TECHNISCHE UNIVERSITÄT DRESDEN

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

LTCS–Report

Terminological cycles in a description logic with existential restrictions

Franz Baader

LTCS-Report 02-02

This is an updated version of the original report, in which some errors in Section 3.1 of the original report have been corrected.

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

Hans-Grundig-Str. 25
01062 Dresden
Germany

Terminological cycles in a description logic with existential restrictions

Franz Baader*
Theoretical Computer Science
Dresden University of Technology
D-01062 Dresden, Germany
e-mail: baader@inf.tu-dresden.de

Abstract

Cyclic definitions in description logics have until now been investigated only for description logics allowing for value restrictions. Even for the most basic language $\mathcal{FL}_0$, which allows for conjunction and value restrictions only, deciding subsumption in the presence of terminological cycles is a PSPACE-complete problem. This report investigates subsumption in the presence of terminological cycles for the language $\mathcal{EL}$, which allows for conjunction and existential restrictions. In contrast to the results for $\mathcal{FL}_0$, subsumption in $\mathcal{EL}$ remains polynomial, independent of whether we use least fixpoint semantics, greatest fixpoint semantics, or descriptive semantics. These results are shown via a characterization of subsumption through the existence of certain simulation relations between nodes of the description graph associated with a given cyclic terminology.

1 Introduction

The first thorough investigation of cyclic terminologies in description logics (DL) is due to Nebel [22], who introduced three different semantics for such terminologies: least fixpoint (lfp) semantics, which considers only the models that interpret the defined concepts as small as possible; greatest fixpoint (gfp) semantics, which considers only the models that interpret the defined concepts as large as possible; and descriptive semantics, which considers all models.

In [1, 2], subsumption w.r.t. cyclic terminologies in the small DL $\mathcal{FL}_0$, which allows for conjunction and value restrictions only, was characterized with the help

*Partially supported by the DFG under grant BA 1122/4-1.

of finite automata. This characterization provided PSPACE decision procedures for subsumption in $\mathcal{FL}_0$ with cyclic terminologies for the three types of semantics introduced by Nebel. In addition, it was shown in [1, 2] that subsumption is PSPACE-hard both for gfp- and lfp-semantics. For descriptive semantics, the exact complexity of the subsumption problem is still open. However, Nebel [21] showed that, even for acyclic terminologies (where the three types of semantics agree), subsumption in $\mathcal{FL}_0$ is at least coNP-hard. The results for cyclic $\mathcal{FL}_0$-terminologies where extended by Küsters [14] to $\mathcal{ALN}$, which extends $\mathcal{FL}_0$ by atomic negation and number restrictions. For all three types of semantics, subsumption w.r.t. cyclic $\mathcal{ALN}$-terminologies is PSPACE-complete.

Schild's observation [23] that the DL $\mathcal{ALC}$ (which extends $\mathcal{FL}_0$ by full negation) is a syntactic variant of the multi-modal logic K opened a way for treating cyclic terminologies and more general recursive definitions in more expressive languages like $\mathcal{ALC}$ and extensions thereof by a reduction to the modal mu-calculus [24, 7]. In this setting, one can use a mix of the three types of semantics introduced by Nebel. However, the complexity of the subsumption problem is EXPTIME-complete.

In spite of these very general results for cyclic definitions in expressive languages, there are still good reasons to look at cyclic terminologies in less expressive (in particular sub-Boolean) description logics. One reason is, of course, the lower complexity of the subsumption problem (for $\mathcal{FL}_0$ and $\mathcal{ALN}$ PSPACE rather than EXPTIME). In addition, the growing interest in non-standard inferences like computing the least common subsumer and the most specific concept [5, 6, 3, 4, 16, 15, 18, 17] has also led to a renewed interest in sub-Boolean description logics since some of these inferences (like the most common subsumer) make sense only if not all Boolean operators are present. In this context, cyclic definitions come into play since the most specific concept of a given ABox individual need not exit in languages allowing for number restrictions or existential restrictions. For $\mathcal{ALN}$ it was shown in [3] that the most specific concept always exists if one allows for cyclic concept definitions with gfp-semantics. For languages with existential restrictions, another solution to the non-existence of most specific concepts was proposed by Küsters and Molitor [17]. They considered the languages $\mathcal{EL}$ (which allows for conjunction and existential restrictions) and $\mathcal{ALE}$ (which additionally allows for atomic negation and value restrictions) and showed how the most specific concept can be approximated there. One reason for choosing an approximation approach rather than an exact characterization of the most specific concept using cyclic definitions was that the impact of cyclic definitions in description logics with existential restrictions was largely unexplored.

This report tries to overcome this deficit. It considers cyclic terminologies in $\mathcal{EL}$ w.r.t. the three types of semantics introduced by Nebel, 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 $\mathcal{FL}_0$, where adding cyclic terminologies increases the complexity of subsumption from polynomial (for concept descriptions) to PSPACE. The main tool used to show these results is a characterization of subsumption through the existence of so-called simulation relations. There is an interesting connection between this characterization and the characterization of subsumption between $\mathcal{EL}$-concept descriptions given in [4]. There it was shown that subsumption corresponds to the existence of a homomorphism between the description trees (basically the syntax trees) of the descriptions. This showed that subsumption between $\mathcal{EL}$-concept descriptions is decidable in polynomial time since the existence of a homomorphism between *trees* is a polynomial time problem. Intuitively, if one goes from concept descriptions to cyclic terminologies, then one obtains a description *graph* rather than a tree. Thus, an obvious conjecture would be that subsumption in $\mathcal{EL}$ with cyclic terminologies can be characterized through the existence of a homomorphism between the corresponding description graphs. Fortunately, this conjecture is not true. In fact, the existence of a homomorphism between graphs is an NP-complete problem [9] whereas the existence of a simulation is a polynomial time problem [12]. It is only for trees that the existence of a simulation implies the existence of a homomorphism. Thus, the characterization of subsumption through the existence of a simulation appears to be the deeper reason why subsumption of $\mathcal{EL}$-concept descriptions is polynomial.

In the next section we will introduce the DL $\mathcal{EL}$ as well as cyclic terminologies and the three types of semantics for these terminologies. Then we will show in Section 3 how such terminologies can be translated into description graphs. In this section, we will also define the notion of a simulation between nodes of a description graph, and prove some useful properties of simulations. The next three sections are then devoted to the characterization of subsumption in $\mathcal{EL}$ w.r.t. gfp, lfp, and descriptive semantics, respectively.

2 Cyclic terminologies in the DL $\mathcal{EL}$

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.

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.

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 [22]. For some applications, it is more appropriate to interpret cyclic concept definitions with the help of an appropriate fixpoint semantics. Before defining least and greatest fixpoint semantics formally, let us illustrate their effect on an example.

Example 1 Assume that our interpretations are graphs where we have nodes (elements of the concept name Node) and edges (represented by the role edge), and we want to define the concept Inode of all nodes lying on an infinite (possibly cyclic) path of the graph. The following is a possible definition of Inode:

$$\mathsf{Inode} \equiv \mathsf{Node} \sqcap \exists \mathsf{edge}.\mathsf{Inode}.$$

Now consider the following interpretation of the primitive concepts and roles:

$$\begin{aligned}\Delta^{\mathcal{J}} &:= \{m_0, m_1, m_2, \ldots\} \cup \{n_0\},\\ \mathsf{Node}^{\mathcal{J}} &:= \Delta^{\mathcal{I}},\\ \mathsf{edge}^{\mathcal{J}} &:= \{(m_i, m_{i+1}) \mid i \geq 0\} \cup \{(n_0, n_0)\}.\end{aligned}$$

Where $M := \{m_0, m_1, m_2, \ldots\}$ and $N := \{n_0\}$, there are four possible ways of extending this interpretation of the primitive concepts and roles to a model of the TBox consisting of the above concept definition: Inode can be interpreted by $M \cup N$, M, N, or $\emptyset$. All these models are admissible w.r.t. descriptive semantics,

whereas the first is the gfp-model and the last is the lfp-model of the TBox. Obviously, only the gfp-model captures the intuition underlying the definition (namely, nodes lying on an infinite path) correctly.
It should be noted, however, that in other cases descriptive semantics appears to be more appropriate. For example, consider the definitions

$$\mathsf{Tiger} \equiv \mathsf{Animal} \sqcap \exists \mathsf{parent}.\mathsf{Tiger} \quad \text{and} \quad \mathsf{Lion} \equiv \mathsf{Animal} \sqcap \exists \mathsf{parent}.\mathsf{Lion}.$$

With respect to gfp-semantics, the defined concepts Tiger and Lion must always be interpreted as the same set whereas this is not the case for descriptive semantics.[1]

Before we can define lfp- and gfp-semantics formally, we must introduce some notation. Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox containing the roles N_{role}, the primitive concepts N_{prim}, and the defined concepts $N_{def} := \{A_1, \ldots, A_k\}$. A *primitive interpretations* $\mathcal{J}$ for $\mathcal{T}$ is given by a domain $\Delta^{\mathcal{J}}$, an interpretation of the roles $r \in N_{role}$ by binary relations $r^{\mathcal{J}}$ on $\Delta^{\mathcal{J}}$, and an interpretation of the primitive concepts in $P \in N_{prim}$ by subsets $P^{\mathcal{J}}$ of $\Delta^{\mathcal{J}}$. Obviously, a primitive interpretation differs from an interpretation in that it does not interpret the defined concepts in N_{def}. We say that the interpretation $\mathcal{I}$ 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} and N_{prim}. 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 names 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 names: if $\mathcal{I}_1, \mathcal{I}_2 \in Int(\mathcal{J})$, then

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

It is easy to see that $\preceq_{\mathcal{J}}$ is 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* [25, 19] applies to all monotonic functions from $Int(\mathcal{J})$ to $Int(\mathcal{J})$. This theorem states the following: if $O\colon Int(\mathcal{J}) \to Int(\mathcal{J})$ is a function such that $\mathcal{I}_1 \preceq_{\mathcal{J}} \mathcal{I}_2$ implies $O(\mathcal{I}_1) \preceq_{\mathcal{J}} O(\mathcal{I}_2)$ (monotonicity), then O has a *fixpoint*, i.e., there is an $\mathcal{I}$ in $Int(\mathcal{J})$ such that $O(\mathcal{I}) = \mathcal{I}$. To be more precise, O has also a least fixpoint (i.e., a fixpoint smaller w.r.t. $\preceq_{\mathcal{J}}$ than all other fixpoints) and a greatest fixpoint (i.e., a fixpoint larger w.r.t. $\preceq_{\mathcal{J}}$ than all other fixpoints).

Definition 2 The 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 $i, 1 \leq i \leq k$.

[1] This example is similar to the "humans and horses" example used by Nebel [22] to illustrate the difference between descriptive semantics and gfp-semantics in $\mathcal{ALN}$.

Monotonicity of this function is an immediate consequence of the following lemma, which can be proved by induction on the structure of $\mathcal{EL}$-concept descriptions.

Lemma 3 *Let D be an $\mathcal{EL}$-concept description and $\mathcal{I}_1, \mathcal{I}_2$ interpretations based on the primitive interpretation $\mathcal{J}$. Then $\mathcal{I}_1 \preceq_{\mathcal{J}} \mathcal{I}_2$ implies $D^{\mathcal{I}_1} \subseteq D^{\mathcal{I}_2}$.*

Consequently, $O_{\mathcal{T},\mathcal{J}}$ has both a least and a greatest fixpoint, and possibly other fixpoints in-between (see Example 1). The following proposition is an immediate consequence of the definition of $O_{\mathcal{T},\mathcal{J}}$.

Proposition 4 *Let $\mathcal{I}$ be an interpretation based on the primitive interpretation $\mathcal{J}$. Then $\mathcal{I}$ is a fixpoint of $O_{\mathcal{T},\mathcal{J}}$ iff $\mathcal{I}$ is a model of $\mathcal{T}$.*

This shows that any primitive interpretation $\mathcal{J}$ can be extended to a model of $\mathcal{T}$. In particular, there is always a greatest and a least model of $\mathcal{T}$ extending $\mathcal{J}$.

Definition 5 Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox. The model $\mathcal{I}$ of $\mathcal{T}$ is called *gfp-model (lfp-model)* of $\mathcal{T}$ iff there is a primitive interpretation $\mathcal{J}$ such that $\mathcal{I} \in \mathit{Int}(\mathcal{J})$ is the greatest (least) fixpoint of $O_{\mathcal{T},\mathcal{J}}$. *Greatest (least) fixpoint semantics* considers only gfp-models (lfp-models) as admissible models.

