Using Automata Theory for Characterizing the Semantics of Terminological Cycles

Franz Baader

Lehr- und Forschungsgebiet Theoretische Informatik
RWTH Aachen, Ahornstraße 55, 52074 Aachen, Germany
E-mail: `baader@informatik.rwth-aachen.de`

In most of the implemented terminological knowledge representation systems it is not possible to state recursive concept definitions, so-called terminological cycles. One reason is that it is not clear what kind of semantics to use for such cyles. In addition, the inference algorithms used in such systems may go astray in the presence of terminological cycles. In this paper we consider terminological cycles in a very small terminological representation language. For this language, the effect of the three types of semantics introduced by B. Nebel can completely be described with the help of finite automata. These descriptions provide for a rather intuitive understanding of terminologies with recursive definitions, and they give an insight into the essential features of the respective semantics. In addition, one obtains algorithms and complexity results for the subsumption problem and for related inference tasks. The results of this paper may help to decide what kind of semantics is most appropriate for cyclic definitions, depending on the representation task

Subject classification: AMS(MOS) 68T30
Keywords: terminological knowledge representation, cyclic definitions, automata theory

1 – Introduction

Terminological representation systems can be used to represent the taxonomic and conceptual knowledge of a problem domain in a structured way. To describe this kind of knowledge, one starts with atomic concepts (unary predicates) and roles (binary predicates), and constructs more complex concept descriptions (called concept terms in the following) using the operations provided by the concept language of the particular formalism. For example, if Human and Male are atomic concepts and child is a role, then the concept terms Human ⊓ Male and ∀child: Male describe, respectively, the set of all individuals that are both human and male, and the set of all individuals that have male children only. The concept forming construct used in the first term is called concept *conjunction* and the one in the second term is called *value restriction*. The semantics of such expressions can be described in a model-theoretic way (see Section 3 below), or by a translation into first-order predicate logic. The concept terms of our example correspond to formulas with one free variable: Human(x) ∧ Male(x) and ∀y: (child(x,y) → Male(y)). For a given interpretation, the concept represented by a concept term consists of the individuals (elements of the domain of the interpretation) that satisfy the corresponding formula when substituted for its free variable.

The *terminology* (T-box) of a terminological representation system consists of concept definitions that assign names to complex terms. For example, one can introduce the name "man" for the concept of all male humans via the definition Man = Human ⊓ Male, which has the obvious semantics ∀x: (Man(x) ⇔ Human(x) ∧ Male(x)). This semantics will be called descriptive semantics in the following. As long as there are no cyclic dependencies in the concept definitions of a terminology, descriptive semantics is clearly appropriate. In this case, definitions merely introduce abbreviations (macros) for complex terms, and the defined names occurring in a concept term can simply be replaced via successive macro-expansions (see, e.g., [30], Section 3.2.5). If there are cyclic dependencies in the definitions of a T-box then such an expansion process need not terminate, which is the reason why the inference methods of most of the existing terminological systems (e.g., KRYPTON [7], NIKL [21], LOOM [26], CLASSIC [33], or KRIS [5]) cannot handle terminological cycles. Another problem with terminological cycles is that descriptive semantics, as introduced above, need no longer be appropriate, and it is not obvious what type of semantics should be employed instead.

Cyclic concept definitions may be very useful and intuitive, though. For example, they can be used to express value restrictions with respect to the transitive closure of roles. Assume that we have the role child, and that we want to describe the concept of all men having only male offspring, for short Momo. Obviously, we cannot just introduce a new atomic role offspring because then there would be no connection between the two roles child and offspring. But the intended meaning of offspring is that it is the transitive closure of the role child.

Here a cyclic definition of Momo seems to be quite natural: a man having only male offspring is himself a man, and all his children are men having only male offspring. This can be expressed by the concept definition

$$\mathsf{Momo} \quad = \quad \mathsf{Man} \sqcap \forall \mathsf{child}{:}\, \mathsf{Momo},$$

provided that an appropriate fixed-point semantics is used. The results of this paper will show that greatest fixed-point semantics is the semantics that captures our intuition here. For similar reasons, recursive axioms with fixed-point semantics are considered in data base research (see e.g., [1,20,41,28,43,42]). In [1], Aho and Ullman have shown that the transitive closure of relations cannot be expressed in the relational calculus, which is a standard relational query language. They propose to add cyclic definitions that are interpreted by least fixed-point semantics. This was also the starting point for an extensive study of fixed-point extensions of first-order logic (see e.g., [16,17]).

