thus, either $B_n = u_n$ (in which case we are done), or $B_n(u_n) \in \mathcal{A}$. In the second case, the definition of the selection function S yields $B_{n+1} = u_{n+1}$. □

The complexity of the instance problem

In order to show that the instance problem is tractable, it remains to be shown that the existence of a synchronized simulation relation satisfying the conditions stated in (2) of Theorem 21 can be decided in polynomial time.

Corollary 28 *The instance problem w.r.t. descriptive semantics in $\mathcal{EL}$ can be decided in polynomial time.*

Obviously, the simulation Y defined in the proof of $(1 \Rightarrow 2)$ of Theorem 21 can be computed in polynomial time. Thus, the above corollary is an immediate consequence of the following proposition.

Proposition 29 *The following are equivalent:*

1. *There exists a simulation Z satisfying the conditions stated in* (2) *of Theorem 21.*
2. *The simulation Y defined in the proof of* $(1 \Rightarrow 2)$ *of Theorem 21 satisfies* $(A, a) \in Y$.

[3] Definition 10 actually states this condition for simulations from $\mathcal{G}_{\mathcal{T}}$ to $\mathcal{G}_{\mathcal{T}}$. It is, however, obvious that this can be extended to simulations from $\mathcal{G}_{\mathcal{T}}$ to $\mathcal{G}_{\mathcal{A}}$.

Proof. First, assume that there exists a simulation Z satisfying the conditions stated in (2) of Theorem 21. Then Theorem 21 yields $\mathcal{A} \models_{\mathcal{T}} A(a)$. But then the proof of ($1 \Rightarrow 2$) of Theorem 21 shows that $(A, a) \in Y$ (see (i) of Lemma 27).

Second, assume that $(A, a) \in Y$. The proof of (ii) of Lemma 27 shows that Y is (B, u)-synchronized for all $(B, u) \in Y$, and thus Y satisfies the conditions stated in (2) of Theorem 21. $\square$

A stronger version of Theorem 21

Proposition 29 also allows us to strengthen the formulation of Theorem 21. This stronger version will be useful in the next section.

Let Z: $\mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{A}}$ be a simulation. Recall that a selection function S for B, u and Z assigns to each partial (B, u)-simulation chain

$$\begin{array}{ccccccccccc}
B & = & B_0 & \stackrel{r_1}{\to} & B_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & B_{n-1} & \stackrel{r_n}{\to} & B_n \\
 & & Z{\downarrow} & & Z{\downarrow} & & & & Z{\downarrow} & & \\
u & = & u_0 & \stackrel{r_1}{\to} & u_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & u_{n-1} & &
\end{array}$$

a node u_n such that (u_{n-1}, r_n, u_n) is an edge in $\mathcal{G}_{\mathcal{T}}$ and $(B_n, u_n) \in Z$.

Definition 30 We call a selection function S *nice* iff it satisfies the following two conditions:

1. It is *memoryless*, i.e., its result u_n depends only on $B_{n-1}, u_{n-1}, r_n, B_n$, and not on the other parts of the partial (B, u)-simulation chain.
2. If $B_{n-1} = u_{n-1}$, then its result u_n is just B_n.

The simulation relation Z is called *strongly (B, u)-synchronized* iff there exists a *nice* selection function S for B, u and Z such that the following holds: for every infinite S-selected (B, u)-simulation chain

$$\begin{array}{ccccccccccc}
B & = & B_0 & \stackrel{r_1}{\to} & B_1 & \stackrel{r_2}{\to} & B_2 & \stackrel{r_3}{\to} & B_3 & \stackrel{r_4}{\to} & \cdots \\
 & & Z{\downarrow} & & Z{\downarrow} & & Z{\downarrow} & & Z{\downarrow} & & \\
u & = & u_0 & \stackrel{r_1}{\to} & u_1 & \stackrel{r_2}{\to} & u_2 & \stackrel{r_3}{\to} & u_3 & \stackrel{r_4}{\to} & \cdots
\end{array}$$

there exists an $i \geq 0$ such that $u_i = B_i$.

The following corollary is an immediate consequence of Proposition 29 since the simulation Y defined in the proof of ($1 \Rightarrow 2$) of Theorem 21 is strongly (B, u)-synchronized for all $(B, u) \in Y$ (see the definition of the selection function S in the proof of Lemma 27).

Corollary 31 *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: $\mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{A}}$ such that*
   - $(A, a) \in Z$.
   - *Z is* strongly *(B, u)-synchronized for all $(B, u) \in Z$.*

As shown in [4], strongly (B, u)-synchronized simulations satisfy the following property:

Lemma 32 *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{A}$ an $\mathcal{EL}$-ABox, and $\mathcal{G}_{\mathcal{A}}$ the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}$, as introduced in Definition 17. Assume that $\mathcal{T}$ contains n defined concepts, and that $\mathcal{G}_{\mathcal{A}}$ contains m nodes. If Z: $\mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{A}}$ is strongly (B, u)-synchronized with nice selection function S, then for any infinite S-selected (B, u)-simulation chain of the form depicted in Definition 30 there exists $k < mn$ such that $B_k = u_k$.*

The proof of this lemma is very similar to the proof of Lemma 27 in [4].

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 33 *Let $\mathcal{T}_1 = \emptyset$ and $\mathcal{A} = \{r(a, a)\}$. Then a 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 a. Let $\mathcal{G}_{\mathcal{A}}$ be the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}_1$, as introduced in Definition 17. Since a is an instance of E, there is a simulation $Z\colon \mathcal{G}_{\mathcal{T}_2} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{A}}$ such that $(E, a) \in Z$ 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 a to a defined concept in $\mathcal{T}_2$. Thus, the fact that Z is (E, a)-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 uppers-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 $\mathcal{A} \models_{\mathcal{T}_3} F_n(a)$. Since E is an msc of a, this implies that $E \sqsubseteq_{\mathcal{T}_3} F_n$. Thus, there is an (F_n, E)-synchronized simulation $Y\colon \mathcal{G}_{\mathcal{T}_3} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{T}_3}$ 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. $\square$

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 17.