We are now ready to define subsumption w.r.t. the three different types of semantics introduced above.

Definition 6 Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and A, B be defined names[2] 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 $\mathcal{T}$.
- A is subsumed by B w.r.t. gfp-semantics ($A \sqsubseteq_{gfp,\mathcal{T}} B$) iff $A^{\mathcal{I}} \subseteq B^{\mathcal{I}}$ holds for all gfp-models $\mathcal{I}$ of $\mathcal{T}$.
- A is subsumed by B w.r.t. lfp-semantics ($A \sqsubseteq_{lfp,\mathcal{T}} B$) iff $A^{\mathcal{I}} \subseteq B^{\mathcal{I}}$ holds for all lfp-models $\mathcal{I}$ of $\mathcal{T}$.

The main goal of this report is to show that all three subsumption problems are decidable in polynomial time. To be able to do that, we need some more information on how least and greatest fixpoints can be constructed. If the function

[2]Obviously, we can restrict the attention to subsumption between defined concepts since subsumption between arbitrary concept descriptions can be reduced to this problem by introducing definitions for the descriptions.

is not only monotonic, but also downward (upward) continuous, then the greatest (least) fixpoint can be constructed by a simple ω-iteration. Otherwise, we can still get the fixpoints through an iteration process, but this process may need larger ordinals than ω (see [19, 2] for a more detailed description).

Given an increasing chain $\mathcal{I}_0 \preceq_{\mathcal{J}} \mathcal{I}_1 \preceq_{\mathcal{J}} \mathcal{I}_2 \preceq_{\mathcal{J}} \ldots$ of interpretations based on $\mathcal{J}$, its least upper bound (lub) is the interpretation $\mathcal{I}$ based on $\mathcal{J}$ such $A_i^{\mathcal{I}} = \bigcup_{j\geq 0} A_i^{\mathcal{I}_j}$ holds for all $i, 1 \leq i \leq k$. The function O: $\mathit{Int}(\mathcal{J}) \to \mathit{Int}(\mathcal{J})$ is *upward continuous* iff

$$O(\mathit{lub}(\{\mathcal{I}_j \mid j \geq 0\})) = \mathit{lub}(\{O(\mathcal{I}_j) \mid j \geq 0\}).$$

Accordingly, the greatest lower bound (glb) of the decreasing chain $\mathcal{I}_0 \succeq_{\mathcal{J}} \mathcal{I}_1 \succeq_{\mathcal{J}} \mathcal{I}_2 \succeq_{\mathcal{J}} \ldots$ is the interpretation $\mathcal{I}$ based on $\mathcal{J}$ such $A_i^{\mathcal{I}} = \bigcap_{j\geq 0} A_i^{\mathcal{I}_j}$ holds for all $i, 1 \leq i \leq k$. The function O: $\mathit{Int}(\mathcal{J}) \to \mathit{Int}(\mathcal{J})$ is *downward continuous* iff

$$O(\mathit{glb}(\{\mathcal{I}_j \mid j \geq 0\})) = \mathit{glb}(\{O(\mathcal{I}_j) \mid j \geq 0\}).$$

Proposition 7 *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and $\mathcal{J}$ a primitive interpretation. Then $O_{\mathcal{T},\mathcal{J}}$ is upward continuous, but not necessarily downward continuous.*

Proof. (1) Let $\mathcal{I}_0 \preceq_{\mathcal{J}} \mathcal{I}_1 \preceq_{\mathcal{J}} \mathcal{I}_2 \preceq_{\mathcal{J}} \ldots$ be an increasing chain in $\mathit{Int}(\mathcal{J})$, and let $\mathcal{I}$ be its least upper bound. Upward continuity of $O_{\mathcal{T},\mathcal{J}}$ is an immediate consequence of the fact that

$$D^{\mathcal{I}} = \bigcup_{j\geq 0} D^{\mathcal{I}_j}$$

holds for all $\mathcal{EL}$-concept descriptions D. This can in turn easily be shown by induction on the structure of $\mathcal{EL}$-concept descriptions.

(2) Consider the TBox $\mathcal{T} := \{A \equiv \exists r.A\}$, and the primitive interpretation $\mathcal{J}$ with

- $\Delta^{\mathcal{J}} := \{a_0\} \cup \{a_{i,j} \mid 1 \leq j \leq i\}$;
- $r^{\mathcal{J}} := \{(a_0, a_{i,1}) \mid i \geq 1\} \cup \{(a_{i,j}, a_{i,j+1}) \mid 1 \leq j < i\}$.

If we consider $r^{\mathcal{J}}$ as the edges of a graph with nodes $\Delta^{\mathcal{J}}$, then this graph is a tree with root a_0, which is infinitely branching. All other nodes have at most one successor node. The root is the origin of infinitely many paths, one of length 1, one of length 2, etc. The interpretations $\mathcal{I}_\nu$ ($\nu \geq 0$) based on $\mathcal{J}$ are now defined as follows:

- $A^{\mathcal{I}_\nu} := \{a_0\} \cup \{a_{i,j} \mid i - j \geq \nu\}$.

It is easy to see that $\mathcal{I}_0 \succeq_{\mathcal{J}} \mathcal{I}_1 \succeq_{\mathcal{J}} \mathcal{I}_2 \succeq_{\mathcal{J}} \dots$ and that $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_\nu) = \mathcal{I}_{\nu+1}$. Consequently, $\mathcal{I} := glb(\{O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_\nu) \mid \nu \geq 0\}) = glb(\{\mathcal{I}_\nu \mid \nu \geq 1\})$. In particular, since $a_0 \in A^{\mathcal{I}_\nu}$ for all $\nu \geq 1$, this implies $a_0 \in A^{\mathcal{I}}$.

On the other hand, let $\mathcal{I}' := glb(\{\mathcal{I}_\nu \mid \nu \geq 0\})$ and $\mathcal{I}'' := O_{\mathcal{T},\mathcal{J}}(\mathcal{I}')$. Then $A^{\mathcal{I}'} = \{a_0\}$, and thus $A^{\mathcal{I}''} = \emptyset \neq A^{\mathcal{I}}$. □

The least and the greatest fixpoint of $O_{\mathcal{T},\mathcal{J}}$ can be obtained by iterated application of $O_{\mathcal{T},\mathcal{J}}$, respectively starting with the least and the greatest interpretation based on $\mathcal{J}$.

Definition 8 Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{J}$ a primitive interpretation, and $\mathcal{I}_{top}$ the greatest and $\mathcal{I}_{bot}$ the least interpretation based on $\mathcal{J}$, i.e., $A_i^{\mathcal{I}_{top}} = \Delta^{\mathcal{J}}$ and $A_i^{\mathcal{I}_{bot}} = \emptyset$ for all $i, 1 \leq i \leq k$. Then we define for every ordinal α:

- $\mathcal{I}^{\uparrow\alpha} := \mathcal{I}_{bot}$ and $\mathcal{I}^{\downarrow\alpha} := \mathcal{I}_{top}$ if $\alpha = 0$;
- $\mathcal{I}^{\uparrow\alpha+1} := O_{\mathcal{T},\mathcal{J}}(\mathcal{I}^{\uparrow\alpha})$ and $\mathcal{I}^{\downarrow\alpha+1} := O_{\mathcal{T},\mathcal{J}}(\mathcal{I}^{\downarrow\alpha})$;
- $\mathcal{I}^{\uparrow\alpha} := lub(\{\mathcal{I}^{\uparrow\beta} \mid \beta < \alpha\}$ and $\mathcal{I}^{\downarrow\alpha} := glb(\{\mathcal{I}^{\downarrow\beta} \mid \beta < \alpha\}$
  if α is a limit ordinal.

As usual, let ω denote the first infinite ordinal (i.e., the order type of the non-negative integers). Since $O_{\mathcal{T},\mathcal{J}}$ is upward continuous, Tarski's fixpoint theorem says that $\mathcal{I}^{\uparrow\omega}$ is the least fixpoint of $O_{\mathcal{T},\mathcal{J}}$. Since $O_{\mathcal{T},\mathcal{J}}$ need not be downward continuous, $\mathcal{I}^{\downarrow\omega}$ need not be a fixpoint of $O_{\mathcal{T},\mathcal{J}}$. However, Tarski's fixpoint theorem says that there exists an ordinal α such that $\mathcal{I}^{\downarrow\alpha}$ is the greatest fixpoint of $O_{\mathcal{T},\mathcal{J}}$.

In Section 6 we will also consider models of $\mathcal{T}$ that are the greatest models below a given interpretation $\mathcal{I}_0$.

Definition 9 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 describes a more general sufficient condition for the greatest $\mathcal{I}_0$-model of $\mathcal{T}$ to exist.

Proposition 10 *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 $\mathcal{J}$.*

Proof. If $\mathcal{I} \in \mathit{Int}(\mathcal{J})$ is such that $\mathcal{I} \preceq_{\mathcal{J}} \mathcal{I}_0$, then the monotonicity of $O_{\mathcal{T},\mathcal{J}}$ implies that $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}) \preceq_{\mathcal{J}} O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_{\mathcal{J}} \mathcal{I}_0$. Consequently, $O_{\mathcal{T},\mathcal{J}}$ is also an operator on $\{\mathcal{I} \mid \mathcal{I} \preceq_{\mathcal{J}} \mathcal{I}_0\}$. Since it is monotonic, it has a greatest fixpoint in this set as well, which is obviously the $\mathcal{I}_0$-gfp-model of $\mathcal{T}$. □

Since $\mathcal{I}_0$ is the greatest element of the set $\{\mathcal{I} \mid \mathcal{I} \preceq_{\mathcal{J}} \mathcal{I}_0\}$, the proof of the proposition shows that the $\mathcal{I}_0$-gfp-model of $\mathcal{T}$ can be obtained by iterated application of the operator $O_{\mathcal{T},\mathcal{J}}$, starting with $\mathcal{I}_0$.

Corollary 11 *Let $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_{\mathcal{J}} \mathcal{I}_0$. We define $\mathcal{I}_0^{\downarrow 0} := \mathcal{I}_0$, $\mathcal{I}_0^{\downarrow \alpha+1} := O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0^{\downarrow\alpha})$, and $\mathcal{I}_0^{\downarrow\alpha} := glb(\{\mathcal{I}_0^{\downarrow\beta} \mid \beta < \alpha\}$ if α is a limit ordinal. Then there exists an ordinal α such that $\mathcal{I}_0^{\downarrow\alpha}$ is the $\mathcal{I}_0$-gfp-model of $\mathcal{T}$.*

3 Description graphs and simulations

In this section, we will show that $\mathcal{EL}$-TBoxes as well as primitive interpretations can be represented as description graphs. Then, we will introduce the notion of a simulation between nodes of a description graph, and show some useful properties of simulations.

3.1 Normalized $\mathcal{EL}$-TBoxes

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 $D = \top$.

First, we illustrate this normalization process by a typical example.

Example 12 Consider the $\mathcal{EL}$-TBox $\mathcal{T}$ consisting of the following concept definitions:

$$\begin{aligned}
A_1 &\equiv P_1 \sqcap A_2 \sqcap \exists r_1.\exists r_2.A_3, \\
A_2 &\equiv P_2 \sqcap A_3 \sqcap \exists r_2.\exists r_1.A_1, \\
A_3 &\equiv P_3 \sqcap A_2 \sqcap \exists r_1.(P_1 \sqcap P_2).
\end{aligned}$$

By introducing auxiliary definitions, we obtain the new TBox $\mathcal{T}'$:

$$\begin{array}{lcl} A_1 & \equiv & P_1 \sqcap A_2 \sqcap \exists r_1.B_1, \\ B_1 & \equiv & \exists r_2.A_3, \\ A_2 & \equiv & P_2 \sqcap A_3 \sqcap \exists r_2.B_2, \\ B_2 & \equiv & \exists r_1.A_1, \\ A_3 & \equiv & P_3 \sqcap A_2 \sqcap \exists r_1.B_3, \\ B_3 & \equiv & P_1 \sqcap P_2. \end{array}$$

This TBox is not yet normalized since the definitions of A_1, A_2 and A_3 contain defined concepts in their top-level conjunction.
Let us first concentrate on the definitions of A_2 and A_3. The occurrence of A_3 in the top-level conjunction of the definition of A_2 shows that A_2 is subsumed by A_3, and the occurrence of A_2 in the top-level conjunction of the definition of A_3 shows that A_3 is subsumed by A_2. Thus, the concepts A_2 and A_3 are equivalent (i.e., are interpreted by the same set in all models of the TBox). In addition A_2 (and the equivalent concept A_3) is subsumed by $P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3$. Thus, we can replace every occurrence of A_3 in $\mathcal{T}'$ by A_2, and the definition of A_2 by the inclusion constraint $A_2 \sqsubseteq P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3$, with the obvious semantics that the interpretation of A_2 must be contained in the interpretation of the concept description on the right-hand side:

$$\begin{array}{lcl} A_1 & \equiv & P_1 \sqcap A_2 \sqcap \exists r_1.B_1, \\ B_1 & \equiv & \exists r_2.A_2, \\ A_2 & \sqsubseteq & P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3, \\ B_2 & \equiv & \exists r_1.A_1, \\ B_3 & \equiv & P_1 \sqcap P_2. \end{array}$$

In order to transform this back into a TBox, we must get rid of the inclusion constraint. How to do this depends on the semantics used for cyclic definitions.
If we use *descriptive semantics*, then we can employ Nebel's approach [20] to turn inclusion statements into definitions: we introduce a new primitive concept $\bar{A}_2$ and replace $A_2 \sqsubseteq P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3$ by the definition

$$A_2 \equiv \bar{A}_2 \sqcap P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3.$$

For fixpoint semantics, this approach cannot be employed. The reason is that the interpretation of the primitive concept $\bar{A}_2$ is fixed by the primitive interpretation, and thus cannot be maximized or minimized.
If we use *gfp-semantics*, then we can replace the inclusion constraint by the definition

$$A_2 \equiv P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3.$$

In fact, this is the largest possible interpretation of A_2 that the inclusion constraint allows.