As another example that illustrates the possible use of cyclic definitions in terminologies, assume that we want to define the concept Dag,[1] which should consist of all nodes belonging to a finite directed acyclic graph whose connections are given by a relation "arc." This concept can be described using the following cyclic definition:

$$\mathsf{Dag} \quad = \quad \mathsf{Node} \sqcap \forall \mathsf{arc}{:}\, \mathsf{Dag}.$$

As for the definition of Momo, a fixed-point semantics is more appropriate than descriptive semantics here. The results presented in this paper will show that, unlike the definition of Momo, which should be interpreted with greatest fixed-point semantics, the definition of Dag requires least fixed-point semantics.

It is, of course, not enough to have a system that just stores concept definitions. The system must also be able to reason about this knowledge. An important inference service of a terminological system is "classification." The classifier computes all *subsumption relationships* between concepts, i.e., all subconcept-superconcept relationships induced by the concept definitions. The choice of the semantics strongly influences which subsumption relationships hold in a terminology. In addition, different semantics may require different algorithmic method for determining subsumption relationships, and they might be responsible for different behaviour with respect to decidability and complexity of the subsumption problem. As mentioned above, the subsumption algorithms implemented in most of the existing terminological systems cannot handle cyclic definitions.

The first thorough investigation of cycles in terminological knowledge

[1] This example is taken from [15].

representation languages can be found in [29,30,32], where B. Nebel considers three different types of semantics for cyclic definitions in his language $\mathcal{NTF}$, namely, least fixed-point semantics, greatest fixed-point semantics, and descriptive semantics. Due to the fact that this language is relatively expressive this investigation does not provide us with a deep insight into the meaning of cycles with respect to these three types of semantics. For the two fixed-point semantics, Nebel explains his point just with a few examples. The meaning of descriptive semantics—which, in Nebel's opinion, comes "closest to the intuitive understanding of terminological cycles" ([30], p. 120)—is treated more thoroughly. But even in this case the results are not quite satisfactory. For example, decidability of the subsumption problem is proved by an argument that cannot be used to derive a practical algorithm, and which does not give insight into the reason why one concept defined by some cyclic definition subsumes another one. Roughly speaking, the argument says that it is sufficient to consider only finite interpretations to determine subsumption relationships. An interesting observation concerning descriptive semantics in Nebel's paper is that structurally identical definitions need not lead to semantically equivalent concepts (i.e., concepts that mutually subsume each other). For example, assume that in addition to the definition of Momo from above, we also define a concept Mnfo (for man without female offspring):

$$\mathsf{Mnfo} \quad = \quad \mathsf{Man} \sqcap \forall \mathsf{child}: \mathsf{Mnfo}.$$

Beside the acronym chosen for the concept to be defined, the definitions of Momo and Mnfo are identical. For (greatest or least) fixed-point semantics, this is reflected by the fact that the two concepts are equivalent. For descriptive semantics, no such equivalence holds since names of defined concepts are important as well. The characterization of subsumption with respect to descriptive semantics given in the present paper will clarify this dependency on names.

Before we can determine what kind of semantics is most appropriate for terminological cycles, we need a better understanding of their intended meaning. The same argument applies to the decision whether to allow or disallow cycles. Even if cycles are prohibited, this should not be done just because one does not know what they mean and how they can be handled.

In this paper, we shall consider terminological cycles in a very small terminological representation language. It provides only concept conjunction and value-restrictions as constructs for building concept terms. For this language, the effect of the three types of semantics mentioned above can completely be described with the help of finite automata. These descriptions provide a rather intuitive understanding of terminologies with cyclic definitions, and they give insight into the essential features of the respective semantics. In addition, the subsumption problem for each type of semantics can be reduced to a (more or less) well-known decision problem for finite automata. Hence, existing algorithms can be used to decide subsumption, and known complexity results yield the complexity of the subsumption problem.

In the next section we shall recall some definitions and results concerning ordinals, fixed-points and finite automata that will be used in subsequent sections. Syntax and (descriptive) semantics of our small terminological language $\mathcal{FL}_0$ is introduced in Section 3. In Section 4, alternative types of semantics—namely least and greatest fixed-point semantics—are considered, which may be more appropriate in the presence of terminological cycles. We shall see that, from a constructive point of view, greatest fixed-point semantics shows a better behaviour than least fixed-point semantics since greatest fixed-point models can be obtained by a single limit process. In Section 5, the three types of semantics are characterized with the help of finite automata. The characterization of the greatest fixed-point semantics is easy and intuitively clear. Subsumption with respect to greatest fixed-point semantics, and—after some modifications of the automaton—also with respect to least fixed-point semantics can be reduced to inclusion of regular languages. For descriptive semantics, we have to consider inclusion of certain languages of infinite words that are defined by the automaton. Fortunately, these languages have already been investigated in the context of monadic second-order logic (see [8]). In Section 6, we shall see how the inclusion problem for these languages can be solved. This yields a subsumption algorithm for descriptive semantics. Extensions of the results for gfp-semantics are considered in Section 7. In the first subsection we shall consider cycles in the larger language $\mathcal{FL}^-$ introduced in [23]. The second subsection contains results about hybrid inferences. Finally, we shall point out related work on cyclic definitions in terminologies.