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 $\mathcal{T}_1$. We will show in this subsection that, w.r.t. descriptive semantics, this is the case iff the ABox satisfies a certain acyclicity condition.

Proposition 34 *The defined concept C_a in $\mathcal{T}_2$ is the msc of a in $\mathcal{A}$ and $\mathcal{T}_1$ iff $\mathcal{A} \models_{\mathcal{T}_2} C_a(a)$.*

Proof. The direction from left to right is obvious.

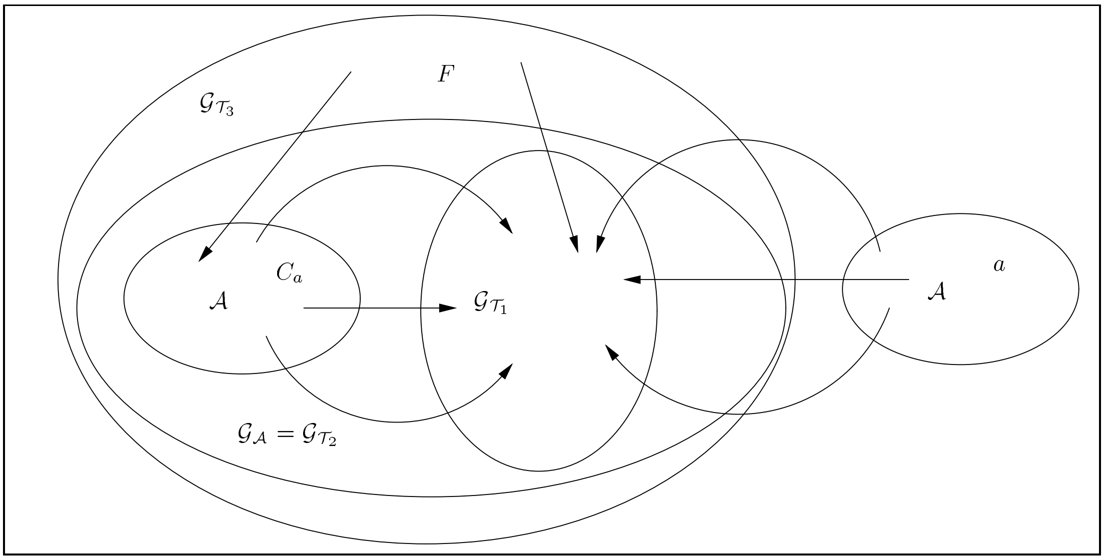

Figure 3: The $\mathcal{EL}$-description graph $\widehat{\mathcal{G}}_{\mathcal{A}}$.

To show the direction from right to left, assume that $\mathcal{A} \models_{\mathcal{T}_2} C_a(a)$. We must show that the second condition in Definition 3 is also satisfied. Thus, assume that $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_2$ and that F is a defined concept in $\mathcal{T}_3$ such that $\mathcal{A} \models_{\mathcal{T}_3} F(a)$. Let $\widehat{\mathcal{G}}_{\mathcal{A}}$ be the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}_3$, as introduced in Definition 17. By Theorem 21, $\mathcal{A} \models_{\mathcal{T}_3} F(a)$ implies that there is a simulation Z: $\mathcal{G}_{\mathcal{T}_3} \stackrel{\rightarrow}{\sim} \widehat{\mathcal{G}}_{\mathcal{A}}$ such that $(F, a) \in Z$ and $(B, u) \in Z$ for a defined concept B of $\mathcal{T}_3$ implies that Z is (B, u)-synchronized.

We must show that $C_a \sqsubseteq_{\mathcal{T}_3} F$. By Theorem 11, it is enough to show that there is an (F, C_a)-synchronized simulation Y: $\mathcal{G}_{\mathcal{T}_3} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{T}_3}$ such that $(F, C_a) \in Y$.

To define Y, first note that the set of nodes of $\widehat{\mathcal{G}}_{\mathcal{A}}$ consists of the nodes of $\mathcal{G}_{\mathcal{T}_3}$ and the individuals occurring in $\mathcal{A}$. Also note that $\mathcal{T}_3$ extends $\mathcal{T}_2$, and that $\mathcal{G}_{\mathcal{T}_2}$ in principle also contains the individuals occurring in $\mathcal{A}$. However, we assume without loss of generality that the individual names b in $\mathcal{T}_2$ have been renamed into concept names C_b. The definition of $\widehat{\mathcal{G}}_{\mathcal{A}}$ is illustrated in Figure 3. The arrows indicate that there may be edges from one subgraph into the other. The inner oval marked with $\mathcal{A}$ indicates the ABox $\mathcal{A}$ as used within $\mathcal{G}_{\mathcal{T}_2}$. There, the individual name a is renamed into C_a (and an analogous renaming is done for the other individual names).