Finally, if we use *lfp-semantics*, then A_2 becomes unsatisfiable, i.e., A_2 is interpreted by the empty set in all lfp-models of the TBox together with the inclusion constraint. In fact, the empty set is the smallest interpretation of A_2 that the inclusion constraint allows. Consequently, all defined concepts whose right-hand sides contain A_2 are also interpreted by the empty set in all lfp-models. The same is true for all defined concepts whose defintions contain these concepts, etc. For this reason, we can remove from the TBox the inclusion constraint together with all definitions that refer (directly or indirectly) to A_2. (When deciding subsumption w.r.t. lfp-semantics, one must just keep in mind that all the concepts whose definitions have been removed are unsatisfiable, and thus are subsumed by all the other concepts.)
For the three types of semantics, we thus have shown how to remove the inclusion constraint. The TBoxes obtained this way still need not be in normal form since (for gfp- and descriptive semantics) the definition of A_1 still refers to A_2 on the top-level. However, we can now just replace the top-level A_2 in the definition of A_1 by its defining concept description. This way, we end up with a normalized TBox. For gfp- and descriptive semantics, we can now add a definition for A_3, which just has the same right-hand side as the definition of A_2.
With respect to gfp-semantics, we thus obtain the following normalized TBox $\mathcal{T}^{gfp}$:

$$\begin{array}{lcl}
A_1 & \equiv & P_1 \sqcap P_2 \sqcap P_3 \sqcap \exists r_1.B_1 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3, \\
B_1 & \equiv & \exists r_2.A_2, \\
A_2 & \equiv & P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3, \\
B_2 & \equiv & \exists r_1.A_1, \\
A_3 & \equiv & P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3, \\
B_3 & \equiv & P_1 \sqcap P_2,
\end{array}$$

and w.r.t. descriptive semantics, we obtain the normalized TBox $\mathcal{T}^{desc}$:

$$\begin{array}{lcl}
A_1 & \equiv & P_1 \sqcap \bar{A}_2 \sqcap P_2 \sqcap P_3 \sqcap \exists r_1.B_1 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3, \\
B_1 & \equiv & \exists r_2.A_2, \\
A_2 & \equiv & \bar{A}_2 \sqcap P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3, \\
B_2 & \equiv & \exists r_1.A_1, \\
A_3 & \equiv & \bar{A}_2 \sqcap P_2 \sqcap P_3 \sqcap \exists r_2.B_2 \sqcap \exists r_1.B_3, \\
B_3 & \equiv & P_1 \sqcap P_2.
\end{array}$$

With respect to lfp-semantics, only the definition

$$B_3 \equiv P_1 \sqcap P_2$$

remains, whereas all the other defined concepts are unsatisfiable.

The normalization approach used in the example can easily be generalized to arbitrary $\mathcal{EL}$-TBoxes. Assume, without loss of generality, that the introduction of auxiliary definitions (as illustrated in Example 12) has already been done. Let $\mathcal{G}$ be the graph whose nodes are the defined concepts of the TBox, and where there is an edge from A to B iff B occurs in the top-level conjunction of the definition of A. We write

- $B \preceq A$ iff there is a path in $\mathcal{G}$ leading from A to B,
- $A \cong B$ iff $A \preceq B$ and $B \preceq A$, and
- $B \prec A$ iff $B \preceq A$ and not $A \cong B$.

In Example 12 we have $A_2 \cong A_3$ and $A_2 \prec A_1$.

By definition, $\preceq$ is a quasi-ordering and $\cong$ is the equivalence induced by $\preceq$. On the $\cong$-equivalence classes, $\preceq$ induces a partial ordering:

$$[A] \preceq [B] \text{ iff } A \preceq B,$$

where $[C] = \{C' \mid C \cong C'\}$.

All the concepts that belong to the same $\cong$-equivalence class are obviously interpreted by the same set in all models of the TBox. We start with a minimal equivalence classes w.r.t. $\preceq$, and treat it as illustrated with the help of A_2 and A_3 in Example 12. Then, we replace the occurrences of elements of this class on the top-level by their new definition, and continue with the next equivalence class.

Since only top-level occurrences are replaced, the replacement of defined concepts by their definitions cannot lead to an exponential blow-up as in the general case (by using idempotency of $\sqcap$). To sum up, we have sketched how to prove the following proposition:

Proposition 13 *Subsumption between concepts defined in an $\mathcal{EL}$-TBox w.r.t. lfp-, gfp, and descriptive semantics can be reduced in polynomial time to subsumption between concepts defined in a normalized $\mathcal{EL}$-TBox.*

3.2 Description graphs

In the following, we will assume without loss of generality 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 subsection, we fix a normalized $\mathcal{EL}$-TBox $\mathcal{T}$ with primitive concepts N_{prim}, defined concepts N_{def}, and roles N_{role}.

Definition 14 An *$\mathcal{EL}$-description graph* is a graph $\mathcal{G} = (V, E, L)$ where

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

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

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

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

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

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

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

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

An example of an $\mathcal{EL}$-description graph can be found in Figure 1. The translation between $\mathcal{EL}$-TBoxes (primitive interpretations) to $\mathcal{EL}$-description graphs works in both directions, i.e., any $\mathcal{EL}$-description graph can also be view as an $\mathcal{EL}$-TBox (primitive interpretation). For example, the $\mathcal{EL}$-description graph of Figure 1 can also be viewed as representing the following primitive interpretation $\mathcal{J}$:

- $\Delta^\mathcal{J} := \{A_1, A_2, A_3, B_1, B_2, B_3\}$;
- $P_1^\mathcal{J} := \{A_1, B_3\}$, $P_2^\mathcal{J} := \{A_1, A_2, A_3, B_3\}$, and $P_3^\mathcal{J} := \{A_1, A_2, A_3\}$;
- $r_1^\mathcal{J} := \{(A_1, B_1), (A_1, B_3), (A_2, B_3), (A_3, B_3), (B_2, A_1)\}$ and
  $r_2^\mathcal{J} := \{(A_1, B_2), (A_2, B_2), (A_3, B_2), (B_1, A_2)\}$.

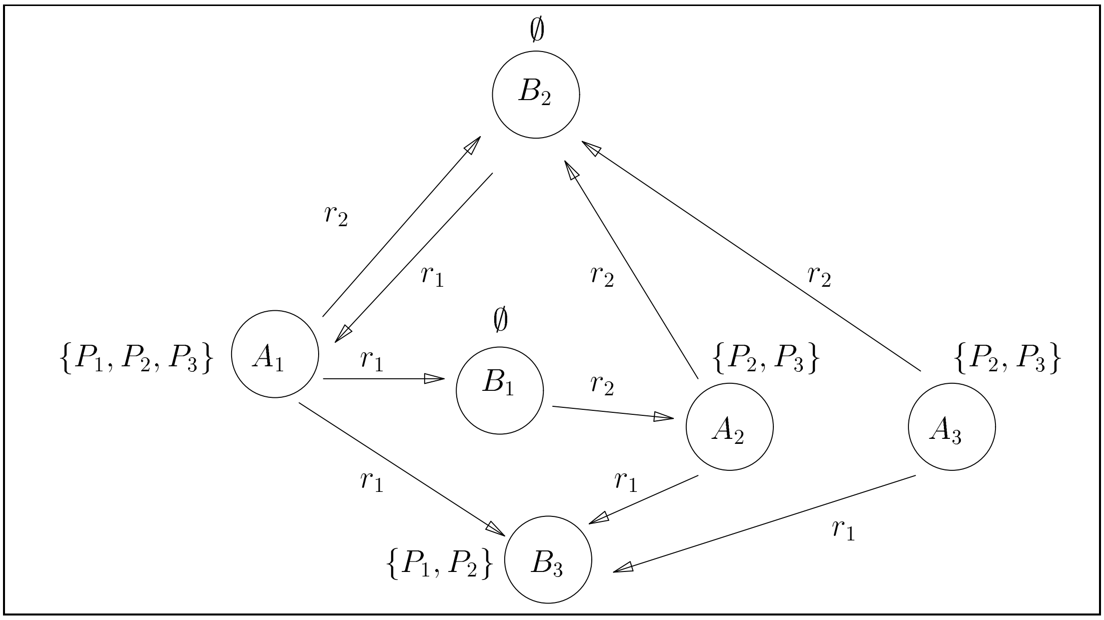

Figure 1: The $\mathcal{EL}$-description graph of the normalized TBox $\mathcal{T}^{gfp}$ in Example 12.

3.3 Simulations

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

Definition 15 Let $\mathcal{G}_i = (V_i, E_i, L_i)$ $(i = 1, 2)$ be two $\mathcal{EL}$-description graphs. The binary relation $Z \subseteq V_1 \times V_2$ is a *simulation* from $\mathcal{G}_1$ to $\mathcal{G}_2$ iff

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

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

We write Z: $\mathcal{G}_1 \overset{\rightharpoonup}{\sim} \mathcal{G}_2$ 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. Basically, one starts with

$$Z_0 := \{(v_1, v_2) \in V_1 \times V_2 \mid L_1(v_1) \subseteq L_2(v_2)\},$$

and then removes tuples if they violate (S2) until no more tuples can be removed. Since testing whether (S2) is violated for a given pair of nodes can be realized in polynomial time and Z_0 contains only polynomially many tuples, this procedures terminates in polynomial time, and it is easy to show that it computes the greatest simulation from $\mathcal{G}_1$ to $\mathcal{G}_2$. A more efficient algorithm for computing the greatest simulation between two finite graphs can be found in [12]. Its complexity is $O(mn)$, where m is the number of edges and n is the number of nodes of the two graphs (assuming that $m \geq n$).

Proposition 16 *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 be decide in polynomial time whether there is a simulation $Z\colon \mathcal{G}_1 \stackrel{\rightarrow}{\sim} \mathcal{G}_2$ such that $(v_1, v_2) \in Z$.*

Proof. It is easy to see that there is a simulation $Z\colon \mathcal{G}_1 \stackrel{\rightarrow}{\sim} \mathcal{G}_2$ such that $(v_1, v_2) \in Z$ iff the greatest simulation $\widehat{Z}\colon \mathcal{G}_1 \stackrel{\rightarrow}{\sim} \mathcal{G}_2$ satisfies $(v_1, v_2) \in \widehat{Z}$. Thus, the proposition immediately follows from the fact that $\widehat{Z}$ can be computed in polynomial time. □

Definition 15 also covers the case where $\mathcal{G}_1 = \mathcal{G}_2$. In this case, the identity on the nodes of $\mathcal{G}_1 = \mathcal{G}_2$ is a simulation. Consequently, the greatest simulation contains the identity.

We will later use the fact that the class of all simulations is closed under composition.