2 – Formal Preliminaries

For the readers convenience, we shall recall some definitions and results concerning ordinals, fixed-points and finite automata. Those familiar with these topics may skip this section and come back to it if necessary.

In the introduction we have mentioned the “transitive closure” of a binary relation as a motivation for cyclic definitions. This notion can be formally defined as follows: Let R be a binary relation on a set D, i.e., $R \subseteq D \times D$. We define $R^0 := \{(d,d);\ d \in D\}$ and, for $n \geq 0$, $R^{n+1} := R \circ R^n$ where “$\circ$” denotes composition of binary relations. The *transitive closure* of R is the relation

$\cup_{n\geq 1}R^n$ and the *reflexive-transitive closure* is $\cup_{n\geq 0}R^n$.

2.1 – Ordinals

A more detailed account of the order-theoretic approach to ordinals used below can be found in [35]. A set-theoretic definition of ordinals is, for example, given in [18]. Some elementary properties of ordinals are also stated in [24], p.28–29.

A partial ordering ≤ on a set D is a *well-ordering* iff it is *linear* (i.e., for all a, b in D we have a ≤ b or b ≤ a) and *well-founded* (i.e., there are no infinite strictly decreasing chains $a_0 > a_1 > a_2 > ...$). *Ordinals* can be defined as the order types of well-ordered sets. There are *finite ordinals* such as 2, 6, 17. For example, 6 is the order type of the set {0, 1, 2, 3, 4, 5} with the usual ordering on non-negative integers. The first infinite ordinal is ω, which is the order type of the non-negative integers {0, 1, 2, ...}. Ordinals can be ordered as follows: $\alpha \leq \beta$ iff α is isomorphic to an initial segment of β. For example, 2 < 6 and the finite ordinals are exactly the ordinals that are smaller than ω. This ordering on ordinals is well-founded and linear. Hence any set of ordinals has a least element and a least upper bound.

If α is an ordinal then the *successor* α+1 of α is the least ordinal greater than α. An ordinal that is a successor of another ordinal is called *successor ordinal*. The other ordinals are called *limit ordinals*. For example, ω is a limit ordinal, and 6 is a successor ordinal because 6 = 5+1 is the successor of 5. The successor ω + 1 of ω is the order typ of {0, 1, 2, ...} ∪ {∞} where {0, 1, 2, ...} is ordered as usual and all elements of {0, 1, 2, ...} are smaller than ∞. A limit ordinal α can be obtained as the least upper bound of all smaller ordinals, i.e., $\alpha = \mathrm{lub}(\{\beta; \beta < \alpha\})$.

Properties for ordinals can be proved by *transfinite induction*. Let P be a property of ordinals. Assume that **(1)** P(0) holds; **(2)** if P(α) holds then P(α+1) holds; and **(3)** if λ is a limit ordinal and P(α) holds for all α < λ then P(λ) holds. Then P(β) holds for all ordinals β.

2.2 – Fixed-Points

The definitions and results mentioned in this subsection can be found in [24], Chapter 1, §5 and [38], Chapter 6. An account of the history of various fixed-point theorems is given in [22].

Let D be a partially ordered set (*poset)*. The poset D is a *complete lattice* if all subsets C of D have a least upper bound lub(C) in D. In this case, any subset C has also a greatest lower bound glb(C) = lub({d ∈ D; d is a lower bound of C}), and D has a least element bottom = lub(∅) and a greatest element top = lub(D).

The following example will be reconsidered in Section 4.

Example 1
Consider the n-fold cartesian product $D = 2^S \times ... \times 2^S$, where 2^S denotes the set of all subsets of a set S. The set D is ordered componentwise by inclusion: $(A_1,\ldots,A_n) \subseteq (B_1,\ldots,B_n)$ iff $A_1 \subseteq B_1$, …, and $A_n \subseteq B_n$. Greatest lower bounds and least upper bounds with respect to this ordering are obtained by componentwise set intersection and set union, top = (S,…,S), and bottom = (Ø,…,Ø).

