Computing the least common subsumer in the description logic $\mathcal{EL}$ w.r.t. terminological cycles with descriptive semantics

Franz Baader

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

Abstract. Computing the least common subsumer (lcs) is one of the most prominent non-standard inference in description logics. Baader, Küsters, and Molitor have shown that the lcs of concept descriptions in the description logic $\mathcal{EL}$ always exists and can be computed in polynomial time. In the present paper, we try to extend this result from concept descriptions to concepts defined in a (possibly cyclic) $\mathcal{EL}$-terminology interpreted with descriptive semantics, which is the usual first-order semantics for description logics. In this setting, the lcs need not exist. However, we are able to define possible candidates P_k $(k \geq 0)$ for the lcs, and can show that the lcs exists iff one of these candidates is the lcs. Since each of these candidates is a common subsumer, they can also be used to approximate the lcs even if it does not exist. In addition, we give a sufficient condition for the lcs to exist, and show that, under this condition, it can be computed in polynomial time.

1 Introduction

Computing 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 describe the concept to be defined by examples, which are themselves given as concepts.[1] These examples are then generalized into a new concept by computing their least common subsumer (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. Another application of the least common subsumer computation is structuring of DL knowledge bases. In fact, in many cases these knowledge bases are rather "flat" in the sense that their subsumption hierarchy is not deep and that a given concept may have a huge number of direct descendants in this hierarchy. To support browsing such hierarchies, one would like to introduce meaningful intermediate concepts, and this can again be facilitated by computing

[1] If the examples are not given as concepts, but as individuals in a DL ABox, then one must first generalize the individuals into concepts by computing their most specific concept [4, 8].

the lcs of subsets of the direct descendants of concepts with many descendants. These applications (and how formal concept analysis can be employed in this context) are described in more detail in [3].

The least common subsumer (lcs) in DLs with existential restrictions was investigated in [5]. In particular, it was shown there that the lcs in the small DL $\mathcal{EL}$ (which allows conjunctions, existential restrictions, and the top-concept) always exists, and that the binary lcs can be computed in polynomial time. In the present paper, we try to extend this result from concept descriptions to concepts defined in a (possibly cyclic) $\mathcal{EL}$-terminology interpreted with descriptive semantics, which is the usual first-order semantics for description logics.

The report [2] considers cyclic terminologies in $\mathcal{EL}$ w.r.t. the three types of semantics (greatest fixpoint (gfp), least fixpoint (lfp), and descriptive semantics) introduced by Nebel [9], and shows that the subsumption problem can be decided in polynomial time in all three cases. This is in strong contrast to the case of DLs with value restrictions. Even for the small DL $\mathcal{FL}_0$ (which allows conjunctions and value restrictions only), adding cyclic terminologies increases the complexity of the subsumption problem from polynomial (for concept descriptions) to PSPACE. The main tool in the investigation of cyclic definitions in $\mathcal{EL}$ is a characterization of subsumption through the existence of so-called simulation relations, which can be computed in polynomial time [7].

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 can be used to characterize the lcs via the product of this graph with itself [1]. 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 [5].)

In the present paper, we concentrate on the lcs w.r.t. cyclic terminologies in $\mathcal{EL}$ with descriptive semantics. Here things are not as rosy as for gfp-semantics. We will show that, in general, the lcs need not exist (Section 4.1). We then introduce possible candidates P_k ($k \geq 0$) for the lcs, and show that the lcs exists iff one of these candidates is the lcs (Section 4.2). Finally, we give a sufficient condition for the lcs to exist, and show that, under this condition, it can be computed in polynomial time (Section 4.3).

Before we can start presenting the new results, we must first introduce $\mathcal{EL}$ with cyclic terminologies as well as the lcs (Section 2), and recall the important definitions and results from [2] (Section 3). Full proofs for the results presented in this paper can be found in [1].

2 Cyclic TBoxes and least common subsumers 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

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	$A \equiv D$	$A^\mathcal{I} = D^\mathcal{I}$

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

($\exists r.C$). 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 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 $\mathcal{T}$.

The semantics of (possibly cyclic) $\mathcal{EL}$-TBoxes we have just defined is called *descriptive semantic* by Nebel [9]. For some applications, it is more appropriate to interpret cyclic concept definitions with the help of an appropriate fixpoint semantics. However, in this paper we will restrict our attention to descriptive semantic (see [1] for a treatment of subsumption in $\mathcal{EL}$ w.r.t. greatest fixpoint, least fixpoint, and descriptive semantics, and [2] for a treatment of the lcs w.r.t. greatest fixpoint semantics).