Lemma 17 *Let $\mathcal{G}_1, \mathcal{G}_2, \mathcal{G}_3$ be $\mathcal{EL}$-description graphs, and let $Z_1\colon \mathcal{G}_1 \stackrel{\rightarrow}{\sim} \mathcal{G}_2$ and $Z_2\colon \mathcal{G}_2 \stackrel{\rightarrow}{\sim} \mathcal{G}_3$ be simulations. Then*

$$Z_1 \circ Z_2 := \{(v, v'') \mid \textit{there exists } v' \textit{ such that } (v, v') \in Z_1 \textit{ and } (v', v'') \in Z_2\}$$

is also a simulation.

4 Subsumption w.r.t. gfp-semantics

In the following, let $\mathcal{T}$ be a normalized $\mathcal{EL}$-TBox with primitive concepts N_{prim}, defined concepts N_{def}, and roles N_{role}. In this section, we will show that $A \sqsubseteq_{gfp,\mathcal{T}} B$ holds for two defined concepts A, B iff there is a simulation $Z\colon \mathcal{G}_\mathcal{T} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{T}$ such that $(B, A) \in Z$. As an auxiliary result we give a characterization of when an individual of a gfp-model belongs to a defined concept in this model.

Proposition 18 *Let $\mathcal{J}$ be a primitive interpretation and $\mathcal{I}$ the gfp-model of $\mathcal{T}$ based on $\mathcal{J}$. Then the following are equivalent for any $A \in N_{def}$ and $x \in \Delta^\mathcal{J}$:*

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

Proof. Let $\mathcal{G}_\mathcal{T} = (N_{def}, E_\mathcal{T}, L_\mathcal{T})$ and $\mathcal{G}_\mathcal{J} = (\Delta^\mathcal{J}, E_\mathcal{J}, L_\mathcal{J})$.

$(1 \Rightarrow 2)$ Assume that $x \in A^\mathcal{I}$. The relation $Z \subseteq N_{def} \times \Delta^\mathcal{J}$ is defined as follows:

$$Z := \{(B, y) \in N_{def} \times \Delta^\mathcal{J} \mid y \in B^\mathcal{I}\}.$$

Since $x \in A^\mathcal{I}$, we have $(A, x) \in Z$. It remains to be shown that Z satisfies (S1) and (S2) of Definition 15. Thus, let $(B, y) \in Z$, and let

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

be the definition of B in $\mathcal{T}$.

(S1) Since $(B, y) \in Z$, we have $y \in B^\mathcal{I}$, and thus $y \in P_i^\mathcal{I} = P_i^\mathcal{J}$ for $i = 1, \ldots, m$. Consequently, $L_\mathcal{T}(B) = \{P_1, \ldots, P_m\} \subseteq \{P \in N_{prim} \mid y \in P^\mathcal{J}\} = L_\mathcal{J}(y)$.

(S2) Now consider B_i with $(B, r_i, B_i) \in E_\mathcal{T}$. Since $y \in B^\mathcal{I} \subseteq (\exists r_i.B_i)^\mathcal{I}$, we know that there exists a $y_i \in \Delta^\mathcal{J}$ such that $(y, y_i) \in r_i^\mathcal{J}$ and $y_i \in B_i^\mathcal{I}$. But then we have $(y, r_i, y_i) \in E_\mathcal{J}$ and $(B_i, y_i) \in Z$.

$(2 \Rightarrow 1)$ Assume that $Z\colon \mathcal{G}_\mathcal{T} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{J}$ is a simulation such that $(A, x) \in Z$. Since $\mathcal{I}$ is the gfp-model of $\mathcal{T}$ based on $\mathcal{J}$, there is an ordinal α such that $\mathcal{I} = \mathcal{I}^{\downarrow\alpha}$.

Now, we consider triples (B, y, β) consisting of a defined concept $B \in N_{def}$, an individual $y \in \Delta^\mathcal{J}$, and an ordinal β, and show (by transfinite induction on β) that $(B, y) \in Z$ implies $y \in B^{\mathcal{I}^{\downarrow\beta}}$. For the triple (A, x, α) this yields $x \in A^{\mathcal{I}^{\downarrow\alpha}} = A^\mathcal{I}$.

Assume that $(B, y) \in Z$, but $y \notin B^{\mathcal{I}^{\downarrow\beta}}$.

Case 1: β is a limit ordinal. Then we have

$$B^{\mathcal{I}^{\downarrow\beta}} = B^{glb(\{\mathcal{I}^{\downarrow\gamma} \mid \gamma < \beta\})} = \bigcap_{\gamma < \beta} B^{\mathcal{I}^{\downarrow\gamma}},$$

and thus there exists an ordinal $\gamma < \beta$ such that $(B, y) \in Z$, but $y \notin B^{\mathcal{I}^{\downarrow\gamma}}$. However, the induction assumption for the smaller ordinal γ says that $(B, y) \in Z$ implies $y \in B^{\mathcal{I}^{\downarrow\gamma}}$.

Case 2: β is a successor ordinal, i.e., $\beta = \gamma + 1$. Let

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

be the definition of B in $\mathcal{T}$. Then,

$$\begin{aligned} B^{\mathcal{I}^{\downarrow\beta}} = O_{\mathcal{T},\mathcal{J}}(B^{\mathcal{I}^{\downarrow\gamma}}) &= (P_1 \sqcap \ldots \sqcap P_m)^{\mathcal{I}^{\downarrow\gamma}} \cap (\exists r_1.B_1 \sqcap \ldots \sqcap \exists r_\ell.B_\ell)^{\mathcal{I}^{\downarrow\gamma}} \\ &= P_1^\mathcal{J} \cap \ldots \cap P_m^\mathcal{J} \cap (\exists r_1.B_1 \sqcap \ldots \sqcap \exists r_\ell.B_\ell)^{\mathcal{I}^{\downarrow\gamma}}. \end{aligned}$$

Since $(B, y) \in Z$ implies $L_\mathcal{T}(B) = \{P_1, \ldots, P_m\} \subseteq \{P \in N_{prim} \mid y \in P^\mathcal{J}\} = L_\mathcal{J}(y)$, we know that $y \in P_i^\mathcal{J}$ for all $i = 1, \ldots, m$. Consequently, $y \notin B^{\mathcal{I}^{\downarrow\beta}}$ is due to the fact that $y \notin (\exists r_j.B_j)^{\mathcal{I}^{\downarrow\gamma}}$ for some $j, 1 \leq j \leq \ell$.

Since $(B, y) \in Z$ and $(B, r_j, B_j) \in E_{\mathcal{T}}$, the fact that Z is a simulation implies that there exists an individual $y' \in \Delta^{\mathcal{J}}$ such that $(y, r_j, y') \in E_{\mathcal{J}}$ and $(B_j, y') \in Z$. This yields $(y, y') \in r_j^{\mathcal{J}}$ (by definition of $E_{\mathcal{J}}$) and $y' \in B_j^{\mathcal{I}\downarrow\gamma}$ (by induction since $\gamma < \beta$). But then $y \in (\exists r_j.B_j)^{\mathcal{I}\downarrow\gamma}$, contradicting our assumption that $y \notin (\exists r_j.B_j)^{\mathcal{I}\downarrow\gamma}$ is responsible for the fact that $y \notin B^{\mathcal{I}\downarrow\beta}$. □

This proposition can now be used to prove the following characterization of subsumption w.r.t. gfp-semantics in $\mathcal{EL}$.

Theorem 19 *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and A, B defined concepts in $\mathcal{T}$. Then the following are equivalent:*

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

Proof. $(2 \Rightarrow 1)$ Assume that the simulation Z: $\mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{T}}$ satisfies $(B, A) \in Z$. Let $\mathcal{J}$ be a primitive interpretation and $\mathcal{I}$ the gfp-model of $\mathcal{T}$ based on $\mathcal{J}$. We must show that $x \in A^{\mathcal{I}}$ implies $x \in B^{\mathcal{I}}$.

By Proposition 18, $x \in A^{\mathcal{I}}$ implies that there is a simulation Y: $\mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}}$ such that $(A, x) \in Y$. But then $X := Z \circ Y$ is a simulation from $\mathcal{G}_{\mathcal{T}}$ to $\mathcal{G}_{\mathcal{J}}$ such that $(B, x) \in X$. By Proposition 18, this implies $x \in B^{\mathcal{I}}$.

$(1 \Rightarrow 2)$ Assume that $A \sqsubseteq_{gfp,\mathcal{T}} B$. We consider the graph $\mathcal{G}_{\mathcal{T}}$, and view it as an $\mathcal{EL}$-description graph of a primitive interpretation. Thus, let $\mathcal{J}$ be the primitive interpretation with $\mathcal{G}_{\mathcal{T}} = \mathcal{G}_{\mathcal{J}}$, and let $\mathcal{I}$ be the gfp-model of $\mathcal{T}$ based on $\mathcal{J}$.

Since the identity is a simulation *Id*: $\mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{T}} = \mathcal{G}_{\mathcal{J}}$ that satisfies $(A, A) \in Id$, Proposition 18 yields $A \in A^{\mathcal{I}}$. But then $A \sqsubseteq_{gfp,\mathcal{T}} B$ implies $A \in B^{\mathcal{I}}$, and thus Proposition 18 yields the existence of a simulation Z: $\mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}} = \mathcal{G}_{\mathcal{T}}$ such that $(B, A) \in Z$. □

The theorem together with Proposition 16 shows that subsumption w.r.t. gfp-semantics in $\mathcal{EL}$ is tractable.

Corollary 20 *Subsumption w.r.t. gfp-semantics in $\mathcal{EL}$ can be decided in polynomial time.*

This result is quite surprising since, for the DL $\mathcal{FL}_0$ (which allows for conjunction and value restrictions only), subsumption w.r.t. gfp-semantics is already PSPACE-complete.

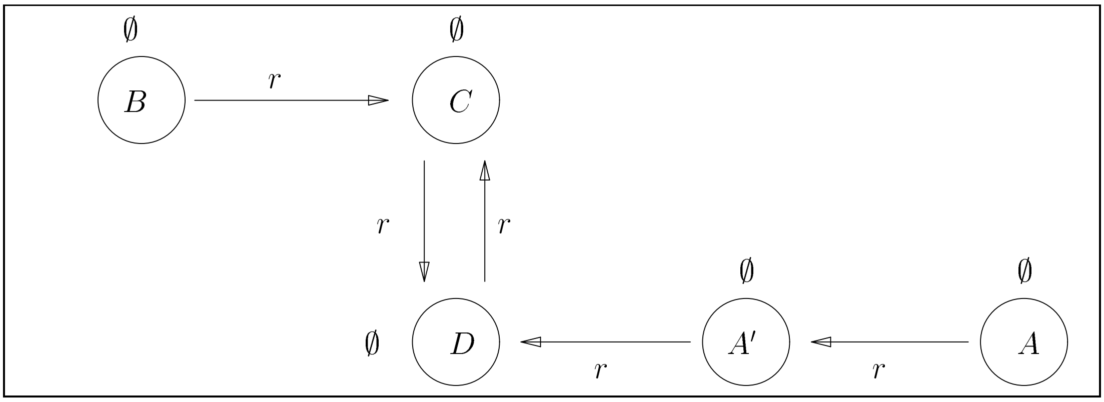

Figure 2: The $\mathcal{EL}$-description graph of the TBox in Example 21.

Example 21 Consider the TBox $\mathcal{T}$ consisting of the following concept definitions:

$$B \equiv \exists r.C, \quad C \equiv \exists r.D, \quad D \equiv \exists r.C,$$
$$A \equiv \exists r.A', \quad A' \equiv \exists r.D.$$

The $\mathcal{EL}$-description graph $\mathcal{G}_\mathcal{T}$ corresponding to this TBox can be found in Figure 2. Let $V_\mathcal{T} = \{A, A', B, C, D\}$ denote the set of nodes of this graph. Then $Z := V \times V$ is a simulation from $\mathcal{G}_\mathcal{T}$ to $\mathcal{G}_\mathcal{T}$. Consequently, all the defined concepts in $\mathcal{T}$ subsume each other w.r.t. gfp-semantics.

5 Subsumption w.r.t. lfp-semantics

For the sake of completeness, we also treat lfp-semantics in this report. It should be noted, however, that the results of this section demonstrate that lfp-semantics is not interesting in $\mathcal{EL}$.

Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and $\mathcal{G}_\mathcal{T}$ the corresponding $\mathcal{EL}$-description graph. Where A, B are nodes of $\mathcal{G}_T$, we write $A \overset{*}{\to}_\mathcal{T} B$ to denote that there is a path in $\mathcal{G}_T$ from A to B, and $A \overset{+}{\to}_\mathcal{T} B$ to denote that there is a non-empty path in $\mathcal{G}_T$ from A to B. We define

$$Cyc_\mathcal{T} := \{A \mid \text{there exists a node } B \text{ such that } A \overset{*}{\to}_\mathcal{T} B \overset{+}{\to}_\mathcal{T} B\},$$

i.e., $Cyc_\mathcal{T}$ consists of the nodes in $\mathcal{G}_T$ that can reach a cyclic path in $\mathcal{G}_T$. The following lemma is an easy consequence of the definition of $Cyc_\mathcal{T}$.