Let D be a poset and let T: D → D be a mapping. Then T is *monotonic* iff for all a, b in D, a ≤ b implies T(a) ≤ T(b). A *fixed-point* of T is an element f ∈ D such that T(f) = f holds. If D is a complete lattice, then any monotonic mapping T: D → D has a fixed-point. More precisely, T has a *least fixed-point* lfp(T) and a *greatest fixed-point* gfp(T), and possibly other fixed-points, which lie between the least and the greatest fixed point. The least and the greatest fixed-point can be characterized in terms of ordinal powers of T. The *ordinal powers* $T{\uparrow}^{\alpha}$ and $T{\downarrow}^{\alpha}$ are inductively defined as follows:

(1) $T{\uparrow}^{0}$:= bottom and $T{\downarrow}^{0}$:= top;
(2) $T{\uparrow}^{\alpha+1} := T(T{\uparrow}^{\alpha})$ and $T{\downarrow}^{\alpha+1} := T(T{\downarrow}^{\alpha})$;
(3) If α is a limit ordinal then $T{\uparrow}^{\alpha} := lub(\{T{\uparrow}^{\beta};\ \beta < \alpha\})$ and $T{\downarrow}^{\alpha} := glb(\{T{\downarrow}^{\beta};\ \beta < \alpha\})$.

Theorem 2 (least and greatest fixed-points)
Let D be a complete lattice, and let T: D → D be a monotonic mapping. Then, for any ordinal α, $T{\uparrow}^{\alpha} \leq lfp(T)$ and $T{\downarrow}^{\alpha} \geq gfp(T)$. Furthermore, there exist ordinals β, γ such that $T{\uparrow}^{\beta} = lfp(T)$ and $T{\downarrow}^{\gamma} = gfp(T)$.

The ordinals β, γ may be greater than ω, but there are sufficient conditions under which they are less or equal ω. Let D be a complete lattice, and let T: D → D be a mapping. Then T is *upward ω-continuous* (resp. *downward ω-continuous*) iff for any increasing chain $d_0 \leq d_1 \leq d_2 \leq \ldots$ (resp. decreasing chain $d_0 \geq d_1 \geq d_2 \geq \ldots$) we have $T(lub(\{d_i;\ i \geq 0\})) = lub(\{T(d_i);\ i \geq 0\})$ (resp. $T(glb(\{d_i;\ i \geq 0\})) = glb(\{T(d_i);\ i \geq 0\})$). It is easy to see that any upward or downward ω-continuous mapping is also monotonic.

Theorem 3 (fixed-points of continuous mappings)
Let D be a complete lattice, and let T: D → D be an upward ω-continuous (resp. downward ω-continuous) mapping. Then $lfp(T) = T{\uparrow}^{\omega} = lub(\{T^n(bottom);\ n \geq 0\})$ (resp. $gfp(T) = T{\downarrow}^{\omega} = glb(\{T^n(top);\ n \geq 0\})$).

The notation “n ≥ 0” is used as an abbreviation for “0 ≤ n < ω”. Here and in the following, we use the convention that n, i, k range only over finite ordinals.

In Section 5.3 we shall need a slightly generalized version of Theorem 3 for downward ω-continuous mappings.

Corollary 4
Let D be a complete lattice, and let T: D → D be a downward ω-continuous mapping. Let d be an element of D such that d ≥ T(d). Then d-gfp(T) := glb({T^n(d); n ≥ 0}) is a fixed-point of T. More precisely, d-gfp(T) is the greatest fixed-point of T that is less or equal d.

Proof
Since T is downward ω-continuous and thus monotonic, d ≥ T(d) yields d ≥ T(d) ≥ T^2(d) ≥ T^3(d) ≥ … . Hence T(glb({T^n(d); n ≥ 0})) = glb({T^{n+1}(d); n ≥ 0})) = glb({T^n(d); n ≥ 0}) since d = T^0(d) ≥ T(d) by assumption. This shows that d-gfp(T) is a fixed-point, and obviously, d ≥ d-gfp(T). If f is a fixed-point with d ≥ f then T(d) ≥ T(f) = f, since T is monotonic, and f is a fixed-point. Iterating this argument we obtain T^n(d) ≥ f for all n ≥ 0, and hence glb({T^n(d); n ≥ 0}) ≥ f. ❑

2.3 – Automata and Words

The notions introduced below can, for example, be found in [27,19,12], possibly with a slightly different terminology.

Let Σ be a finite alphabet. The set of all (finite) words over Σ will be denoted by Σ^* and the empty word by ε. A word W = $\sigma_0 \ldots \sigma_{n-1}$ over Σ of length n can be seen as a mapping W of the finite ordinal n = {0, …, n–1} into Σ, namely, W(i) := σ_i for i = 0, …, n–1. This motivates the following definition of infinite words. An *infinite word* W is a mapping of the ordinal ω into Σ. The set of all infinite words over Σ will be denoted by Σ^ω. A given infinite word W: ω → Σ will sometimes be written as an infinite sequence W(0)W(1)W(2)... .