Definition 1. *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and let A, B be defined concepts occurring in $\mathcal{T}$. Then, A is subsumed by B w.r.t. descriptive semantics ($A \sqsubseteq_\mathcal{T} B$) iff $A^\mathcal{I} \subseteq B^\mathcal{I}$ holds for all models $\mathcal{I}$ of $\mathcal{T}$.*

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

To avoid this problem, we allow the original TBox to be extended by new definitions when constructing the lcs. 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 $\mathcal{T}_1$.

Lemma 1. *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 2. *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox containing the defined concepts A, B, and let $\mathcal{T}_2$ be a conservative extension of $\mathcal{T}_1$ containing the new defined concept E. Then E in $\mathcal{T}_2$ is a* least common subsumer *of A, B in $\mathcal{T}_1$* w.r.t. descriptive semantics *(lcs) iff the following two conditions are satisfied:*

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

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

Proposition 1. *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox containing the defined concepts A, B. Assume that $\mathcal{T}_2$ and $\mathcal{T}_2'$ are conservative extensions of $\mathcal{T}_1$ such that*

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

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

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 [2]. 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 [2], one can (without loss of generality) restrict the attention to normalized TBox. In the following, we thus assume that all TBoxes are normalized. Normalized $\mathcal{EL}$-TBoxes can be viewed as graphs whose nodes are the defined concepts, which are labeled by sets of primitive concepts, and whose edges are given by the existential restrictions. For the rest of this section, we fix a normalized $\mathcal{EL}$-TBox $\mathcal{T}$ with primitive concepts N_{prim}, defined concepts N_{def}, and roles N_{role}.

Definition 3. *An $\mathcal{EL}$-*description graph *is a graph $\mathcal{G} = (V, E, L)$ where*

- *V is a set of nodes;*
- *$E \subseteq V \times N_{role} \times V$ is a set of edges labeled by role names;*
- *$L\colon V \rightarrow 2^{N_{prim}}$ 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 4. *Let $\mathcal{G}_i = (V_i, E_i, L_i)$ $(i = 1, 2)$ be two $\mathcal{EL}$-description graphs. The binary relation $Z \subseteq V_1 \times V_2$ is a* simulation *from $\mathcal{G}_1$ to $\mathcal{G}_2$ iff*

(S1) *$(v_1, v_2) \in Z$ implies $L_1(v_1) \subseteq L_2(v_2)$; and*
(S2) *if $(v_1, v_2) \in Z$ and $(v_1, r, v_1') \in E_1$, then there exists a node $v_2' \in V_2$ such that $(v_1', v_2') \in Z$ and $(v_2, r, v_2') \in E_2$.*

We write $Z\colon \mathcal{G}_1 \stackrel{\rightharpoonup}{\sim} \mathcal{G}_2$ 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 [7]. As an easy consequence of this fact, the following proposition is proved in [2].

Proposition 2. *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 $Z\colon \mathcal{G}_1 \stackrel{\rightharpoonup}{\sim} \mathcal{G}_2$ such that $(v_1, v_2) \in Z$.*

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

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

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

Fig. 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 $Z\colon \mathcal{G}_\mathcal{T} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{T}$ such that $(B, A) \in Z$ (see [2]). 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 5. *The path* $p_1\colon B = B_0 \xrightarrow{r_1} B_1 \xrightarrow{r_2} B_2 \xrightarrow{r_3} B_3 \xrightarrow{r_4} \cdots$ *in* $\mathcal{G}_\mathcal{T}$ *is* Z-simulated *by the path* $p_2\colon A = A_0 \xrightarrow{r_1} A_1 \xrightarrow{r_2} A_2 \xrightarrow{r_3} A_3 \xrightarrow{r_4} \cdots$ *in* $\mathcal{G}_\mathcal{T}$ *iff* $(B_i, A_i) \in Z$ *for all* $i \geq 0$. *In this case we say that the pair* (p_1, p_2) *is a* (B, A)-simulation chain w.r.t. Z *(see Figure 1).*

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

Definition 6. *A* partial (B, A)-simulation chain *is of the form depicted in Figure 2. A* selection function S *for* A, B *and* Z *assigns to each partial* (B, A)-*simulation chain of this form a defined concept* A_n *such that* (A_{n-1}, r_n, A_n) *is an edge in* $\mathcal{G}_\mathcal{T}$ *and* $(B_n, A_n) \in Z$.