Lemma 22 *If $A \in Cyc_\mathcal{T}$, then there exist a defined concept $A' \in Cyc_\mathcal{T}$ and a role r such that (A, r, A') is an edge in $\mathcal{G}_\mathcal{T}$.*

Proposition 23 *Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and A a defined concept in $\mathcal{T}$. If $A \in Cyc_{\mathcal{T}}$, then A is unsatisfiable w.r.t. lfp-semantics, i.e., $A^{\mathcal{I}} = \emptyset$ holds for all lfp-models $\mathcal{I}$ of $\mathcal{T}$.*

Proof. Let $\mathcal{J}$ be a primitive interpretation and $\mathcal{I}$ the lfp-model of $\mathcal{T}$ based on $\mathcal{J}$. Since $O_{\mathcal{T},\mathcal{J}}$ is upward continuous by Proposition 7, we know that $\mathcal{I} = \mathcal{I}^{\uparrow\omega}$, and thus $A^{\mathcal{I}} = \bigcup_{n\geq 0} A^{\mathcal{I}^{\uparrow n}}$. We show by induction on n that $A^{\mathcal{I}^{\uparrow n}} = \emptyset$ holds for all $n \geq 0$, which yields $A^{\mathcal{I}} = \emptyset$.

$(n = 0)$ $A^{\mathcal{I}^{\uparrow 0}} = \emptyset$ by definition of $\mathcal{I}^{\uparrow 0}$.

$(n \to n+1)$ By Lemma 22 there exists a defined concept $A' \in Cyc_{\mathcal{T}}$ and a role r such that (A, r, A') is an edge in $\mathcal{G}_{\mathcal{T}}$. Thus, if $A \equiv D$ is the definition of A in $\mathcal{T}$, then D contains the conjunct $\exists r.A'$ in its top-level conjunction. By induction, we know that $A'^{\mathcal{I}^{\uparrow n}} = \emptyset$, and thus $A^{\mathcal{I}^{\uparrow n+1}} = O_{\mathcal{T},\mathcal{J}}(A^{\mathcal{I}^{\uparrow n}}) = D^{\mathcal{I}^{\uparrow n}} = \emptyset$. □

Since all the defined concepts in $Cyc_{\mathcal{T}}$ are unsatisfiable, their definitions can be removed from the TBox without changing the meaning of the concepts whose definition does not refer to an element of $Cyc_{\mathcal{T}}$. This leaves us with an acyclic terminology. Consequently, the only thing that cyclic definitions can express w.r.t. lfp-semantics in $\mathcal{EL}$ is unsatisfiability of a defined concept. However, since in $\mathcal{EL}$ all concepts referring to an unsatisfiable concept are also unsatisfiable, this does not buy us much.

In Example 21, all the defined concepts belong to $Cyc_{\mathcal{T}}$, and thus they are all unsatisfiable w.r.t. lfp-semantics.

Corollary 24 *Subsumption w.r.t. lfp-semantics in $\mathcal{EL}$ can be decided in polynomial time.*

Proof. Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and A, B be defined concepts in $\mathcal{T}$. We want to decide whether or not $A \sqsubseteq_{lfp,\mathcal{T}} B$ holds. Obviously, $Cyc_{\mathcal{T}}$ can be computed in polynomial time.

Case 1: A and B do not belong to $Cyc_{\mathcal{T}}$. Let $\mathcal{T}'$ be the TBox obtained from $\mathcal{T}$ by removing all the definitions for elements in $Cyc_{\mathcal{T}}$. It is easy to see that $\mathcal{T}'$ is an acyclic TBox that does not contain any of the concept names in $Cyc_{\mathcal{T}}$ (also not on the right-hand side of a definition). Since the definitions of A, B do not refer to any element of $Cyc_{\mathcal{T}}$, we have $A \sqsubseteq_{lfp,\mathcal{T}} B$ iff $A \sqsubseteq_{lfp,\mathcal{T}'} B$. Since $\mathcal{T}'$ is acyclic, lfp-semantics and gfp-semantics agree on $\mathcal{T}'$ [22], and thus $A \sqsubseteq_{lfp,\mathcal{T}} B$ iff $A \sqsubseteq_{gfp,\mathcal{T}'} B$. By Corollary 20, $A \sqsubseteq_{gfp,\mathcal{T}'} B$ can be decided in polynomial time.

Case 2: $A \in Cyc_{\mathcal{T}}$. Since any lfp-model of $\mathcal{T}$ interprets A by the empty set, we clearly have $A \sqsubseteq_{lfp,\mathcal{T}} B$.

Case 3: $A \notin Cyc_{\mathcal{T}}$ and $B \in Cyc_{\mathcal{T}}$. Then $A \sqsubseteq_{lfp,\mathcal{T}} B$ does *not* hold. To see this it is enough to show that there is an lfp-model of $\mathcal{T}$ that interprets A by a non-empty set. Consider the TBox $\mathcal{T}'$ constructed in Case 1. Any lfp-model of $\mathcal{T}'$ can be extended to an lfp-model of $\mathcal{T}$ by assigning the empty set to the elements of $Cyc_{\mathcal{T}}$. However, the lfp-models of $\mathcal{T}'$ are just the gfp-models of $\mathcal{T}'$. Now, let us view $\mathcal{G}_{\mathcal{T}'}$ as the graph of a primitive interpretation $\mathcal{J}$, and let $\mathcal{I}$ be the gfp-model based on $\mathcal{J}$. The identity on the nodes of $\mathcal{G}_{\mathcal{T}'}$ is a simulation that contains the pair (A, A). By Proposition 18, this shows that $A \in A^{\mathcal{I}}$. □

6 Subsumption w.r.t. descriptive semantics

Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox and $\mathcal{G}_{\mathcal{T}}$ the corresponding $\mathcal{EL}$-description graph. Since every gfp-model of $\mathcal{T}$ is a model of $\mathcal{T}$, $A \sqsubseteq_{\mathcal{T}} B$ implies $A \sqsubseteq_{gfp,\mathcal{T}} B$. Consequently, $A \sqsubseteq_{\mathcal{T}} B$ implies that there is a simulation $Z\colon \mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{T}}$ with $(B, A) \in Z$. In the following we will show what additional properties the simulation Z must satisfy for the implication in the other direction to hold.

To get an intuition on the difference between gfp- and descriptive semantics, let us consider Example 21. With respect to gfp-semantics, all the defined concepts of $\mathcal{T}$ are equivalent (i.e., subsume each other). With respect to descriptive semantics, A, B, D are still equivalent, C is equivalent to A', but A' is not equivalent to B, and C and D are also not equivalent (in both cases, the concepts are not even comparable w.r.t. subsumption).

To see that C and A' are equivalent w.r.t. descriptive semantics, it is enough to note that the following identities hold in every model $\mathcal{I}$ of $\mathcal{T}$:

$$A'^{\mathcal{I}} = (\exists r.D)^{\mathcal{I}} = C^{\mathcal{I}}.$$

A similar argument shows that B and D are equivalent. In addition, equivalence of C and A' obviously also implies equivalence of A and B. The following model of $\mathcal{T}$ is a counterexample to the other subsumption relationships:

1. $\Delta^{\mathcal{I}} := \{c, d\}$;
2. $r^{\mathcal{I}} := \{(c, d), (d, c)\}$;
3. $A^{\mathcal{I}} := \{d\}$, $A'^{\mathcal{I}} := \{c\}$, $C^{\mathcal{I}} := \{c\}$, $D^{\mathcal{I}} := \{d\}$, $B^{\mathcal{I}} := \{d\}$.

We will see below that the reason for A' and B not being equivalent is that in the infinite path in $\mathcal{G}_{\mathcal{T}}$ starting with A', one reaches D with an odd number of edges, whereas C is reached with an even number; for the path starting with B, it is just the opposite. In contrast, the infinite paths starting respectively with A and B "synchronize" after a finite number of steps.

To formalize this intuition, we must introduce some notation. Let $\mathcal{T}$ be an $\mathcal{EL}$-TBox, $\mathcal{G}_{\mathcal{T}}$ the corresponding $\mathcal{EL}$-description graph, and $Z\colon \mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{T}}$ a simulation.

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

Figure 3: A (B, A)-simulation chain.

Definition 25 The path p_1: $B = B_0 \xrightarrow{r_1} B_1 \xrightarrow{r_2} B_2 \xrightarrow{r_3} B_3 \xrightarrow{r_4} \cdots$ is *Z-simulated* by the path p_2: $A = A_0 \xrightarrow{r_1} A_1 \xrightarrow{r_2} A_2 \xrightarrow{r_3} A_3 \xrightarrow{r_4} \cdots$ 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 3).

Consider the TBox $\mathcal{T}$ and the simulation Z introduced in Example 21. Then

$$\begin{array}{ccccccccccc} B & \xrightarrow{r} & C & \xrightarrow{r} & D & \xrightarrow{r} & C & \xrightarrow{r} & D & \xrightarrow{r} & \cdots \\ Z\downarrow & & Z\downarrow & & Z\downarrow & & Z\downarrow & & Z\downarrow & & \\ A & \xrightarrow{r} & A' & \xrightarrow{r} & D & \xrightarrow{r} & C & \xrightarrow{r} & D & \xrightarrow{r} & \cdots \end{array}$$

is a (B, A)-simulation chain w.r.t. Z, and

$$\begin{array}{ccccccccccc} B & \xrightarrow{r} & C & \xrightarrow{r} & D & \xrightarrow{r} & C & \xrightarrow{r} & D & \xrightarrow{r} & \cdots \\ Z\downarrow & & Z\downarrow & & Z\downarrow & & Z\downarrow & & Z\downarrow & & \\ A' & \xrightarrow{r} & D & \xrightarrow{r} & C & \xrightarrow{r} & D & \xrightarrow{r} & C & \xrightarrow{r} & \cdots \end{array}$$

is a (B, A')-simulation chain w.r.t. Z. Note that the first chain synchronizes after a finite number of steps in the sense that there is a Z-link (in fact infinitely many in this case) between the same defined concept. In contrast, the second chain does not synchronize in this sense. We will see below that this is responsible for the fact that A is subsumed by B w.r.t. descriptive semantics, but A' is not subsumed by B w.r.t. descriptive semantics.

If $(B, A) \in Z$, then (S2) of Definition 15 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 26 A *partial* (B, A)*-simulation chain* is of the form depicted in Figure 4. 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

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

Figure 4: A partial (B, A)-simulation chain.

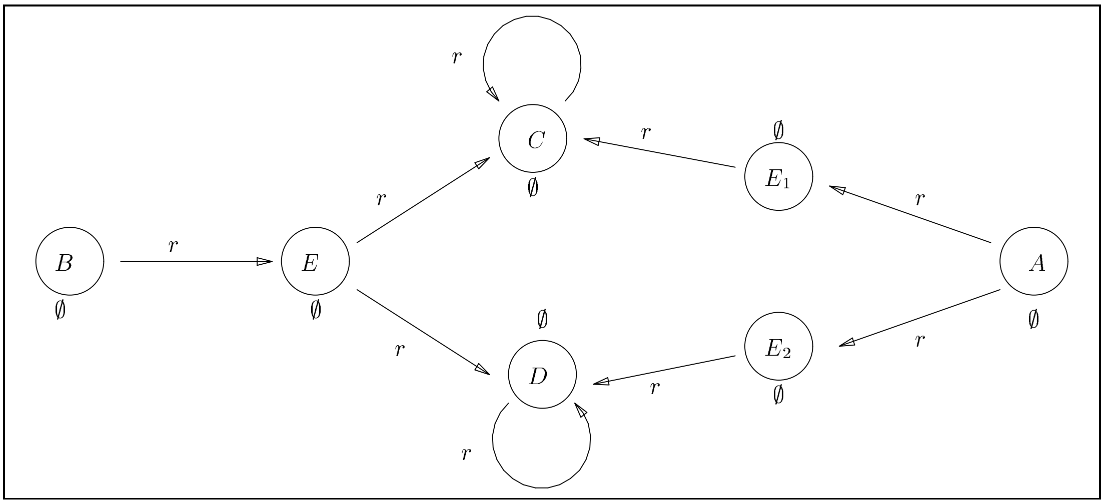

Figure 5: An $\mathcal{EL}$-description graph of a cyclic $\mathcal{EL}$-TBox.