The simulation Y is defined as follows:

$$\begin{aligned} Y \;:=\; & \{(u, v) \in Z \mid v \text{ is a node of } \mathcal{G}_{\mathcal{T}_3}\} \;\cup \\ & \{(u, C_b) \mid (u, b) \in Z \text{ and } b \text{ is an individual name in } \mathcal{A}\}. \end{aligned}$$

Since $(F, a) \in Z$, we have $(F, C_a) \in Y$. The proof of Lemma 49 in [4] shows that Y is a simulation relation. It remains to be shown that Y is (F, C_a)-synchronized. This means that we must define an appropriate selection function S such that all infinite S-selected (F, C_a)-simulation chains satisfy the synchronization property. Basically, S is induced by the selection function S' that ensures that $Z\colon \mathcal{G}_{\mathcal{T}_3} \rightleftarrows\!\!\sim \widehat{\mathcal{G}}_{\mathcal{A}}$ is (F, a)-synchronized. To be more precise, consider the partial (F, C_a)-simulation chain

$$\begin{array}{ccccc} F & = & F_0 & \xrightarrow{r_1} & F_1 \\ & & Y\downarrow & & \\ C_a & = & C_0 & & \end{array}$$

Then there exists a corresponding partial (F, a)-simulation chain

$$\begin{array}{ccccc} F & = & F_0 & \xrightarrow{r_1} & F_1 \\ & & Z\downarrow & & \\ a & = & u_0 & & \end{array}$$

The selection function S' yields a node u_1 such that (u_0, r_1, u_1) is an edge in $\widehat{\mathcal{G}}_{\mathcal{A}}$ and $(F_1, u_1) \in Z$. The node u_1 is either an individual in $\mathcal{A}$ or a defined concept in $\mathcal{T}_1$ (see Figure 3).

- If $u_1 = b_1$ is an individual in $\mathcal{A}$, then S selects $C_1 := C_{b_1}$. We have that (C_a, r_1, C_{b_1}) is an edge in $\mathcal{G}_{\mathcal{T}_3}$ and $(F_1, C_{b_1}) \in Y$.
- If $u_1 = C_1$ is a defined concept in $\mathcal{T}_1$ (and thus a node of $\mathcal{G}_{\mathcal{T}_3}$), then S selects C_1. We have that (C_a, r_1, C_1) is an edge in $\mathcal{G}_{\mathcal{T}_3}$ and $(F_1, C_1) \in Y$.

Now, consider the partial S-selected (F, C_a)-simulation chain

$$\begin{array}{ccccccccccc} F & = & F_0 & \xrightarrow{r_1} & F_1 & \xrightarrow{r_2} & \cdots & \xrightarrow{r_{n-1}} & F_{n-1} & \xrightarrow{r_n} & F_n \\ & & Y\downarrow & & Y\downarrow & & & & Y\downarrow & & \\ C_a & = & C_0 & \xrightarrow{r_1} & C_1 & \xrightarrow{r_2} & \cdots & \xrightarrow{r_{n-1}} & C_{n-1} & & \end{array}$$

By induction, we may assume that there is a corresponding partial S'-selected (F, a)-simulation chain

$$\begin{array}{ccccccccccc} F & = & F_0 & \xrightarrow{r_1} & F_1 & \xrightarrow{r_2} & \cdots & \xrightarrow{r_{n-1}} & F_{n-1} & \xrightarrow{r_n} & F_n \\ & & Z\downarrow & & Z\downarrow & & & & Z\downarrow & & \\ a & = & u_0 & \xrightarrow{r_1} & u_1 & \xrightarrow{r_2} & \cdots & \xrightarrow{r_{n-1}} & u_{n-1} & & \end{array}$$

where

- $u_i = a_i$ is an individual in $\mathcal{A}$, in which case $C_i = C_{a_i}$, or
- u_i is a defined concept in $\mathcal{T}_1$, in which case $C_i = u_i$ and all u_j for $i < j < n$ are also defined concepts in $\mathcal{T}_1$.

The selection function S' yields a node u_n such that (u_{n-1}, r_n, u_n) is an edge in $\widehat{\mathcal{G}}_{\mathcal{A}}$ and $(F_n, u_n) \in Z$. The node u_n is either an individual in $\mathcal{A}$ or a defined concept in $\mathcal{T}_1$. We can now proceed as in the case $n = 1$ above, i.e.,

- If $u_n = b_n$ is an individual in $\mathcal{A}$, then S selects C_{b_n}. In this case u_{n-1} is an ABox individual b_{n-1} and $C_{n-1} = C_{b_{n-1}}$. We have that $(C_{b_{n-1}}, r_n, C_{b_n})$ is an edge in $\mathcal{G}_{\mathcal{T}_3}$ and $(F_n, C_{b_n}) \in Y$.

- If $u_n = C_n$ is a defined concept in $\mathcal{T}_1$ (and thus a node of $\mathcal{G}_{\mathcal{T}_3}$), then S selects C_n. It is easy to see that (C_{n-1}, r_n, C_n) is an edge in $\mathcal{G}_{\mathcal{T}_3}$ and $(F_n, C_n) \in Y$.

Now, consider an infinite S-selected (F, C_a)-simulation chain:

$$\begin{array}{ccccccccccc} F & = & F_0 & \stackrel{r_1}{\to} & F_1 & \stackrel{r_2}{\to} & F_2 & \stackrel{r_3}{\to} & F_3 & \stackrel{r_4}{\to} & \cdots \\ & & Y{\downarrow} & & Y{\downarrow} & & Y{\downarrow} & & Y{\downarrow} & & \\ C_a & = & C_0 & \stackrel{r_1}{\to} & C_1 & \stackrel{r_2}{\to} & C_2 & \stackrel{r_3}{\to} & C_3 & \stackrel{r_4}{\to} & \cdots \end{array}$$

Then there is a corresponding infinite S'-selected (F, a)-simulation chain

$$\begin{array}{ccccccccccc} F & = & F_0 & \stackrel{r_1}{\to} & F_1 & \stackrel{r_2}{\to} & F_2 & \stackrel{r_3}{\to} & F_3 & \stackrel{r_4}{\to} & \cdots \\ & & Z{\downarrow} & & Z{\downarrow} & & Z{\downarrow} & & Z{\downarrow} & & \\ a & = & u_0 & \stackrel{r_1}{\to} & u_1 & \stackrel{r_2}{\to} & u_2 & \stackrel{r_3}{\to} & u_3 & \stackrel{r_4}{\to} & \cdots \end{array}$$

where

- $u_i = a_i$ is an individual in $\mathcal{A}$, in which case $C_i = C_{a_i}$, or

- u_i is a defined concept in $\mathcal{T}_1$, in which case $C_i = u_i$ and all u_j for $i < j$ are also defined concepts in $\mathcal{T}_1$.