Given a path $B = B_0 \xrightarrow{r_1} B_1 \xrightarrow{r_2} B_2 \xrightarrow{r_3} B_3 \xrightarrow{r_4} \cdots$ and a defined concept A such that $(B, A) \in Z$, one can use a selection function S for A, B and Z to construct a Z-simulating path. In this case we say that the resulting (B, A)-simulation chain is *S-selected*.

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

We are now ready to state the characterization of subsumption w.r.t. descriptive semantics proved in [2].

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

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

In [2] it is also shown that, for a given $\mathcal{EL}$-TBox $\mathcal{T}$ and defined concepts A, B in $\mathcal{T}$, the existence of a (B, A)-synchronized simulation $Z\colon \mathcal{G}_{\mathcal{T}} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{T}}$ with $(B, A) \in Z$ can be decided in polynomial time.

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

4 The lcs w.r.t. descriptive semantics

Deriving a characterization of the lcs (w.r.t. descriptive semantics) from Theorem 1 is not straightforward. First, we will show that, w.r.t. descriptive semantics, the lcs of two concepts defined in an $\mathcal{EL}$-TBox need not exist. Subsequently, we will introduce possible "candidates" P_k ($k \geq 0$) for the lcs, and show that the lcs exists iff one of these candidates is the lcs. Finally, we will give a sufficient condition for the existence of the lcs.

4.1 The lcs need not exist

Theorem 2. *Let $\mathcal{T}_1 := \{A \equiv \exists r.A,\ \ B \equiv \exists r.B\}$. Then, A, B in $\mathcal{T}_1$ do not have an lcs.*

Proof. Assume to the contrary that $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$ and that the defined concept E in $\mathcal{T}_2$ is an lcs of A, B in $\mathcal{T}_1$. Let $\mathcal{G}_2 = (V_2, E_2, L_2)$ be the description graph induced by $\mathcal{T}_2$.

First, we show that there cannot be an infinite path in $\mathcal{G}_2$ starting with E. In fact, assume that

$$E = E_0 \xrightarrow{r_1} E_1 \xrightarrow{r_2} E_2 \xrightarrow{r_3} \cdots$$

is such an infinite path. Since $A \sqsubseteq_{\mathcal{T}_1} E$, there is an (E, A)-synchronized simulation $Z_1\colon \mathcal{G}_2 \overset{\rightharpoonup}{\sim} \mathcal{G}_2$ such that $(E, A) \in Z_1$. Consequently, the corresponding selection function S_1 can be used to turn the above infinite chain issuing from E into an (E, A)-simulation chain. Since the only edge with source A is the edge (A, r, A), this simulation chain is actually of the form

$$\begin{array}{ccccccccc}
E = E_0 & \xrightarrow{r} & E_1 & \xrightarrow{r} & E_2 & \xrightarrow{r} & E_3 & \xrightarrow{r} & \cdots \\
Z_1\!\downarrow & & Z_1\!\downarrow & & Z_1\!\downarrow & & Z_1\!\downarrow & & \\
A & \xrightarrow{r} & A & \xrightarrow{r} & A & \xrightarrow{r} & A & \xrightarrow{r} & \cdots
\end{array}$$

Since Z_1 is (E, A)-synchronized with selection function S_1, this implies that there is an index j_1 such that $E_{j_1} = A$, and thus $E_i = A$ for all $i \geq j_1$.

Analogously, we can show that there is an index j_2 such that $E_{j_2} = B$, and thus $E_i = B$ for all $i \geq j_2$. Since $A \neq B$, this is a contradiction. Thus, we know that there is a positive integer n_0 such that every path in $\mathcal{G}_2$ starting with E has length $\leq n_0$.

Second, we define conservative extensions $\mathcal{T}'_n$ ($n \geq 1$) of $\mathcal{T}_2$ such that the defined concept F_n in $\mathcal{T}'_n$ is a common subsumer of A, B:

$$\mathcal{T}'_n := \mathcal{T}_2 \cup \{F_n \equiv \exists r.F_{n-1}, ..., F_1 \equiv \exists r.F_0, F_0 \equiv \top\}.$$

It is easy to see that $A \sqsubseteq_{\mathcal{T}'_n} F_n$ and $B \sqsubseteq_{\mathcal{T}'_n} F_n$.