Z-simulating path. In this case we say that the resulting (B, A)-simulation chain is S*-selected*.

Example 27 Consider the $\mathcal{EL}$-description graph of Figure 5. Where V denotes the set of all nodes of this graph, it is easy to see that $Z := V \times V$ is a simulation such that $(B, A) \in Z$. There are two selection functions for A, B and Z. The function S_1 that assigns E_1 to the partial (B, A)-simulation chain

$$\begin{array}{ccc} B & \xrightarrow{r} & E \\ Z\downarrow & & \\ A & & \end{array}$$

and the function S_2 that assigns E_2 to this chain.

Definition 28 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 S-selected (B, A)-simulation chain of the form depicted in Figure 3 there exists an $i \geq 0$ such that $A_i = B_i$.

The simulation Z of Example 27 is not (B, A)-synchronized. In fact, if we take the selection function S_1, then the S_1-selected (B, A)-simulation chain induced by the infinite path $B \xrightarrow{r} E \xrightarrow{r} D \xrightarrow{r} D \xrightarrow{r} \cdots$ does not satisfy the condition stated in Definition 28. If we take the selection function S_2 instead, then the S_2-selected (B, A)-simulation chain induced by the infinite path $B \xrightarrow{r} E \xrightarrow{r} C \xrightarrow{r} C \xrightarrow{r} \cdots$ does not satisfy this condition.

We are now ready to state our characterization of subsumption w.r.t. descriptive semantics.

Theorem 29 *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 $Z \colon \mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{T}}$ such that $(B, A) \in Z$.*

As in the case of gfp-semantics, we prove the theorem by first giving a characterization of when an individual of a model belongs to a defined concept in this model. 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 30 *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 $Z \colon \mathcal{G}_{\mathcal{T}} \overset{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}}$ such that*
   - *(a) $(A, x) \in Z$; and*
   - *(b) if $(B, y) \in Z$ then $y \in B^{\mathcal{I}_0}$.*

Proof. Instead of proving this result directly, we will reduce it to Proposition 18. To this purpose, we define the new TBox

$$\mathcal{T}' := \{B \equiv D \sqcap P_B \mid B \equiv D \in \mathcal{T}\},$$

where the P_B are new primitive concepts. Obviously, $\mathcal{T}$ and $\mathcal{T}'$ have the same defined concepts. For $\mathcal{T}'$ we define the primitive interpretation $\mathcal{J}'$ as follows:

- $\Delta^{\mathcal{J}'} := \Delta^{\mathcal{J}}$;
- $r^{\mathcal{J}'} := r^{\mathcal{J}}$ for all role names r;
- $P^{\mathcal{J}'} := P^{\mathcal{J}}$ if P is a primitive concept in $\mathcal{T}$;

- $P_B^{\mathcal{J}'} := B^{\mathcal{I}_0}$ where B is a defined concept.

We define:

- $\mathcal{I}^{\downarrow 0} := \mathcal{I}_0$ and $\mathcal{I}'^{\downarrow 0} := \mathcal{I}'_{top}$, where $\mathcal{I}'_{top}$ is the interpretation based on $\mathcal{J}'$ such that $B^{\mathcal{I}'_{top}} = \Delta^{\mathcal{J}'}$ for all defined concepts B;
- $\mathcal{I}^{\downarrow \alpha+1} := O_{\mathcal{T},\mathcal{J}}(\mathcal{I}^{\downarrow \alpha})$ and $\mathcal{I}'^{\downarrow \alpha+1} := O_{\mathcal{T}',\mathcal{J}'}(\mathcal{I}'^{\downarrow \alpha})$;
- $\mathcal{I}^{\downarrow \alpha} := glb(\{\mathcal{I}^{\downarrow \beta} \mid \beta < \alpha\}$ and $\mathcal{I}'^{\downarrow \alpha} := glb(\{\mathcal{I}'^{\downarrow \beta} \mid \beta < \alpha\}$ if α is a limit ordinal.

Let B be a defined concept. We claim that $B^{\mathcal{I}^{\downarrow n}} \supseteq B^{\mathcal{I}'^{\downarrow n+1}} \supseteq B^{\mathcal{I}^{\downarrow n+1}}$ holds for all $n \geq 0$. Before proving this claim, we show that it implies the statement of the proposition.

The claim obviously implies that $\mathcal{I}^{\downarrow \omega}$ agrees with $\mathcal{I}'^{\downarrow \omega}$ on all defined concepts, and thus the same is true for all larger ordinals. This implies that the $\mathcal{I}_0$-gfp-model $\mathcal{I}$ of $\mathcal{T}$ based on $\mathcal{J}$ agrees on all defined concepts with the gfp-model $\mathcal{I}'$ of $\mathcal{T}'$ based on $\mathcal{J}'$. Consequently, $x \in A^{\mathcal{I}}$ iff $x \in A^{\mathcal{I}'}$.

By Proposition 18, $x \in A^{\mathcal{I}'}$ is equivalent to the existence of a simulation $Z'\colon \mathcal{G}_{\mathcal{T}'} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}'}$ such that $(A, x) \in Z'$. The only difference between $\mathcal{G}_{\mathcal{T}'}$ and $\mathcal{G}_{\mathcal{T}}$ is that in $\mathcal{G}_{\mathcal{T}'}$ the label of each node B additionally contains P_B. The only difference between $\mathcal{G}_{\mathcal{J}'}$ and $\mathcal{G}_{\mathcal{J}}$ is that in $\mathcal{G}_{\mathcal{J}'}$ the labels of nodes may additionally contain the new primitive concepts P_B. Consequently, Z' is a simulation also from $\mathcal{G}_{\mathcal{T}}$ to $\mathcal{G}_{\mathcal{J}}$. In addition, $(B, y) \in Z'$ implies that P_B belongs to the label of y in $\mathcal{G}_{\mathcal{J}'}$, and thus $y \in P_B^{\mathcal{J}'} = B^{\mathcal{I}_0}$. Conversely, if $Z\colon \mathcal{G}_{\mathcal{T}} \stackrel{\rightarrow}{\sim} \mathcal{G}_{\mathcal{J}}$ is a simulation satisfying ($2b$) of the proposition, then it is also a simulation from $\mathcal{G}_{\mathcal{T}'}$ to $\mathcal{G}_{\mathcal{J}'}$.

To finish the proof of the proposition, we show by induction on n that $B^{\mathcal{I}^{\downarrow n}} \supseteq B^{\mathcal{I}'^{\downarrow n+1}} \supseteq B^{\mathcal{I}^{\downarrow n+1}}$ holds for all $n \geq 0$. Let $B \equiv D$ be the definition of B in $\mathcal{T}$. The definition of B in $\mathcal{T}'$ is then $B \equiv D \sqcap P_B$.

($n = 0$) We have

$$\begin{array}{rcl} B^{\mathcal{I}^{\downarrow 1}} & = & B^{O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0)} = D^{\mathcal{I}_0} \text{ and} \\ B^{\mathcal{I}'^{\downarrow 1}} & = & B^{O_{\mathcal{T}',\mathcal{J}'}(\mathcal{I}'_{top})} = D^{\mathcal{I}'_{top}} \cap P_B^{\mathcal{J}'} = D^{\mathcal{I}'_{top}} \cap B^{\mathcal{I}_0}. \end{array}$$

Monotonicity of the concept constructors of $\mathcal{EL}$ implies that $D^{\mathcal{I}_0} \subseteq D^{\mathcal{I}'_{top}}$ and the assumption $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_{\mathcal{J}} \mathcal{I}_0$ yields $B^{\mathcal{I}^{\downarrow 1}} = B^{O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0)} \subseteq B^{\mathcal{I}_0} = B^{\mathcal{I}^{\downarrow 0}}$. Thus, we have $B^{\mathcal{I}^{\downarrow 1}} \subseteq D^{\mathcal{I}'_{top}}$ and $B^{\mathcal{I}^{\downarrow 1}} \subseteq B^{\mathcal{I}_0}$, which taken together yields $B^{\mathcal{I}^{\downarrow 1}} \subseteq D^{\mathcal{I}'_{top}} \cap B^{\mathcal{I}_0} = B^{\mathcal{I}'^{\downarrow 1}} \subseteq B^{\mathcal{I}_0} = B^{\mathcal{I}^{\downarrow 0}}$.

($n \to n + 1$) Assume that $B^{\mathcal{I}^{\downarrow i}} \supseteq B^{\mathcal{I}'^{\downarrow i+1}} \supseteq B^{\mathcal{I}^{\downarrow i+1}}$ holds for all $i \leq n$. Then we have

1. $B^{\mathcal{I}\downarrow n+2} = D^{\mathcal{I}\downarrow n+1} \subseteq D^{\mathcal{I}'\downarrow n+1}$ by induction and the monotonicity of the concept constructors of $\mathcal{EL}$.

2. $B^{\mathcal{I}\downarrow n+2} \subseteq B^{\mathcal{I}\downarrow n+1} \subseteq B^{\mathcal{I}\downarrow n+1} \subseteq \cdots \subseteq B^{\mathcal{I}_0} = P_B^{\mathcal{J}'}$ since $\mathcal{I}^{\downarrow 0} = \mathcal{I}_0 \succeq_{\mathcal{J}} O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) = \mathcal{I}^{\downarrow 1}$ and the monotonicity of $O_{\mathcal{T},\mathcal{J}}$ imply $\mathcal{I}^{\downarrow 0} \succeq_{\mathcal{J}} \mathcal{I}^{\downarrow 1} \succeq_{\mathcal{J}} \mathcal{I}^{\downarrow 2} \succeq_{\mathcal{J}} \cdots$.

3. Consequently $B^{\mathcal{I}\downarrow n+2} \subseteq D^{\mathcal{I}'\downarrow n+1} \cap P_B^{\mathcal{J}'} = B^{O_{\mathcal{T}',\mathcal{J}'}(\mathcal{I}'\downarrow n+1)} = B^{\mathcal{I}'\downarrow n+2}$.

4. Finally, $B^{\mathcal{I}'\downarrow n+2} = D^{\mathcal{I}'\downarrow n+1} \cap P_B^{\mathcal{J}'} = D^{\mathcal{I}'\downarrow n+1} \cap B^{\mathcal{I}_0} \subseteq D^{\mathcal{I}\downarrow n} \cap B^{\mathcal{I}_0} = B^{\mathcal{I}\downarrow n+1}$. The inclusion holds by induction and the monotonicity of the concept constructors of $\mathcal{EL}$, and the last identity holds since $D^{\mathcal{I}\downarrow n} = B^{\mathcal{I}\downarrow n+1} \subseteq B^{\mathcal{I}\downarrow 0} = B^{\mathcal{I}_0}$.

To sum up, we have shown $B^{\mathcal{I}\downarrow n+2} \subseteq B^{\mathcal{I}'\downarrow n+2} \subseteq B^{\mathcal{I}\downarrow n+1}$, which completes the induction proof. □

Proof of (2) → (1) of Theorem 29

Assume that $Z\colon \mathcal{G}_{\mathcal{T}} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{T}}$ is a (B, A)-synchronized simulation such that $(B, A) \in Z$, and let S be the selection function required in the definition of a (B, A)-synchronized simulation.

To show $A \sqsubseteq_{\mathcal{T}} B$, we consider an arbitrary model $\mathcal{I}$ of $\mathcal{T}$ such that $x \in A^{\mathcal{I}}$, and show that $x \in B^{\mathcal{I}}$. Let $\mathcal{J}$ be the primitive interpretation on which $\mathcal{I}$ is based. Then $\mathcal{I}$ is itself the $\mathcal{I}$-gfp-model of $\mathcal{T}$ based on $\mathcal{J}$. Consequently, Proposition 30 shows that $x \in A^{\mathcal{I}}$ implies the existence of a simulation $Y\colon \mathcal{G}_{\mathcal{T}} \overset{\rightharpoonup}{\sim} \mathcal{G}_{\mathcal{J}}$ such that

(a) $(A, x) \in Y$, and

(b) $(C, y) \in Y$ implies $y \in C^{\mathcal{I}}$.

Now, assume that $x \notin B^{\mathcal{I}}$. 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 $x \notin P_i^{\mathcal{I}} = P_i^{\mathcal{J}}$ or an index $j, 1 \leq j \leq \ell$ such that $x \notin (\exists s_j.C_j)^{\mathcal{I}}$. The facts that $(B, A) \in Z$ and $x \in A^{\mathcal{I}}$ obviously imply that the first alternative cannot occur. Thus, there is an index $j, 1 \leq j \leq \ell$ such that $x \notin (\exists s_j.C_j)^{\mathcal{I}}$.