A *semi-automaton with word transitions* is a triple $\mathcal{A}$ = (Σ,Q,E), which consists of a finite alphabet Σ, a finite set of states Q, and a finite set of transitions (or edges) E ⊆ Q × Σ^* × Q. Thus, a transition connects two states of Q, and it is labeled by a finite word over Σ.

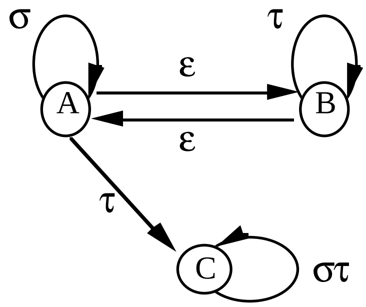
If all transitions are labeled by words of length one, then $\mathcal{A}$ is called *semi-automaton with letter transitions*. In situations where the distinction between word transitions and letter transitions is irrelevant, we shall simply use the term semi-automaton. Unlike the usual finite automata, semi-automata have no fixed set of initial and final states. This will be convenient later on since we must consider the same semi-automaton with varying initial and terminal states.

Example 5 (a semi-automaton with word transitions)

$\Sigma = \{ \sigma, \tau \}$

$Q = \{ A, B, C \}$

$E = \{ (A,\sigma,A), (A,\varepsilon,B),$
$(A,\tau,C), (B,\tau,B),$
$(C,\sigma\tau,C), (B,\varepsilon,A)\}$



Let $\mathcal{A}$ be a semi-automaton with word transitions, and let p, q be states of $\mathcal{A}$. A *finite path* from p to q in $\mathcal{A}$ is a sequence $p_0, U_1, p_1, U_2, p_2, \ldots, U_n, p_n$, where $p = p_0$, $q = p_n$, and for each i, $1 \leq i \leq n$, (p_{i-1},U_i,p_i) is a transition of $\mathcal{A}$. This path has the finite word $U_1U_2\ldots U_n$ as label. As a special case, the empty path p from p to p has the empty word ε as label. In the example, A, σ, A, ε, B, ε, A, τ, C, $\sigma\tau$, C is a finite path from A to C with label $\sigma\tau\sigma\tau$. Obviously, a non-empty path (i.e., a path where $n \geq 1$) may also have the empty word as label. An *infinite path* starting with p is an infinite sequence $p_0, U_1, p_1, U_2, p_2, \ldots$, where $p = p_0$ and for each $i \geq 1$, (p_{i-1},U_i,p_i) is a transition of $\mathcal{A}$. The label $U_1U_2U_3\ldots$ of this infinite path may be a finite or an infinite word. In the example, the infinite path A, σ, A, ε, B, ε, A, ε, B, ε, A, ε, B, ε, A, … has the finite word σ as label, and the infinite path A, τ, C, $\sigma\tau$, C, $\sigma\tau$, C, … has the infinite word $\tau\sigma\tau\sigma\tau\ldots$ as label. We shall sometimes omit some of the insignificant intermediate states in the description of a path. For example, assume that we are interested in those infinite paths starting with p where the state q is reached infinitely often. Such a path may be written as p, W_0, q, W_1, q, W_2, … where W_0 is the label of a path from p to q and the W_i for $i \geq 1$ are labels of non-empty paths from q to q.

For two states p, q of the semi-automaton with word transitions $\mathcal{A}$, let $L_{\mathcal{A}}(p,q)$ denote the set of all finite words that are labels of paths fom p to q. If it is clear from the context, we shall omit the index $\mathcal{A}$. In Example 5, $L(A,B) = (\sigma\cup\tau)^* = \Sigma^*$ and $L(A,C) = (\sigma\cup\tau)^*\tau(\sigma\tau)^* = \{W\tau(\sigma\tau)^m;\ W \in \Sigma^*, m \geq 0\}$. Obviously, the languages L(p,q) are regular, and on the other hand, any regular language can be obtained this way. If the regular language $L = L(\mathcal{A})$ is accepted by a finite automaton $\mathcal{A}$ with initial state q_0 and set of terminal state Q_{fin}, i.e., $L = \cup_{t\in Q_{fin}} L(q_0,t)$, then we can add a new state q_{fin} to $\mathcal{A}$, and transitions (t,ε,q_{fin}) for all $t \in Q_{fin}$. Then $L = L(q_0,q_{fin})$. The case of more than one initial state can be treated analogously.