Third, we claim that, for $n > n_0$, $E \not\sqsubseteq_{\mathcal{T}'_n} F_n$. In fact, the path

$$F_n \xrightarrow{r} F_{n-1} \xrightarrow{r} F_{n-2} \xrightarrow{r} \cdots \xrightarrow{r} F_0$$

has length n, and thus it cannot be simulated by any path starting with E. This shows that $E \not\sqsubseteq_{\mathcal{T}'_n} F_n$, and thus contradicts our assumption that E in $\mathcal{T}_2$ is the lcs of A, B in $\mathcal{T}_1$. $\square$

4.2 Characterizing when the lcs exists

Given an $\mathcal{EL}$-TBox $\mathcal{T}_1$ and defined concepts A, B in $\mathcal{T}_1$, we will define for each $k \geq 0$ a conservative extension $\mathcal{T}_2^{(k)}$ of $\mathcal{T}_1$ containing a defined concept P_k, and show that A, B have an lcs iff there is a k such that P_k is the lcs of A, B. To prove this result, we will need a slight modification of Theorem 1. However, this modified theorem follows easily from the the proof of Theorem 1 given in [2].

Definition 8. *(i) We call a selection function S* nice *iff it satisfies the following two conditions:*

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

(ii) The simulation relation Z is called strongly (B, A)-synchronized *iff there exists a* nice *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 $A_i = B_i$.*

Corollary 2. *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, and A, B be defined concepts in $\mathcal{T}$. Then the following are equivalent:*

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

Strongly (B, A)-synchronized simulations satisfy the following property:

Lemma 2. *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox containing at most n defined concepts, A, B be defined concepts in $\mathcal{T}$, and Z: $\mathcal{G}_\mathcal{T} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{T}$ be a strongly (B, A)-synchronized simulation relation. Consider an infinite S-selected (B, A)-simulation chain of the form depicted in Figure 1. Then there exists an $m < n^2$ such that $B_m = A_m$.*

Obviously, the lemma also holds for finite S-selected (B, A)-simulation chains, provided that they are long enough, i.e., of length at least n^2.

Now, let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox, let $\mathcal{G}_{\mathcal{T}_1} = (N_{def}, E_{\mathcal{T}_1}, L_{\mathcal{T}_1})$ be the corresponding description graph, and let A, B be defined concepts in $\mathcal{T}_1$ (i.e., elements of N_{def}). W.r.t. gfp-semantics, the node (A, B) in the product $\mathcal{G} := \mathcal{G}_{\mathcal{T}_1} \times \mathcal{G}_{\mathcal{T}_1}$ of $\mathcal{G}_{\mathcal{T}_1}$ with itself yields the lcs of A, B [1]. The nodes of $\mathcal{G}$ are pairs (u, v) of nodes of $\mathcal{G}_{\mathcal{T}_1}$; there is an edge $((u, v), r, (u', v'))$ in $\mathcal{G}$ iff $(u, r, u') \in E_{\mathcal{T}_1}$ and $(v, r, v') \in E_{\mathcal{T}_1}$; and the label of (u, v) in $\mathcal{G}$ is $L_{\mathcal{T}_1}(u) \cap L_{\mathcal{T}_1}(v)$.

W.r.t. descriptive semantics, the product graph $\mathcal{G}$ as a whole cannot be part of the lcs of A, B since it may contain cycles reachable from (A, B), which would prevent the subsumption relationship between A and (A, B) to hold. Nevertheless, the lcs must "contain" paths in $\mathcal{G}$ starting with (A, B) up to a certain length k. In order to obtain these paths without also getting the cycles in $\mathcal{G}$, we make copies of the nodes in $\mathcal{G}$ on levels between 1 and k. Actually, we will not need nodes of the form (u, u) since they are represented by the nodes u in $\mathcal{G}_{\mathcal{T}_1}$.

To be more precise, we define

$$\mathcal{P}_k := \{(A, B)^0\} \cup \{(u, v)^n \mid u \neq v, (u, v) \in N_{def} \times N_{def} \text{ and } 1 \leq n \leq k\}.$$

For $p = (u, v)^n \in \mathcal{P}_k$ we call (u, v) the node of p and n the level of p.

The edges of $\mathcal{G}$ induce edges between elements of $\mathcal{P}_k$. To be more precise, we define the set of edges $E_{\mathcal{P}_k}$ as follows: $(p, r, q) \in E_{\mathcal{P}_k}$ iff the following conditions are satisfied:

- $p, q \in \mathcal{P}_k$;
- $p = (u, v)^n$ for some $n, 0 \leq n \leq k$;
- $q = (u', v')^{n+1}$;
- $(u, r, u') \in E_{\mathcal{T}_1}$ and $(v, r, v') \in E_{\mathcal{T}_1}$;

Note that the graph $(\mathcal{P}_k, E_{\mathcal{P}_k})$ is a directed acyclic graph. The only element on level 0 is $(A, B)^0$.