Consider the partial (B, A)-simulation chain

$$\begin{array}{ccccc} B & = & B_0 & \xrightarrow{r_1} & B_1 \\ & & Z\downarrow & & \\ A & = & A_0 & & \end{array}$$

where $B_1 := C_j$ and $r_1 := s_j$. The selection function S yields a defined concept A_1 such that $(B_1, A_1) \in Z$ and (A_0, r_1, A_1) is an edge in $\mathcal{G}_\mathcal{T}$. Since Y is a simulation with $(A_0, x) \in Y$, this implies the existence of an individual $x_1 \in \Delta^\mathcal{J}$ such that (x, r_1, x_1) is an edge in $\mathcal{G}_\mathcal{J}$ and $(A_1, x_1) \in Y$. Thus, we have the following situation:

$$\begin{array}{ccccc} B & = & B_0 & \xrightarrow{r_1} & B_1 \\ & & Z\downarrow & & Z\downarrow \\ A & = & A_0 & \xrightarrow{r_1} & A_1 \\ & & Y\downarrow & & Y\downarrow \\ & & x_0 & \xrightarrow{r_1} & x_1 \end{array}$$

where $x_0 := x$. By our assumption, $x_0 \in A_0^\mathcal{I} \setminus B_0^\mathcal{I}$.

Lemma 31 $x_1 \in A_1^\mathcal{I} \setminus B_1^\mathcal{I}$.

Proof. Since Y is a simulation satisfying condition (b) from above, Proposition 30 shows that $(A_1, x_1) \in Y$ implies $x_1 \in A_1^\mathcal{I}$.

Now, assume that $x_1 \in B_1^\mathcal{I} = C_j^\mathcal{I}$. Since (x, r_1, x_1) is an edge in $\mathcal{G}_\mathcal{J}$, we know that $(x, x_1) \in r_1^\mathcal{J} = r_1^\mathcal{I}$. But then $r_1 = s_j$ yields $x \in (\exists s_j.C_j)^\mathcal{I}$, which contradicts our choice of j. □

The lemma shows that we can now continue with x_1, B_1, A_1 in place of x_0, B_0, A_0, etc. This yields the following pair of simulation chains:

$$\begin{array}{ccccccccccc} B & = & B_0 & \xrightarrow{r_1} & B_1 & \xrightarrow{r_2} & B_2 & \xrightarrow{r_3} & B_3 & \xrightarrow{r_4} & \cdots \\ & & Z\downarrow & & Z\downarrow & & Z\downarrow & & Z\downarrow & & \\ A & = & A_0 & \xrightarrow{r_1} & A_1 & \xrightarrow{r_2} & A_2 & \xrightarrow{r_3} & A_3 & \xrightarrow{r_4} & \cdots \\ & & Y\downarrow & & Y\downarrow & & Y\downarrow & & Y\downarrow & & \\ & & x_0 & \xrightarrow{r_1} & x_1 & \xrightarrow{r_2} & x_2 & \xrightarrow{r_3} & x_3 & \xrightarrow{r_4} & \cdots \end{array}$$

where $x_n \in A_n^\mathcal{I} \setminus B_n^\mathcal{I}$ 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 $A_n = B_n$. This is an obvious contradiction to $x_n \in A_n^\mathcal{I} \setminus B_n^\mathcal{I}$. Thus, our assumption $x \in A^\mathcal{I} \setminus B^\mathcal{I}$ is refuted, which completes the proof of $(2) \to (1)$ of Theorem 29.

Proof of $(1) \to (2)$ of Theorem 29

Assume that $A \sqsubseteq_\mathcal{T} B$. We consider the graph $\mathcal{G}_\mathcal{T} = (V_\mathcal{T}, E_\mathcal{T}, L_\mathcal{T})$, and view it as an $\mathcal{EL}$-description graph describing a primitive interpretation. Let $\mathcal{J}$ denote the primitive interpretation such that $\mathcal{G}_\mathcal{T} = \mathcal{G}_\mathcal{J}$.

First, we will construct an interpretation $\mathcal{I}_0$ based on $\mathcal{J}$ such that $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_\mathcal{J} \mathcal{I}_0$. To this purpose, we construct an appropriate simulation $Y \colon \mathcal{G}_\mathcal{T} \overset{\rightharpoonup}{\sim} \mathcal{G}_\mathcal{T}$, and then define for all defined concepts C:

$$(*) \quad C^{\mathcal{I}_0} := \{C' \mid (C, C') \in Y\}.$$

We define $Y := \bigcup_{n \geq 0} Y_n$, where the relations Y_n are defined by induction on n: Y_0 is the identity on the nodes of $\mathcal{G}_\mathcal{T} = \mathcal{G}_\mathcal{J}$. If Y_n is already defined, then

$$\begin{aligned} Y_{n+1} := Y_n \cup \{(C, C') \mid\ & (1)\ L_\mathcal{T}(C) \subseteq L_\mathcal{T}(C'), \\ & (2)\ (C, r_1, C_1), \ldots, (C, r_\ell, C_\ell) \text{ are all the edges in } \mathcal{G}_\mathcal{T} \\ & \quad\ \text{with source } C, \text{ and} \\ & (3)\ \text{there are edges } (C', r_1, C_1'), \ldots, (C', r_\ell, C_\ell') \text{ in } \mathcal{G}_\mathcal{T} \\ & \quad\ \text{such that } (C_1, C_1') \in Y_n, \ldots, (C_\ell, C_\ell') \in Y_n \qquad \}. \end{aligned}$$

Lemma 32 *Y is a simulation.*

Proof. First, we show by induction on n that all the relations Y_n are simulations.

($n = 0$) The identity is obviously a simulation.

($n \to n + 1$) Assume that Y_n is a simulation. To show that Y_{n+1} is also a simulation, assume that $(C, C') \in Y_{n+1}$ and $(C, r, D) \in E_\mathcal{T}$. If $(C, C') \in Y_n$, then the assumption that Y_n is a simulation yields $L_\mathcal{T}(C) \subseteq L_\mathcal{T}(C')$ and the existence of a defined concept D' such that $(D, D') \in Y_n \subseteq Y_{n+1}$ and $(C', r, D') \in E_\mathcal{T}$.

Thus, assume that $(C, C') \in Y_{n+1} \setminus Y_n$. Then the definition of Y_{n+1} yields $L_\mathcal{T}(C) \subseteq L_\mathcal{T}(C')$ and the existence of a defined concept D' such that $(D, D') \in Y_n \subseteq Y_{n+1}$ and $(C', r, D') \in E_\mathcal{T}$.

Thus, we have shown that all Y_n are simulations. Now, let $(C, C') \in Y$ and $(C, r, D) \in E_\mathcal{T}$. Then there exists an $n \geq 0$ such that $(C, C') \in Y_n$, and thus the fact that Y_n is a simulation yields $L_\mathcal{T}(C) \subseteq L_\mathcal{T}(C')$ and the existence of a defined concept D' such that $(D, D') \in Y_n \subseteq Y$ and $(C', r, D') \in E_\mathcal{T}$. □

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

Lemma 33 $O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0) \preceq_\mathcal{J} \mathcal{I}_0$.

Proof. Let $\mathcal{I}_1 := O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0)$, and let C be a defined concept whose definition in $\mathcal{T}$ is

$$C \equiv P_1 \sqcap \ldots \sqcap P_m \sqcap \exists r_1.C_1 \sqcap \ldots \sqcap \exists r_\ell.C_\ell.$$

Assume that $C' \in C^{\mathcal{I}_1}$. We must show that this implies $C' \in C^{\mathcal{I}_0}$, i.e., that $(C, C') \in Y$.

First, note that $C' \in C^{\mathcal{I}_1} = C^{O_{\mathcal{T},\mathcal{J}}(\mathcal{I}_0)}$ implies that (i) $C' \in P_i^{\mathcal{I}_0} = P_i^\mathcal{J}$ for all $i = 1, \ldots, m$, and (ii) $C' \in (\exists r_j.C_j)^{\mathcal{I}_0}$ for all $j = 1, \ldots, \ell$.

Now, (i) shows that $L_{\mathcal{T}}(C) = \{P_1, \ldots, P_m\} \subseteq L_{\mathcal{T}}(C')$. In addition, (ii) implies that there are defined concepts $C'_1, \ldots, C'_\ell$ such that, for all $j = 1, \ldots, \ell$, we have $(C', C'_j) \in r^{\mathcal{I}_0} = r^{\mathcal{J}}$ (i.e., $(C', r_j, C'_j) \in E_{\mathcal{T}}$) and $C'_j \in C_j^{\mathcal{I}_0}$ (i.e., $(C_j, C'_j) \in Y$). The definition of Y implies that there is an n such that $(C_j, C'_j) \in Y_n$ holds for all $j = 1, \ldots, \ell$. But then $(C, C') \in Y_{n+1} \subseteq Y$. □

By Proposition 10, the lemma implies that $\mathcal{T}$ has an $\mathcal{I}_0$-gfp-model based on $\mathcal{J}$. Let $\mathcal{I}$ denote this model.

Lemma 34 $A \in A^{\mathcal{I}}$.

Proof. The simulation $Y\colon \mathcal{G}_{\mathcal{T}} \rightleftarrows\!\!\!\sim\ \mathcal{G}_{\mathcal{T}} = \mathcal{G}_{\mathcal{J}}$ satisfies

(a) $(A, A) \in Y$ (since $(A, A) \in Y_0 \subseteq Y$);

(b) if $(C, C') \in Y$ then $C' \in C^{\mathcal{I}_0}$ (by definition of $\mathcal{I}_0$).

Thus, Proposition 30 yields $A \in A^{\mathcal{I}}$. □

The lemma together with $A \sqsubseteq_{\mathcal{T}} B$ yields $A \in B^{\mathcal{I}}$. Thus, Proposition 30 implies that there exists a simulation $Z\colon \mathcal{G}_{\mathcal{T}} \rightleftarrows\!\!\!\sim\ \mathcal{G}_{\mathcal{T}} = \mathcal{G}_{\mathcal{J}}$ such that

(a) $(B, A) \in Z$; and

(b) if $(C, C') \in Z$ then $C' \in C^{\mathcal{I}_0}$.

Since $C' \in C^{\mathcal{I}_0}$ iff $(C, C') \in Y$, property (b) is equivalent to $Z \subseteq Y$. Thus, $(B, A) \in Z$ also yields $(B, A) \in Y$.

Lemma 35 *Y is a (B, A)-synchronized simulation satisfying $(B, A) \in Y$.*

Proof. It remain to show that Y is (B, A)-synchronized. To this purpose, we define an appropriate selection function S. Thus, consider the following partial (B, A)-simulation chain:

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

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

Case 1: $k = 0$. Then $B_{n-1} = A_{n-1}$ and the selection function S chooses $A_n := B_n$.

Case 2: $k > 0$. The minimality of k implies that $(B_{n-1}, A_{n-1}) \in Y_k \setminus Y_{k-1}$. 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 A_n such that $(A_{n-1}, r_n, A_n) \in E_\mathcal{T}$ and $(B_n, A_n) \in Y_{k-1}$. The selection function S chooses such an A_n.

It remains to be shown that the selection function S really satisfies the condition stated in Definition 28. Thus, consider the following S-selected (B, A)-simulation chain:

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

Let k_0 be minimal with $(B_0, A_0) \in Y_{k_0}$. If $k_0 = 0$, then we are done since then $A_0 = B_0$. Otherwise, $k_0 > 0$ and then we know that $(B_1, A_1) \in Y_{k_0-1}$. Thus, if k_1 is minimal with $(B_1, A_1) \in Y_{k_1}$, 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 $k_n = 0$, and thus $A_n = B_n$. □

This lemma finishes the proof of (1) → (2) of Theorem 29.

Deciding the existence of a synchronized simulation

It remains to be shown that property (2) of Theorem 29 can be decided in polynomial time. Thus, let $\mathcal{G}_\mathcal{T} = (V_\mathcal{T}, E_\mathcal{T}, L_\mathcal{T})$ be a finite $\mathcal{EL}$-description graph, and $(B, A) \in V_\mathcal{T} \times V_\mathcal{T}$ be a pair of nodes. We consider the simulation $Y\colon \mathcal{G}_\mathcal{T} \stackrel{\rightarrow}{\sim} \mathcal{G}_\mathcal{T}$ defined in the proof of (1) → (2) of Theorem 29. We have shown that Y is a (B, A)-synchronized simulation (see Lemma 32 and Lemma 35).