With respect to the accepted regular languages, semi-automata with word transitions are not more expressive than semi-automata with letter transitions. In fact, any semi-automaton with word transitions $\mathcal{A}$ can be transformed (in polynomial time) into a semi-automaton with letter transitions $\mathcal{B}$ such that for all states p_1, p_2 in $\mathcal{A}$ there exist states q_1, q_2 in $\mathcal{B}$ with $L_{\mathcal{A}}(p_1,p_2) = L_{\mathcal{B}}(q_1,q_2)$ (see [27] or [19]). Words of length greater than one can easily be eliminated by introducing intermediate states. In the example, we could introduce a new state C' and replace the transition (C,$\sigma\tau$,C) by the two transitions (C,σ,C') and (C', τ,C). The elimination of ε-transitions is more difficult (see [19], p. 26). In the example, we could simply join the states A and B to a new state AB with the transitions (AB,σ,AB), (AB,τ,AB), (AB,τ,C).

For a state p of the semi-automaton with word transitions $\mathcal{A}$, let $U_{\mathcal{A}}(p)$ denote the set of all words that are labels of infinite paths starting with p. As for L, we shall often omit the index $\mathcal{A}$. Note that U(p) may also contain finite words that are labels of infinite paths starting with p. In the example, U(A) = U(B) = $\Sigma^* \cup \Sigma^\omega$ and U(C) is the singleton $\{\sigma\tau\sigma\tau\sigma\tau\ldots\}$.

3 – A Small Terminological Representation Language

The language considered in this paper will be called $\mathcal{FL}_0$. It has only two constructs for building complex concept descriptions: concept conjunction and value-restriction.

Definition 6 (concept terms and terminologies)
*Let **C** be a set of concept names and **R** be a set of role names. The set of* concept terms *of $\mathcal{FL}_0$ is inductively defined. As a starting point of the induction,*

*(1) any element of **C** is a concept term.* *(atomic terms)*

Now let C and D be concept terms already defined, and let R be a role name.

(2) Then $C \sqcap D$ is a concept term. *(concept conjunction)*

(3) Then $\forall R{:}C$ is a concept term. *(value-restriction)*

Let A be a concept name and let D be a concept term. Then A = D is a terminological axiom. A terminology *(T-box) is a finite set of terminological axioms with the additional restriction that no concept name may appear more than once as a left hand side of a definition.*

A T-box contains two different kinds of concept names. *Defined concepts* occur on the left hand side of a terminological axiom. The other concepts are called *primitive concepts*.[2] The following is an example of a T-box in this formalism: Let Man, Human, Male, Mos (for "*m*an that has *o*nly *s*ons"), and Momo (for "*m*an that has *o*nly *m*ale *o*ffspring") be concept names and let child be a role name. The T-box consists of the following axioms:

[2]For our language, roles are always primitive since we do not have role definitions.

$$\begin{aligned} \text{Man} &= \text{Human} \sqcap \text{Male} \\ \text{Mos} &= \text{Man} \sqcap \forall \text{child: Man} \\ \text{Momo} &= \text{Man} \sqcap \forall \text{child: Momo.} \end{aligned}$$

This means that a man is human and male. A man that has only sons is a man such that all his children are male humans. Male and Human are primitive concepts while Man and Mos are defined concepts. As mentioned in the introduction, one cannot just introduce a new role name offspring to define the concept Momo. This is so because there would be no connection between the two primitive roles child and offspring, whereas the intended meaning of offspring is that it is the transitive closure of child. Thus we have used a cyclic definition, which intuitively means: A man that has only male offspring is himself a man, and all his children are men having only male offspring. This is a very simple cyclic definition. In general, cycles in terminologies are defined as follows.

Definition 7 (terminological cycles)
Let A, B be concept names and let T be a T-box. We say that A directly uses B in T iff B appears on the right hand side of the definition of A. Let "uses" denote the transitive closure of the relation "directly uses." Then T contains a terminological cycle *iff there exists a concept name A in T such that A uses A.*

The next definition gives a model-theoretic semantics for the language introduced in Definition 6.

Definition 8 (interpretations and models)
An interpretation *I consists of a set dom(I), the domain of the interpretation, and an interpretation function, which associates with each concept name A a subset* A^I *of dom(I) and with each role name R a binary relation* R^I *on dom(I), i.e., a subset of dom(I) × dom(I). The sets* A^I, R^I *are called extensions of A, R with respect to I.*
The interpretation function—which gives an interpretation for atomic terms—can be extended to arbitrary terms as follows: Let C, D be concept terms and R be a role name. Assume that C^I *and* D^I *are already defined. Then*

$$\begin{aligned} (C \sqcap D)^I &:= C^I \cap D^I, \\ (\forall R{:}C)^I &:= \{x \in dom(I);\ \text{for all } y\text{: } (x,y) \in R^I \text{ implies } y \in C^I\}. \end{aligned}$$

An interpretation I is a model *of the T-box T iff it satisfies*

$$A^I = D^I \text{ for all terminological axioms } A = D \text{ in } T.$$

As mentioned in the introduction, an important inference service terminological systems provide their users with is computing the subsumption hierarchy, i.e., computing all subconcept-superconcept relationships induced by the definitions in the T-box.