The label of an element of $\mathcal{P}_k$ is the label of its node in the product graph $\mathcal{G}$, i.e., if $p = (u, v)^n \in \mathcal{P}_k$, then

$$L_{\mathcal{P}_k}(p) = L_{\mathcal{T}_1}(u) \cap L_{\mathcal{T}_1}(v).$$

We are now ready to define an $\mathcal{EL}$-description graph $\mathcal{G}_2^{(k)}$ whose corresponding TBox $\mathcal{T}_2^{(k)}$ is a conservative extension of $\mathcal{T}_1$, and which contains a defined concept P_k that is a common subsumer of A, B.

Definition 9. *For all $k \geq 0$, we define $\mathcal{G}_2^{(k)} := (V_2^{(k)}, E_2^{(k)}, L_2^{(k)})$ where*

- $V_2^{(k)} := N_{def} \cup \mathcal{P}_k$;

– $L_2^{(k)} = L_{\mathcal{T}_1} \cup L_{\mathcal{P}_k}$, *i.e.*

$$L_2^{(k)}(v) := \begin{cases} L_{\mathcal{T}_1}(v) & \textit{if } v \in N_{def} \\ L_{\mathcal{P}_k}(v) & \textit{if } v \in \mathcal{P}_k \end{cases}$$

– $E_2^{(k)}$ *consists of the edges in* $E_{\mathcal{T}_1}$ *and* $E_{\mathcal{P}_k}$, *extended by some additional edges from* $\mathcal{P}_k$ *to* N_{def}*:*

$$E_2^{(k)} := E_{\mathcal{T}_1} \cup E_{\mathcal{P}} \cup \{(p,r,w) \mid p = (u,v)^n \in \mathcal{P}_k,\ w \in N_{def},\ \textit{and}\ (u,r,w) \in E_{\mathcal{T}_1}\ \textit{and}\ (v,r,w) \in E_{\mathcal{T}_1}\}.$$

Let $\mathcal{T}_2^{(k)}$ be the $\mathcal{EL}$-TBox such that $\mathcal{G}_2^{(k)} = \mathcal{G}_{\mathcal{T}_2^{(k)}}$. It is easy to see that $\mathcal{T}_2^{(k)}$ is a conservative extension of $\mathcal{T}_1$.

Lemma 3. $A \sqsubseteq_{\mathcal{T}_2^{(k)}} (A,B)^0$ *and* $B \sqsubseteq_{\mathcal{T}_2^{(k)}} (A,B)^0$.

Proof. To prove $A \sqsubseteq_{\mathcal{T}_2^{(k)}} (A,B)^0$, it is enough to show that there exists an $((A,B)^0, A)$-synchronized simulation $Z\colon \mathcal{G}_{\mathcal{T}_2^{(k)}} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{T}_2^{(k)}}$ such that $((A,B)^0, A) \in Z$. We define the relation Z as follows:

$$Z := \{(p,u) \mid p \in \mathcal{P}_k, u \in N_{def}, \text{ and the node of } p \text{ is of the form } (u,v)\} \cup \{(u,u) \mid u \in N_{def}\}.$$

In [1] it is shown that Z is indeed an $((A,B)^0, A)$-synchronized simulation such that $((A,B)^0, A) \in Z$. □

What we want to show next is that every common subsumer of A, B also subsumes $(A,B)^0$ in $\mathcal{T}_2^{(k)}$ 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 $A \sqsubseteq_{\mathcal{T}_2} F$ and $B \sqsubseteq_{\mathcal{T}_2} F$. For $\mathcal{G}_{\mathcal{T}_2} = (V_2, E_2, L_2)$, this implies that there is

– an (F,A)-synchronized simulation relation $Y_1\colon \mathcal{G}_{\mathcal{T}_2} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{T}_2}$ with selection function S_1 such that $(F,A) \in Y_1$, and
– an (F,B)-synchronized simulation relation $Y_2\colon \mathcal{G}_{\mathcal{T}_2} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{T}_2}$ with selection function S_2 such that $(F,B) \in Y_2$.

By Corollary 2 we may assume without loss of generality that the selection functions S_1, S_2 are nice. Consequently, if $k = |V_2|^2$, then Lemma 2 shows that the selection functions S_1, S_2 ensure synchronization after less than k steps.