Since Z is (F, a)-synchronized, there is an $i \geq 0$ such that $F_i = u_i$. Since F_i is a node in $\mathcal{G}_{\mathcal{T}_3}$, this implies that u_i is a node in $\mathcal{G}_{\mathcal{T}_3}$. However, this can only be the case if u_i is a defined concept in $\mathcal{T}_1$, in which case $C_i = u_i$. Thus, we have $F_i = C_i$, which shows that Y is (F, C_a)-synchronized. □

Next, we show that $\mathcal{A} \models_{\mathcal{T}_2} C_a(a)$ holds iff $\mathcal{A}$ does not contain a cycle that is reachable from a.

Definition 35 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 $a_j = a_n$.

Proposition 36 *$\mathcal{A} \models_{\mathcal{T}_2} C_a(a)$ iff $\mathcal{A}$ is a-acyclic.*

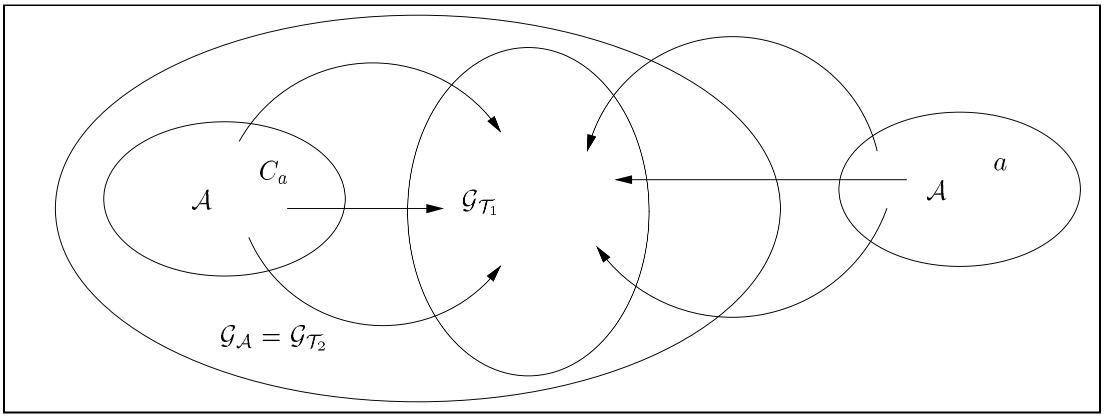

Figure 4: The $\mathcal{EL}$-description graph $\tilde{\mathcal{G}}_{\mathcal{A}}$.

Proof. First, assume that $\mathcal{A} \models_{\mathcal{T}_2} C_a(a)$. Let $\tilde{\mathcal{G}}_{\mathcal{A}}$ be the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}_2$, as introduced in Definition 17 (see Figure 4). By Theorem 21, there is a simulation $Z\colon \mathcal{G}_{\mathcal{T}_2} \overset{\rightarrow}{\sim} \tilde{\mathcal{G}}_{\mathcal{A}}$ such that

- $(C_a, a) \in Z$.
- Z is (B, u)-synchronized for all $(B, u) \in Z$.

If there is a cycle in $\mathcal{A}$ that is reachable from a, then there are individuals $a_1, a_2, \ldots$ and roles $r_1, r_2, \ldots$ such that

$$C_a \xrightarrow{r_1} C_{a_1} \xrightarrow{r_2} C_{a_2} \xrightarrow{r_3} C_{a_3} \xrightarrow{r_4} \cdots$$

is an infinite path in $\mathcal{G}_{\mathcal{A}} = \mathcal{G}_{\mathcal{T}_2}$. Since Z is (C_a, a)-synchronized, this path can be simulated by a path in $\tilde{\mathcal{G}}_{\mathcal{A}}$ starting with a such that the synchronization property is satisfied. However, this cannot be the case since from a one can reach only defined concepts in $\mathcal{T}_1$, and thus none of the concepts C_{a_i}.

Second, assume that $\mathcal{A}$ is a-acyclic. In order to show $\mathcal{A} \models_{\mathcal{T}_2} C_a(a)$ we define a simulation $Z\colon \mathcal{G}_{\mathcal{T}_2} \overset{\rightarrow}{\sim} \tilde{\mathcal{G}}_{\mathcal{A}}$ and show that it satisfies the properties stated in (2) of Theorem 21:

$$\begin{aligned} Z \;:=\; & \{(C_b, b) \mid b \text{ is reachable in } \mathcal{A} \text{ from } a\} \;\cup \\ & \{(C, C) \mid C \text{ is a defined concept in } \mathcal{T}_1\}. \end{aligned}$$

Since a is trivially reachable from a, we have $(C_a, a) \in Z$. In addition, it is easy to see that Z is a simulation. Thus it remains to be shown that Z is (B, u)-synchronized for all $(B, u) \in Z$.

If $(B, u) = (C, C)$ for a defined concept C in $\mathcal{T}_1$, then this is obvious since Z is the identity on defined concepts in $\mathcal{T}_1$.