Proposition 36 *The following are equivalent:*

1. *There exists a (B, A)-synchronized simulation Z satisfying $(B, A) \in Z$.*
2. *$(B, A) \in Y$.*

Proof. (2) → (1) is trivial since we already know that Y is a (B, A)-synchronized simulation (by Lemma 32 and Lemma 35).

(1) → (2) Let S be the selection function that ensures that the simulation Z is (B, A)-synchronized. We use S to construct a tree t_S whose paths are basically initial segments of the S-selected (finite or infinite) (B, A)-simulation chains w.r.t. Z:

- The root of t_S is labeled with (B, A). By our assumption on Z, we have $(B, A) \in Z$.
- Let (B', C') be the label of a node κ already constructed. If $B' = C'$, then this node is a leaf of t_S.

- Let (B', C') be the label of a node κ already constructed, and $B' \neq C'$. By induction, we assume that the path leading to κ in the tree is of the form

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

where $A_1, \ldots, A_{n-1}$ have been selected using the selection function S. Now, let $(B', s_1, C_1), \ldots, (B', s_\ell, C_\ell)$ be all the edges in $\mathcal{G}_\mathcal{T}$ with source B'. For $i = 1, \ldots, \ell$ we consider the partial (B, A)-simulation chain

$$\begin{array}{ccccccccccc} B & = & B_0 & \stackrel{r_1}{\to} & B_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & B_{n-1} & \stackrel{s_i}{\to} & C_i \\ & & Z\downarrow & & Z\downarrow & & & & Z\downarrow & & \\ A & = & A_0 & \stackrel{r_1}{\to} & A_1 & \stackrel{r_2}{\to} & \cdots & \stackrel{r_{n-1}}{\to} & A_{n-1} & & \end{array}$$

  Let C_i' be the node selected by S. In particular, this means that $(C_i, C_i') \in Z$ and $(A_{n-1}, s_i, C_i') \in E_\mathcal{T}$. Now, κ obtains ℓ successor nodes in t_S, which are respectively labeled with $(C_1, C_1'), \ldots, (C_\ell, C_\ell')$. In particular, if $\ell = 0$, then κ is a leaf.

We claim that t_S is finite. In fact, an infinite path in t_S would yield an infinite (B, A)-simulation chain of the form depicted in Figure 3 such that $B_n \neq A_n$ for all $n \geq 0$. But this contradicts our assumption that S is the selection function that ensures that Z is (B, A)-synchronized. Thus, all paths in t_S are finite. Since t_S is also finitely branching, König's lemma shows that t_S is finite.

Next, we claim that, if a node in t_S is labeled with (B', A'), then $(B', A') \in Y$. Since (B, A) labels the root of t_S, this yields $(B, A) \in Y$, and we are done.

Let κ be a node in t_S labeled with (B', A'). We prove $(B', A') \in Y$ by induction on the maximal distance of κ to a leaf in t_S.

Induction base. If the maximal distance of κ to a leaf is 0, then κ is itself a leaf. There are two cases to consider:

1. The node κ has label (B', B'), i.e., $A' = B'$. But then $(B', A') \in Y_0 \subseteq Y$.
2. The node κ has label (B', A') with $A' \neq B'$, but B' has no outgoing edges in $\mathcal{G}_\mathcal{T}$. Since $(B', A') \in Z$, we know that $L_\mathcal{T}(B') \subseteq L_\mathcal{T}(A')$. Thus, the definition of Y_1 yields $(B', A') \in Y_1 \subseteq Y$.

Induction step. Assume that the maximal distance of κ to a leaf is not 0. In particular, this means that κ is not a leaf. Let $(C_1, C_1'), \ldots, (C_\ell, C_\ell')$ be the labels of all the successor nodes of κ in t_S. Consequently, there are roles $s_1, \ldots, s_\ell$ such that

1. $(B', s_1, C_1), \ldots (B', s_\ell, C_\ell)$ are all the edges in $\mathcal{G}_\mathcal{T}$ with source B';

2. $(A', s_1, C'_1), \ldots (A', s_\ell, C'_\ell)$ are edges in $\mathcal{G}_\mathcal{T}$;

3. $(C_1, C'_1), \ldots, (C_\ell, C'_\ell) \in Z$.

By induction, (3) implies $(C_1, C'_1), \ldots, (C_\ell, C'_\ell) \in Y$, and thus there is an n such that $(C_1, C'_1), \ldots, (C_\ell, C'_\ell) \in Y_n$. Since $(B', A') \in Z$ also yields $L_\mathcal{T}(B') \subseteq L_\mathcal{T}(A')$, (1) and (2) thus imply $(B', A') \in Y_{n+1} \subseteq Y$. □

Since Y can obviously be computed in time polynomial in the size of $\mathcal{G}_\mathcal{T}$, this proposition together with Theorem 29 yields the following corollary.

Corollary 37 *Subsumption w.r.t. descriptive semantics in $\mathcal{EL}$ can be decided in polynomial time.*

By using the techniques employed to decided Horn-SAT in linear time [8], it is not hard to show that the set Y can actually be computed in time quadratic in the size of $\mathcal{G}_\mathcal{T}$, and thus subsumption in $\mathcal{EL}$ w.r.t. descriptive semantics can be decided in quadratic time.

Example 38 Consider the graph $\mathcal{G}_\mathcal{T}$ depicted in Figure 5. The computation of Y proceeds as follows:

$$\begin{array}{rcl} Y_0 & = & \{(B,B),(E,E),(C,C),(D,D),(E_1,E_1),(E_2,E_2),(A,A)\}; \\ Y_1 & = & Y_0 \cup \{(E_1,E),(E_2,E),(C,E_1),(E_1,C),(D,E_2),(E_2,D),(C,E),(D,E)\}; \\ Y_2 & = & Y_1 \cup \{(A,B)\} = Y_3 = Y. \end{array}$$

Consequently, we have $B \sqsubseteq_\mathcal{T} A$, but not $A \sqsubseteq_\mathcal{T} B$.

An alternative way for showing the polynomiality result would be to reduce the existence of a (B, A)-synchronized simulation Z satisfying $(B, A) \in Z$ to the strategy problem for a certain two-player game with a positional winning condition. The existence of a winning strategy is in this case a polynomial time problem [10, 11]. Modulo some technicalities, the game graph is the subgraph of the Cartesian product of the graph $\mathcal{G}_\mathcal{T}$ with itself whose nodes satisfy condition (S1) of Definition 15. The winning positions for player two are the nodes (B', A') where either $B' = A'$ or B' has no successor nodes.

7 Conclusion

We have characterized subsumption in $\mathcal{EL}$ w.r.t. cyclic TBoxes for the three types of semantics introduced by Nebel [22]. In contrast to the case of $\mathcal{FL}_0$, where subsumption is no longer tractable if one allows for cyclic terminologies, these characterizations show that subsumption in $\mathcal{EL}$ w.r.t. cyclic TBoxes can be decided in polynomial time, independently of which semantics is used.

Our main motivation for considering cyclic terminologies in $\mathcal{EL}$ was the fact that the most specific concept of an ABox individual need not exist in $\mathcal{EL}$. An example is the simple cyclic ABox $\mathcal{A} := \{r(b,b)\}$, where b has no most specific concept, i.e., there is no least $\mathcal{EL}$-concept description D such that b is an instance of D w.r.t. $\mathcal{A}$ [17]. However, if one allows for cyclic TBoxes with gfp-semantics, then the defined concept B with $B \equiv \exists r.B$ is such a most specific concept. In a yet unpublished paper we have shown that the characterization of subsumption in $\mathcal{EL}$ w.r.t. gfp-semantics also yields an approach for computing the least common subsumer in $\mathcal{EL}$ w.r.t. gfp-semantics. In addition, we have extended the characterization of subsumption in $\mathcal{EL}$ w.r.t. gfp-semantics to the instance problem, and have shown how this can be used to compute the most specific concept.

Regarding related work, simulations and bisimulations play an important rôle in modal logics (and thus also in description logics). However, until now they have mostly been considered for modal logics that are closed under all the Boolean operators, and they have usually not been employed for reasoning in the logic. A notable exception is [13], where bisimulation characterizations are given for sub-Boolean DLs. However, these characterizations are used to give a formal account of the expressive power of these logics. They are not employed for reasoning purposes.

The DL $\mathcal{EL}$ with cyclic terminologies interpreted with one of the three semantics considered in this report yields a small fragment of the modal mu-calculus. For these fragments, the subsumption problem (i.e., the question whether an implication between two formulae is valid) can still be decided in polynomial time. The relationship of this result to possibly existing complexity results for fragments of the modal mu-calculus still needs to be explored. At the moment, we are not aware of any other results for such small fragments of the modal mu-calculus.

References

[1] 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.

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

[3] 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.

[4] 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.

[5] William W. Cohen, Alex Borgida, and Haym Hirsh. Computing least common subsumers in description logics. In William Swartout, editor, *Proc. of the 10th Nat. Conf. on Artificial Intelligence (AAAI'92)*, pages 754–760. AAAI Press/The MIT Press, 1992.

[6] William W. Cohen and Haym Hirsh. Learnability of description logics with equality constraints. *Machine Learning*, 17(2/3):169–200, 1994.

[7] Giuseppe De Giacomo and Maurizio Lenzerini. Concept language with number restrictions and fixpoints, and its relationship with μ-calculus. In *Proc. of the 11th Eur. Conf. on Artificial Intelligence (ECAI'94)*, pages 411–415, 1994.

[8] William F. Dowling and Jean Gallier. Linear-time algorithms for testing the satisfiability of propositional horn formulae. *Journal of Logic Programmming*, 1(3):267–284, 1984.

[9] Michael R. Garey and David S. Johnson. *Computers and Intractability — A guide to NP-completeness*. W. H. Freeman and Company, San Francisco (CA, USA), 1979.

[10] Erich Grädel. Model checking games. *Electronic Notes in Theoretical Computer Science*, 2002. To appear.

[11] Raymond Greenlaw, H.J. Hoover, and Walter L. Ruzzo. *Limits to Parallel Computation: P-Completeness Theory*. Oxford University Press, 1995.

[12] 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.

[13] Natasha Kurtonina and Maarten de Rijke. Expressiveness of concept expressions in first-order description. *Artificial Intelligence*, 107(2):303–333, 1999.

[14] Ralf Küsters. Characterizing the semantics of terminological cycles in $\mathcal{ALN}$ using finite automata. In *Proc. of the 6th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR'98)*, pages 499–510, 1998.

[15] Ralf Küsters. *Non-standard Inferences in Description Logics*, volume 2100 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 2001.

[16] Ralf Küsters and Alex Borgida. What's in an attribute? Consequences for the least common subsumer. *J. of Artificial Intelligence Research*, 14:167–203, 2001.

[17] Ralf Küsters and Ralf Molitor. Approximating most specific concepts in description logics with existential restrictions. In Franz Baader, Gerd Brewka, and Thomas Eiter, editors, *Proceedings of the Joint German/Austrian Conference on Artificial Intelligence (KI 2001)*, volume 2174 of *Lecture Notes in Artificial Intelligence*, pages 33–47, Vienna, Austria, 2001. Springer-Verlag.

[18] Ralf Küsters and Ralf Molitor. Computing least common subsumers in $\mathcal{ALEN}$. In *Proc. of the 17th Int. Joint Conf. on Artificial Intelligence (IJCAI 2001)*, pages 219–224, 2001.

[19] John W. Lloyd. *Foundations of Logic Programming (Second, Extended Edition)*. Springer-Verlag, Berlin, Heidelberg, 1987.

[20] Bernhard Nebel. *Reasoning and Revision in Hybrid Representation Systems*, volume 422 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag, 1990.

[21] Bernhard Nebel. Terminological reasoning is inherently intractable. *Artificial Intelligence*, 43:235–249, 1990.

[22] 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.

[23] Klaus Schild. A correspondence theory for terminological logics: Preliminary report. In *Proc. of the 12th Int. Joint Conf. on Artificial Intelligence (IJCAI'91)*, pages 466–471, 1991.

[24] Klaus Schild. Terminological cycles and the propositional μ-calculus. In J. Doyle, E. Sandewall, and P. Torasso, editors, *Proc. of the 4th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'94)*, pages 509–520, Bonn (Germany), 1994. Morgan Kaufmann, Los Altos.

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