In the following, let $k := |V_2|^2$. In order to have a subsumption relationship between $(A,B)^0$ in $\mathcal{T}_2^{(k)}$ 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 $\mathcal{P}_k$) lies on a cycle, and thus we can rename them without changing the meaning of these concepts. (Note that the characterization of subsumption given in Theorem 1 implies that only for defined concepts occurring on cycles their actual names are relevant.) Thus, $\mathcal{T}_3$ is a conservative extension of both $\mathcal{T}_2^{(k)}$ and $\mathcal{T}_2$.

Lemma 4. $(A,B)^0 \sqsubseteq_{\mathcal{T}_3} F$

Proof. We must show that there is an $(F,(A,B)^0)$-synchronized simulation relation $Y\colon \mathcal{G}_{\mathcal{T}_3} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{T}_3}$ such that $(F,(A,B)^0) \in Y$. The definition of this simulation is based on the "product" of Y_1 and Y_2:

$$\begin{aligned} Y := \{(u,p) \mid\ & (u,v_1) \in Y_1 \text{ and } (u,v_2) \in Y_2 \\ & \text{where } (v_1,v_2) \text{ is the node of } p \in \mathcal{P}_k \ \} \quad \cup \\ \{(u,v) \mid\ & v \in N_{def} \text{ and } (u,v) \in Y_1\}. \end{aligned}$$

In [1] it is shown that Y is indeed an $(F,(A,B)^0)$-synchronized simulation such that $(F,(A,B)^0) \in Y$. □

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 $\mathcal{P}_k$ we could add a lower index k. Thus, $(u,v)_k^n$ denotes a node on level n in $\mathcal{P}_k$. For $k \geq 0$, we denote $(A,B)_k^0$ by P_k.

Theorem 3. *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox and A,B defined concepts in $\mathcal{T}_1$. Then A,B in $\mathcal{T}_1$ have an lcs iff there is a k such that P_k in $\mathcal{T}_2^{(k)}$ is the lcs of A,B in $\mathcal{T}_1$.*

Proof. The direction from right to left is trivial. Thus, assume that $\mathcal{T}_2$ is a conservative extension of $\mathcal{T}_1$ and that P in $\mathcal{T}_2$ is the lcs of A,B. We define $k := n^2$ where n is the number of defined concepts in $\mathcal{T}_2$. Let $\mathcal{T}_3$ be the union of $\mathcal{T}_2$ and $\mathcal{T}_2^{(k)}$, where we assume without loss of generality that the only defined concepts shared by $\mathcal{T}_2$ and $\mathcal{T}_2^{(k)}$ are the ones in $\mathcal{T}_1$. Then Lemma 4 shows that $P_k \sqsubseteq_{\mathcal{T}_3} P$.

Since P_k is a common subsumer of A,B by Lemma 3, the fact that P is the least common subsumer of A,B implies that subsumption in the other direction holds as well: $P \sqsubseteq_{\mathcal{T}_3} P_k$. Thus, P and P_k are equivalent, and this implies that P_k is also an lcs of A,B. □

In [1] it is also shown that the concepts P_k form a decreasing chain w.r.t. subsumption, and that P_k is the lcs of A,B iff it is equivalent to P_{k+i} for all $i \geq 1$.

Example 1. Let us reconsider the TBox $\mathcal{T}_1$ defined in Theorem 2. In this case, the TBoxes $\mathcal{T}_2^{(k)}$ are basically of the form[2]

$$\mathcal{T}_1 \cup \{P_k \equiv \exists r.(A,B)_k^1,\ (A,B)_k^1 \equiv \exists r.(A,B)_k^2,\ \ldots,\ (A,B)_k^{k-1} \equiv \exists r.(A,B)_k^k\},$$

and it is easy to see that there always is a strict subsumption relationship between P_k and P_{k+1} (since P_{k+1} requires an r-chain of length $k+1$ whereas P_k only requires one of length k).

[2] We have restricted the attention to elements of $\mathcal{P}_k$ that are reachable from P_k.

The following is an example where the lcs exists.