If $(B, u) = (C_b, b)$ for an individual b that is reachable in $\mathcal{A}$ from a, then any infinite path starting with C_b must eventually lead to a defined concept in $\mathcal{T}_1$. In

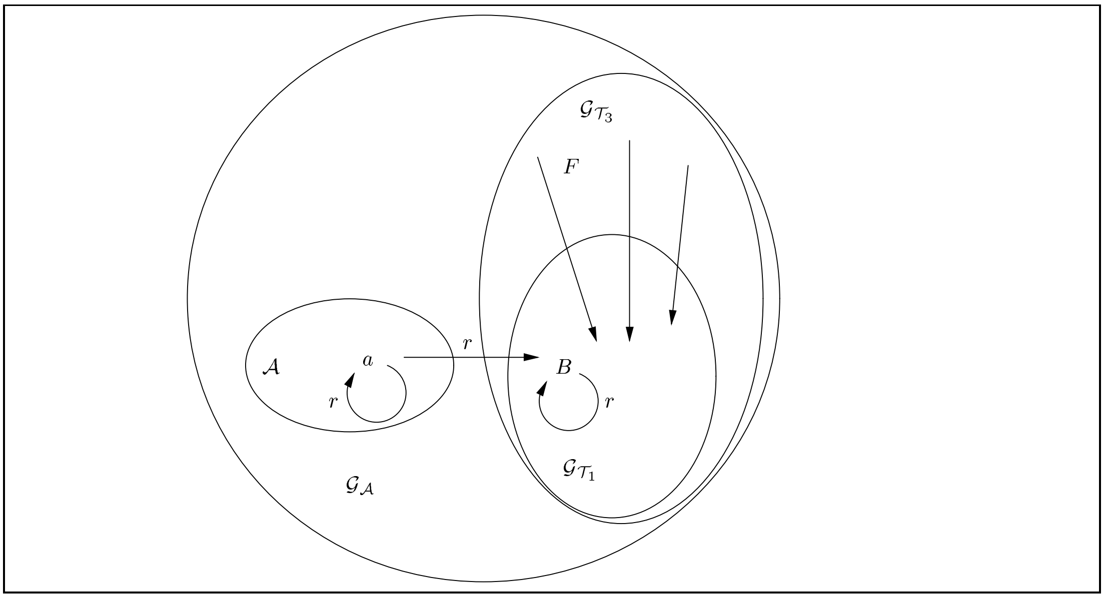

Figure 5: The $\mathcal{EL}$-description graph $\mathcal{G}_{\mathcal{A}}$ in the proof of Proposition 38.

fact, otherwise there is an infinite (and thus cyclic) path in $\mathcal{A}$ starting with b. Since b is reachable from a, this contradicts our assumption that $\mathcal{A}$ is a-acyclic. Now, we can again use the fact that Z is the identity on defined concepts in $\mathcal{T}_1$. □

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.

Theorem 37 *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 example shows that this is not a necessary condition.

Proposition 38 *Let $\mathcal{T}_1 = \{B \equiv \exists r.B\}$ and $\mathcal{A} = \{r(a, a), B(a)\}$. Then B in $\mathcal{T}_1$ is the msc of a in $\mathcal{A}$ and $\mathcal{T}_1$.*

Proof. 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 17 (see Figure 5). Since $\mathcal{A} \models_{\mathcal{T}_3} F(a)$, there is a simulation Z: $\mathcal{G}_{\mathcal{T}_3} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{A}}$ 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 $\mathcal{G}_{\mathcal{T}_3}$.[4]

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. □

Since the ABox $\mathcal{A}$ in Proposition 38 is obviously not a-acyclic, a-acyclicity of $\mathcal{A}$ is not a necessary condition for the existence of the msc.

Corollary 39 *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.*

5.3 Characterizing when the msc exists

The example that demonstrates the non-existence of the msc given above (see Theorem 33) shows that cycles in the ABox are problematic. However, Proposition 38 shows that not all cycles cause problems. Intuitively, the reason for

[4] 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 $\mathcal{T}_1$.

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 40 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 show below that the msc of a exists iff there is a k such that P_k is the msc.[5] Unfortunately, it is not clear how this condition can be decided.

Definition 40 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:

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

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

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

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 Proposition 38. The corresponding graph $\mathcal{G}_\mathcal{A}^{(2)}$ is depicted in Figure 6 (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 6 consists of the following definitions (where nodes corresponding to individuals have been renamed[6]):

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

Lemma 41 $\mathcal{A} \models_{\mathcal{T}_2^{(k)}} P_k(a)$

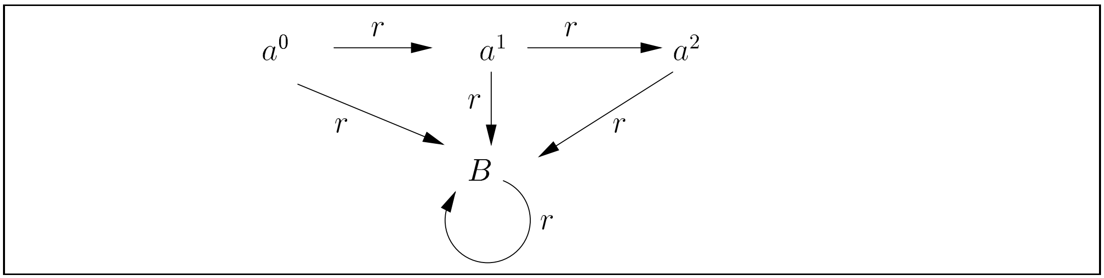

Figure 6: The $\mathcal{EL}$-description graph $\mathcal{G}_{\mathcal{A}}^{(2)}$ of the example in Proposition 38.