Definition 9
Let T be a terminology and let A, B be concept names. We define

$$A \sqsubseteq_T B \text{ iff } A^I \subseteq B^I \text{ for all models } I \text{ of } T.$$

In this case we say that B subsumes *A in T.*

The semantics we have just defined will in the following be called *descriptive semantics*. It is not restricted to non-cyclic terminologies. For cyclic terminologies this kind of semantics may, however, seem unsatisfactory. One might think that the extension of a defined concept should completely be determined by the extensions of the primitive concepts and roles. Otherwise, the use of the term “concept definition” is not really justified. Non-cyclic terminologies satisfy this requirement.

More precisely, let T be a T-box containing the defined concepts $C_1, \ldots, C_n$, the primitive concepts $P_1, ..., P_m$, and the roles $R_1, ..., R_k$. A *primitive interpretation* J consists of a set dom(J), the domain of the primitive interpretation, and extensions $P_1^J, ..., P_m^J, R_1^J, ..., R_k^J$ of the primitive concepts and roles. An interpretation I of T *extends* the primitive interpretation J iff dom(I) = dom(J), $P_1^I = P_1^J, ..., P_m^I = P_m^J$ and $R_1^I = R_1^J, ..., R_k^I = R_k^J$. Such an extension I of J can be described by the n-tuple $(C_1^I,\ldots,C_n^I) \in (2^{dom(J)})^n$, where $2^{dom(J)}$ denotes the set of all subsets of dom(J).

On the other hand, any primitive interpretation J together with an n-tuple $\underline{A} \in (2^{dom(J)})^n$ yields an interpretation I for T. Any defined concept in T corresponds to a component of the tuple $\underline{A}$. If the defined concept B corresponds to the i-component of $\underline{A}$, i.e., $B^I = (\underline{A})_i$, we shall say that index(B) = i. Of course, we are mostly interested in extensions of J that are models of T. If T does not contain cycles, then any primitive interpretation can uniquely be extended to a model of T (see, e.g., [30], Theorem 3.2). If T contains cycles, a given primitive interpretation may have different extensions to models of T.

Example 10
Let Momo and Man be concept names, and child be a role name. The terminology T consists of the single axiom Momo = Man ⊓ ∀child: Momo.
We consider the following primitive interpretation:

dom(J) := {Charles1, Charles2, Charles3, ...} ∪ {James1, ..., JamesLast}
Man^J := dom(J), and
$child^J$:= {(Charlesi,Charlesi+1); i ≥ 1} ∪ {(Jamesi,Jamesi+1; 1 ≤ i < Last}.

This means that the Charles dynasty does not die out, whereas there is a last member of the James dynasty. It is easy to see that this primitive interpretation has two different extensions to models of T. The defined concept Momo may either be interpreted as {James1, ..., JamesLast} or as dom(J). Note that individuals without children (i.e., without $child^J$-successors) are in the extension of the term ∀child: Momo, no matter how Momo is interpreted.

The example also demonstrates that, with respect to descriptive semantics as defined above, the definition Momo = Man ⊓ ∀child: Momo does not express the value-restriction Momo = Man ⊓ ∀offspring: Man for the transitive closure offspring of child. This implies that descriptive semantics does not capture the intuition underlying our definition of the concept Momo. In fact, according to this intuition only the second model (where Momo is interpreted as the whole domain) is appropriate: any male member of the Charles dynasty satisfies the requirement that he is himself a man, and all his children have only male offspring.

To overcome this problem we shall now consider alternative types of semantics for terminological cycles.

4 – Fixed-point Semantics for Terminological Cycles

A terminology may be considered as a parallel assignment where the defined concepts are the variables, and the primitive concepts and roles are parameters.

Example 11
Let R, S be role names and A, B, P be concept names,[3] and let T be the terminology A = Q ⊓ ∀S:B, B = P ⊓ ∀R:B. We consider the following primitive interpretation J, which fixes the values of the parameters P, Q, R, S: dom(J) := $\{a_0, a_1, a_2, ...\}$, $P^J := \{a_1, a_2, a_3, ...\}$, $Q^J := \{a_0\}$, $R^J := \{(a_{i+1},a_i); i \geq 1\}$, and $S^J := \{(a_0,a_i); i \geq 1\}$.
For given values of the variables A, B, the parallel assignment A := Q ⊓ ∀S:B, B := P ⊓ ∀R:B yields new values for A, B. If A and B are interpreted as the empty set, an application of the assignment T yields the values Ø for A and $\{a_1\}$ for B. If we reapply the assignment to these values we obtain Ø for A and $\{a_1, a_2\}$ for B.