Example 2. Let us consider the following TBox

$$\mathcal{T}_1 := \{A \equiv \exists r.A \sqcap \exists r.C,\ B \equiv \exists r.B \sqcap \exists r.C,\ C \equiv \exists r.C\}.$$

In this case, $k = 0$ does the job, and thus the lcs of A, B is P_0:

$$\mathcal{T}_2^{(0)} := \mathcal{T}_1 \cup \{P_0 \equiv \exists r.C\}.$$

In fact, it is easy to see that the path $P_0 \xrightarrow{r} C \xrightarrow{r} C \xrightarrow{r} \cdots$ can simulate any path starting with some P_ℓ for $\ell \geq 1$. Since the infinite paths starting with P_ℓ must eventually also lead to C (after at most ℓ steps), this really yields a synchronized simulation relation.

4.3 A sufficient condition for the existence of the lcs

If we want to use the results from the previous subsection to compute the lcs, we must be able to decide whether there is an index k such that P_k is the lcs of A, B, and if yes we must also be able to compute such a k. Though we strongly conjecture that this is possible, we have not yet found such a procedure. For this reason, we must restrict ourself to give a *sufficient condition* for the lcs of two concepts defined in an $\mathcal{EL}$-TBox to exist.

As before, let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox, let $\mathcal{G}_{\mathcal{T}_1} = (N_{def}, E_{\mathcal{T}_1}, L_{\mathcal{T}_1})$ be the corresponding description graph, and let A, B be defined concepts in $\mathcal{T}_1$ (i.e., elements of N_{def}). We consider the product $\mathcal{G} := \mathcal{G}_{\mathcal{T}_1} \times \mathcal{G}_{\mathcal{T}_1}$ of $\mathcal{G}_{\mathcal{T}_1}$ with itself. Let $\mathcal{G} = (V, E, L)$.

Definition 10. *We say that (A, B) is synchronized in $\mathcal{T}_1$ iff, for every infinite path $(A, B) = (u_0, v_0) \xrightarrow{r_1} (u_1, v_1) \xrightarrow{r_2} (u_2, v_2) \xrightarrow{r_3} \cdots$ in $\mathcal{G}$, there exists an index $i \geq 0$, such that $u_i = v_i$.*

For example, in the TBox $\mathcal{T}_1$ introduced in Theorem 2, (A, B) is *not* synchronized. The same is true for the TBox defined in Example 2. As another example, consider the TBox $\mathcal{T}_1' := \{A' \equiv \exists r_1.A' \sqcap \exists r.C,\ \ B' \equiv \exists r_2.B' \sqcap \exists r.C,\ \ C \equiv \exists r.C\}$. In this TBox, (A', B') is synchronized.

Lemma 5. *Assume that (A, B) is synchronized in $\mathcal{T}_1$, and let $k := |N_{def}|^2$. Then, for every path $(A, B) = (u_0, v_0) \xrightarrow{r_1} (u_1, v_1) \xrightarrow{r_2} (u_2, v_2) \xrightarrow{r_3} \cdots \xrightarrow{r_k} (u_k, v_k)$ in $\mathcal{G}$ of length k, there exists an index $i, 0 \leq i \leq k$ such that $u_i = v_i$.*

As an easy consequence of this lemma we obtain that $k = |N_{def}|^2$ is such that P_k is the lcs of A, B (see [1] for the proof). Thus, the lcs of A, B in $\mathcal{T}_1$ always exists, provided that (A, B) is synchronized in $\mathcal{T}_1$. Our construction of the TBox $\mathcal{T}_2^{(k)}$ is obviously polynomial in k and the size of $\mathcal{T}_1$. Since k is also polynomial in the size of $\mathcal{T}_1$, the size of $\mathcal{T}_2^{(k)}$ is polynomial in the size of $\mathcal{T}_1$.

Theorem 4. *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox, and let A, B be defined concepts in $\mathcal{T}_1$ such that (A, B) is synchronized in $\mathcal{T}_1$. Then the lcs of A, B in $\mathcal{T}_1$ always exists, and it can be computed in polynomial time.*

Example 2 shows that the lcs may exist even if (A, B) is not synchronized in $\mathcal{T}_1$. Thus, this is a sufficient, but not necessary condition for the existence of the lcs. We close this section by showing that this sufficient condition can be decided in polynomial time.