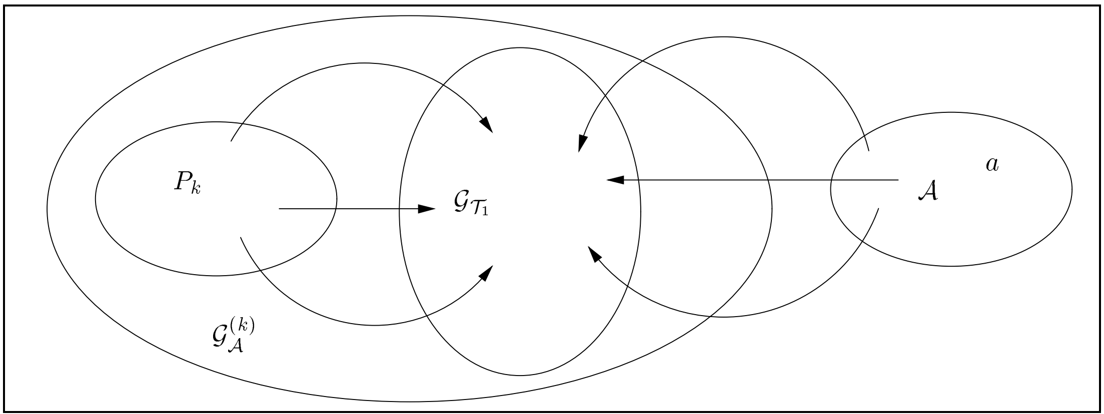

Figure 7: The $\mathcal{EL}$-description graph $\mathcal{G}_{\mathcal{A}}$ in the proof of Lemma 41.

Proof. Let $\mathcal{G}_{\mathcal{A}}$ be the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}_2^{(k)}$, as introduced in Definition 17 (see Figure 7). We must show that there is a simulation $Z\colon \mathcal{G}_{\mathcal{T}_2^{(k)}} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{A}}$ such that $(P_k, a) \in Z$ and Z is (B, u)-synchronized for all $(B, u) \in Z$. We define the relation Z as follows:

$$Z := \{(b^n, b) \mid b \in V_k \setminus V_{\mathcal{T}_1}\} \cup \{(C, C) \mid C \text{ is a defined concept in } \mathcal{T}_1\}.$$

By definition of Z, $(a^0, a) \in Z$. Since P_k is simply our name for a^0 in $\mathcal{G}_{\mathcal{T}_2^{(k)}}$, this shows that $(P_k, a) \in Z$. It is straightforward to show that Z is a simulation. In addition, every infinite path in $\mathcal{G}_{\mathcal{T}_2^{(k)}}$ must eventually lead to a node in $\mathcal{G}_{\mathcal{T}_1}$. Since Z is the identity on these nodes, the synchronization property obviously follows. □

What we want to show next is 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 $\mathcal{G}_{\mathcal{T}_2} = (V_{\mathcal{T}_2}, E_{\mathcal{T}_2}, L_{\mathcal{T}_2})$ and $\mathcal{G}_{\mathcal{A}}$ be the $\mathcal{EL}$-description graph corresponding to $\mathcal{A}$ and $\mathcal{T}_2$, as introduced in Definition 17. Then $\mathcal{A} \models_{\mathcal{T}_2} F(a)$

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

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

implies that there is a simulation $Y\colon \mathcal{G}_{\mathcal{T}_2} \stackrel{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{A}}$ such that $(F, a) \in Y$ and Y is (B, u)-synchronized for all $(B, u) \in Y$.

By Corollary 31 we may assume without loss of generality that the selection functions that ensure synchronization are nice, i.e., Y is strongly (B, u)-synchronized for all $(B, u) \in Y$. Consequently, if $k = |V_2|m$ where m is the number of nodes of $\mathcal{G}_{\mathcal{A}}$, then Lemma 32 shows that the selection functions ensure synchronization after less than k steps.

In the following, let $k := |V_{\mathcal{T}_2}|m$ where m is the number of nodes of $\mathcal{G}_{\mathcal{A}}$ (i.e., $|V_{\mathcal{T}_2}|$ plus the number of individual names occurring 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 11 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 42 *If $k := |V_{\mathcal{T}_2}|m$, then $P_k \sqsubseteq_{\mathcal{T}_3} F$.*

Proof. We need an (F, P_k)-synchronized simulation $Z\colon \mathcal{G}_{\mathcal{T}_3} \stackrel{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{T}_3}$ such that $(F, P_k) \in Z$. We define the relation Z as follows:

$$\begin{aligned} Z \;:=\; & \{(u, b^n) \mid u \text{ is a node in } \mathcal{G}_{\mathcal{T}_2} \text{ and } b^n \in V_k \setminus V_{\mathcal{T}_1} \text{ with } (u, b) \in Y\} \;\cup \\ & \{(u, v) \mid u \text{ is a node in } \mathcal{G}_{\mathcal{T}_2} \text{ and } v \in V_{\mathcal{T}_1} \text{ with } (u, v) \in Y\}. \end{aligned}$$

The proof that Z is indeed an (F, P_k)-synchronized simulation such that $(F, P_k) \in Z$ is similar to the proof of Lemma 30 in [4]. We give it here for the sake of completeness.