Proposition 3. *Let $\mathcal{T}_1$ be an $\mathcal{EL}$-TBox, and let A, B be defined concepts in $\mathcal{T}_1$. Then it can be decided in polynomial time whether (A, B) is synchronized in $\mathcal{T}_1$.*

Proof. As before, consider the product $\mathcal{G} := \mathcal{G}_{\mathcal{T}_1} \times \mathcal{G}_{\mathcal{T}_1}$ of $\mathcal{G}_{\mathcal{T}_1}$ with itself. Let $\mathcal{G} = (V, E, L)$. We define

$$\begin{aligned} W_0 &:= \{(u, u) \mid (u, u) \in V\}, \\ W_{i+1} &:= W_i \cup \{(u, v) \mid (u, v) \in V \text{ and all edges with source } (u, v) \text{ in } \mathcal{G} \\ &\qquad \text{lead to elements of } W_i\}, \quad \text{and} \\ W_\infty &:= \bigcup_{i \geq 0} W_i. \end{aligned}$$

Obviously, W_∞ can be computed in time polynomial in the size of $\mathcal{G}$. In [1] it is shown that (A, B) is synchronized in $\mathcal{T}_1$ iff $(A, B) \in W_\infty$. From this, the proposition immediately follows. □

5 Related and future work

Cyclic definitions in $\mathcal{EL}$ w.r.t. the three types of semantics introduced by Nebel [9] were investigated in [2]. (A short version of this paper is submitted for publication at another conference.) It was shown that the subsumption problem remains polynomial in all three cases. 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 on the graph associated with an $\mathcal{EL}$-terminology.

The characterization of subsumption in $\mathcal{EL}$ w.r.t. gfp-semantics was used in [1] to characterize the lcs w.r.t. gfp-semantics via the product of this graph with itself. This shows that, w.r.t. gfp semantics, the lcs always exists, and that the binary lcs can be computed in polynomial time. The characterization of subsumption w.r.t. gfp-semantics can also be extended to the instance problem in $\mathcal{EL}$. This was used in [1] to show that the most specific concept in $\mathcal{EL}$ with cyclic terminologies interpreted with gfp-semantics always exists, and can be computed in polynomial time. (These results on the lcs and msc in $\mathcal{EL}$ w.r.t. gfp-semantics are submitted for publication at another conference.)

Subsumption is also polynomial w.r.t. descriptive semantics [2]. For the lcs, descriptive semantics is not that well-behaved: the lcs need not exist in general. In addition, we could only give a sufficient condition for the existence of the lcs. If this condition applies, then the lcs can be computed in polynomial time. Thus, one of the main technical problems left open by the present paper is the question how to characterize the cases in which the lcs exists w.r.t. descriptive semantics, and to determine whether in these cases it can always be computed in polynomial time. Another problem that was not addressed by the present paper is the question of how to characterize and compute the most specific concept w.r.t. descriptive semantics.

It should be noted that there are indeed applications where the expressive power of the small DL $\mathcal{EL}$ appears to be sufficient. In fact, SNOMED, the Systematized Nomenclature of Medicine [6] uses $\mathcal{EL}$ [10, 11].

References

1. F. Baader. Least common subsumers, most specific concepts, and role-value-maps in a description logic with existential restrictions and terminological cycles. LTCS-Report 02-07, TU Dresden, Germany, 2002. See http://lat.inf.tu-dresden.de/research/reports.html. A short version will appear in *Proc. IJCAI 2003*.
2. F. Baader. Terminological cycles in a description logic with existential restrictions. LTCS-Report 02-02, Dresden University of Technology, Germany, 2002. See http://lat.inf.tu-dresden.de/research/reports.html. Some of the results in this report will also be published in *Proc. IJCAI 2003*.
3. F. Baader and R. Molitor. Building and structuring description logic knowledge bases using least common subsumers and concept analysis. In *Proc. ICCS 2000*, Springer LNAI 1867, 2000.
4. 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. KI'98*, Springer LNAI 1504, 1998.
5. Franz Baader, Ralf Küsters, and Ralf Molitor. Computing least common subsumers in description logics with existential restrictions. In *Proc. IJCAI'99*, 1999.
6. 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.
7. 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*, 1995.
8. Ralf Küsters and Ralf Molitor. Approximating most specific concepts in description logics with existential restrictions. In *Proc. KI 2001*, Springer LNAI 2174, 2001.
9. Bernhard Nebel. Terminological cycles: Semantics and computational properties. In John F. Sowa, editor, *Principles of Semantic Networks*. Morgan Kaufmann, 1991.
10. K.A. Spackman. Managing clinical terminology hierarchies using algorithmic calculation of subsumption: Experience with SNOMED-RT. *J. of the American Medical Informatics Association*, 2000. Fall Symposium Special Issue.
11. K.A. Spackman. Normal forms for description logic expressions of clinical concepts in SNOMED RT. *J. of the American Medical Informatics Association*, 2001. Symposium Supplement.