By definition of Z, $(F, a) \in Y$ implies $(F, P_k) \in Z$ since P_k stands for $a^0 \in V_k \setminus V_{\mathcal{T}_1}$. In order to show that Z is (F, P_k)-synchronized, we must define an appropriate selection function S. Thus, consider the following partial (F, P_k)-simulation chain:

$$\begin{array}{rcccccccccc} F & = & F_0 & \stackrel{r_1}{\rightarrow} & F_1 & \stackrel{r_2}{\rightarrow} & \cdots & \stackrel{r_{n-1}}{\rightarrow} & F_{n-1} & \stackrel{r_n}{\rightarrow} & F_n \\ & & Z\downarrow & & Z\downarrow & & & & Z\downarrow & & \\ P_k = a^0 & = & w_0 & \stackrel{r_1}{\rightarrow} & w_1 & \stackrel{r_2}{\rightarrow} & \cdots & \stackrel{r_{n-1}}{\rightarrow} & w_{n-1} & & \end{array}$$

Since $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_2$, the nodes F_i are all nodes in $\mathcal{G}_{\mathcal{T}_2}$. In addition, since $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_2^{(k)}$, the nodes w_i are all nodes of $\mathcal{G}_{\mathcal{A}}^{(k)}$, i.e., elements of V_k.

First, assume that w_{n-1} belongs to $V_k \setminus V_{\mathcal{T}_1}$ (and thus also the other nodes w_i). It is easy to see that this implies that $w_i = b_i^i$ for some individual b_i of $\mathcal{A}$ ($i = 1, \ldots, n-1$). The definition of Z yields $(F_i, b_i) \in Y$ for $i = 1, \ldots, n-1$. Thus, we have the following partial simulation chain:

$$\begin{array}{ccccccccccc} F & = & F_0 & \stackrel{r_1}{\to} & F_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & F_{n-1} & \stackrel{r_n}{\to} & F_n \\ & & Y\downarrow & & Y\downarrow & & & & Y\downarrow & & \\ a & = & b_0 & \stackrel{r_1}{\to} & b_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & b_{n-1} & & \end{array}$$

Since Y is (F, a)-synchronized, the corresponding selection function yields a node w_n of $\mathcal{G}_{\mathcal{T}_3}$ such that $(F_n, w_n) \in Y$ and (b_{n-1}, r_n, w_n) is an edge in $\mathcal{G}_{\mathcal{A}}$. By the definition of $\mathcal{G}_{\mathcal{A}}$, this implies that w_n is a node in $V_{\mathcal{T}_1}$ or an individual name in $\mathcal{A}$.

Case 1: $w_n \in V_{\mathcal{T}_1}$.
In this case, there is also an edge $(b_{n-1}^{n-1}, r_n, w_n)$ in $\mathcal{G}_{\mathcal{T}_3}$, and the selection function chooses w_n.

Case 2: $w_n = b_n$ is an individual name in $\mathcal{A}$.
If $n \leq k$, then $b_n^n \in V_k \setminus V_{\mathcal{T}_1}$, and we have $(F_n, b_n^n) \in Z$ and $(b_{n-1}^{n-1}, r_n, b_n^n)$ is an edge in $\mathcal{G}_{\mathcal{A}}^{(k)}$, and thus also in $\mathcal{G}_{\mathcal{T}_3}$. Thus the selection function chooses b_n^n. The case $n > k$ cannot occur. In fact, our choice of k together with the fact that we have assumed (without loss of generality) that Y is strongly (F, a)-synchronized yields $F_{n-1} = b_{n-1}$ (Lemma 32). However, this cannot be the case since F_{n-1} is a defined concept in $\mathcal{T}_2$ whereas b_{n-1} is an individual name in $\mathcal{A}$.

Now, assume that $w_{n-1} \in V_{\mathcal{T}_1}$. But then $(F_{n-1}, w_{n-1}) \in Y$ by the definition of Z. If i is minimal such that $w_i \in V_{\mathcal{T}_1}$, then we can assume (by induction) that there are individuals $b_1, \ldots, b_{i-1}$ in $\mathcal{A}$ such that $w_1 = b_1^1, \ldots, w_{i-1} = b_{i-1}^{i-1}$ and the following is a partial simulation chain w.r.t. Y:

$$\begin{array}{ccccccccccccccc} F & \stackrel{r_1}{\to} & F_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{i-1}}{\to} & F_{i-1} & \stackrel{r_i}{\to} & F_i & \stackrel{r_{i+1}}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & F_{n-1} & \stackrel{r_n}{\to} & F_n \\ Y\downarrow & & Y\downarrow & & & & Y\downarrow & & Y\downarrow & & & & Y\downarrow & & \\ a & \stackrel{r_1}{\to} & b_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{i-1}}{\to} & b_{i-1} & \stackrel{r_i}{\to} & w_i & \stackrel{r_{i+1}}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & w_{n-1} & & \end{array}$$

The selection function that ensures that Y is (F, a)-synchronized yields a node w_n in $\mathcal{G}_{\mathcal{T}_3}$ such that (w_{n-1}, r_n, w_n) is an edge in $\mathcal{G}_{\mathcal{T}_3}$ and $(F_n, w_n) \in Y$. Since $w_{n-1} \in V_{\mathcal{T}_1}$ and $\mathcal{T}_3$ is a conservative extension of $\mathcal{T}_1$, this implies $w_n \in V_{\mathcal{T}_1}$. Consequently, $(F_n, w_n) \in Y$ also yields $(F_n, w_n) \in Z$. Thus, the selection function S chooses w_n.

To show that Z is (F, P_k)-synchronized, we consider the following infinite S-selected (F, P_k)-synchronization chain

$$\begin{array}{ccccccccccc} F & = & F_0 & \stackrel{r_1}{\to} & F_1 & \stackrel{r_2}{\to} & F_2 & \stackrel{r_3}{\to} & F_3 & \stackrel{r_4}{\to} & \cdots \\ & & Z\downarrow & & Z\downarrow & & Z\downarrow & & Z\downarrow & & \\ P_k & = & w_0 & \stackrel{r_1}{\to} & w_1 & \stackrel{r_2}{\to} & w_2 & \stackrel{r_3}{\to} & w_3 & \stackrel{r_4}{\to} & \cdots \end{array}$$

Since any path in $\mathcal{G}_{\mathcal{A}}^{(k)}$ has length at most k, we know that there exists a minimal $i \leq k$ such that $w_i \in V_{\mathcal{T}_1}$. Thus, there are individuals $b_1, \ldots, b_{i-1}$ in $\mathcal{A}$ such that $w_1 = b_1^1, \ldots, w_{i-1} = b_{i-1}^{i-1}$ and the following is an infinite simulation chain w.r.t. Y:

$$\begin{array}{ccccccccccccccc} F & \xrightarrow{r_1} & F_1 & \xrightarrow{r_2} & \cdots & \xrightarrow{r_{i-1}} & F_{i-1} & \xrightarrow{r_i} & F_i & \xrightarrow{r_{i+1}} & F_{i+1} & \xrightarrow{r_n} & \cdots \\ Y\downarrow & & Y\downarrow & & & & Y\downarrow & & Y\downarrow & & Y\downarrow & & \\ a & \xrightarrow{r_1} & b_1 & \xrightarrow{r_2} & \cdots & \xrightarrow{r_{i-1}} & b_{i-1} & \xrightarrow{r_i} & w_i & \xrightarrow{r_{i+1}} & w_{i+1} & \xrightarrow{r_n} & \cdots \end{array}$$

According to our definition of S, this simulation chain is selected w.r.t. the selection function that ensures that Y is (F, a)-synchronized. Thus, there is an index $j \geq i$ such that $F_j = w_j$ (note that $j < i$ is not possible). □

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 $\mathcal{T}_1$. For example, in addition to the upper index describing the level of a node in $V_k \setminus V_{\mathcal{T}_1}$ we could add a lower index k. Thus, b_k^n denotes a node on level n in $\mathcal{G}_{\mathcal{A}}^{(k)}$.

As a consequence of the two lemmas shown above, we can prove that an msc of a must be equivalent to one of the concepts P_k.

Theorem 43 *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 $\mathcal{T}_1$.*

The proof of this theorem is very similar to the proof of Theorem 31 in [4]. The proofs of the following lemma and corollary are also basically identical to the proofs of Lemma 32 and Corollary 33 in [4].

Lemma 44 *Let $\mathcal{T} := \mathcal{T}_2^{(k)} \cup \mathcal{T}_2^{(k+1)}$. Then $P_{k+1} \sqsubseteq_{\mathcal{T}} P_k$.*

Corollary 45 *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 Proposition 38 (see also Figure 6). 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

Computing the least common subsumer (lcs) and the most specific concept (msc) are important steps in the bottom-up construction of DL knowledge bases. In DLs with existential restrictions, the most specific concept of a given ABox individual need not exist. In [4] we have shown that allowing for cyclic definitions with greatest fixpoint (gfp) semantics in the DL $\mathcal{EL}$ overcomes this problem: in this setting, the most specific concept exists and can be computed in polynomial time. But then subsumption and the lcs operation must also be considered w.r.t. cyclic definitions with gfp-semantics. In [1] we have shown that the subsumption problem remains polynomial if one allows for cyclic definitions in $\mathcal{EL}$, and in [4] we have shown that, w.r.t. gfp-semantics, the lcs always exists, and that the binary lcs can be computed in polynomial time.

Because of these positive results regarding gfp-semantics one might think that this should be the semantics of choice for cyclic definitions in $\mathcal{EL}$. However, the problem is that gfp-semantics is not employed by any of the modern DL systems allowing for cycles. In order to be compatible with these systems, one would need to employ descriptive semantics. Subsumption is also polynomial w.r.t. descriptive semantics [1], and we have shown in the present report that the same is true for the instance problem.

For the lcs (which was treated in [4]) and the msc (which was treated in the present report), descriptive semantics is not that well-behaved: the lcs and the msc need not exist in general. In addition, we were only able to give decidable *sufficient* conditions for their existence. If these conditions apply, then the lcs/msc can be computed in polynomial time. Although we were able to characterize the cases in which the lcs/msc exists, and show how the lcs/msc looks like in these cases, it is not clear how to decide this *necessary and sufficient* condition for the existence of the lcs/msc. Thus, one of the main open problems is the question how to give a decidable characterization of the cases in which the lcs/msc exists, and to determine whether it can then be computed in polynomial time.

References

[1] F. Baader. Terminological cycles in a description logic with existential restrictions. LTCS-Report LTCS-02-02, Chair for Automata Theory, Institute for Theoretical Computer Science, Dresden University of Technology, Germany, 2002. See http://lat.inf.tu-dresden.de/research/reports.html.

[2] Franz Baader. Terminological cycles in KL-ONE-based knowledge representation languages. In *Proc. of the 8th Nat. Conf. on Artificial Intelligence (AAAI'90)*, pages 621–626, Boston (Ma, USA), 1990.

[3] Franz Baader. Using automata theory for characterizing the semantics of terminological cycles. *Ann. of Mathematics and Artificial Intelligence*, 18:175–219, 1996.

[4] Franz Baader. Least common subsumers, most specific concepts, and role-value-maps in a description logic with existential restrictions and terminological cycles. LTCS-Report LTCS-02-07, Chair for Automata Theory, Institute

for Theoretical Computer Science, Dresden University of Technology, Germany, 2002. See http://lat.inf.tu-dresden.de/research/reports.html.

[5] 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.

[6] 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.

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

[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.

[17] Alfred Tarski. A lattice-theoretical fixpoint theorem and its applications. *Pacific Journal of Mathematics*, 5:285–309, 1955.