In the general case, a terminology T together with a primitive interpretation J defines a mapping T_J: $(2^{dom(J)})^n \to (2^{dom(J)})^n$, where n is the number of defined

[3]We shall no longer use intuitive names for concepts and roles, since I agree with [6], p.176, that “suggestive names can do more harm than good in semantic networks and other representation schemes.” Suggestive names may seemingly exclude models that are admissible with respect to the formal semantics.

concepts in T.

Definition 12
Let T be the terminology that consists of the concept definitions $C_1 = D_1, \ldots, C_n = D_n$, and let J be a primitive interpretation. The mapping $T_J: (2^{dom(J)})^n \rightarrow (2^{dom(J)})^n$ is defined as follows:
Let $\underline{A}$ be an element of $(2^{dom(J)})^n$ and let I be the interpretation defined by J and $\underline{A}$. Then $T_J(\underline{A}) := (D_1{}^I,\ldots,D_n{}^I)$.

For the above example we have seen that $T_J(\emptyset,\emptyset) = (\emptyset,\{a_1\})$ and $T_J(\emptyset,\{a_1\}) = (\emptyset,\{a_1, a_2\})$.

Obviously, the interpretation defined by J and $\underline{A}$ is a model of T if and only if $\underline{A}$ is a fixed-point of the mapping T_J, i.e., if and only if $T_J(\underline{A}) = \underline{A}$. In our example, the element $(\{a_0\},\{a_1, a_2, a_3, ...\})$ of $(2^{dom(J)})^2$ is a fixed-point of T_J. If we extend J to I by defining $A^I := \{a_0\}$, $B^I := \{a_1, a_2, a_3, ...\}$, we obtain a model of T.

One may now ask whether any primitive interpretation J can be extended to a model of T, or equivalently, whether any mapping T_J has a fixed-point. The answer is yes, because $(2^{dom(J)})^n$, ordered componentwise by inclusion, is a complete lattice (see Example 1) and the mappings T_J are monotonic.[4] Thus the following definition makes sense:

Definition 13 (three types of semantics for cyclic terminologies)
Let T be a terminology, possibly containing terminological cycles.
(1) The descriptive semantics *allows all models of T as admissible models.*
(2) The least fixed-point semantics *(lfp-semantics) allows only those models of T that come from the least fixed-point of a mapping T_J (lfp-models).*
(3) The greatest fixed-point semantics *(gfp-semantics) allows only those models of T that come from the greatest fixed-point of a mapping T_J (gfp-models).*

Any primitive interpretation J can uniquely be extended to a lfp-model (gfp-model) of T. In Example 10, the extension of J that interprets Momo as {James1, ..., JamesLast} is a lfp-model of T, and the extension that interprets Momo as dom(J) is a gfp-model of T. It is easy to see that, for cycle-free terminologies, lfp-, gfp- and descriptive semantics coincide (see [30], p.134). For terminologies with cycles, this is not the case, however, as we have just illustrated by Example 10. Thus one also obtains different notions of subsumption, depending on which semantics is employed.

[4]This can easily be proved; but it is also a consequence of Proposition 4.5, which states that these mappings are even downward ω-continuous.

Definition 14 (subsumption of concepts revisited)
Let T be a terminology and let A, B be concept names.

$$A \sqsubseteq_T B \text{ iff } A^I \subseteq B^I \text{ for all models } I \text{ of } T,$$
$$A \sqsubseteq_{lfp,T} B \text{ iff } A^I \subseteq B^I \text{ for all lfp-models } I \text{ of } T,$$
$$A \sqsubseteq_{gfp,T} B \text{ iff } A^I \subseteq B^I \text{ for all gfp-models } I \text{ of } T.$$

In this case we say that B subsumes *A in T w.r.t. descriptive semantics (resp. lfp-semantics, gfp-semantics).*

The next question we shall consider is how lfp-models (gfp-models) can be constructed from a given primitive interpretation. Nebel [29,30] claimed that the mappings T_J are even upward continuous, and that thus $lfp(T_J) = \cup_{i\geq 0} T_J{}^i(\text{bottom})$, where bottom denotes the least element of $(2^{dom(J)})^n$, namely the n-tuple (Ø,...,Ø). Unfortunately, this is not true.

Proposition 15
In general, we may have $lfp(T_J) \neq \cup_{i\geq 0} T_J{}^i(\text{bottom})$.

Proof
We consider Example 11. It is easy to see that $T_J{}^i(\emptyset,\emptyset) = (\emptyset,\{a_1, a_2, ..., a_i\})$. Thus $\cup_{i\geq 0} T_J{}^i(\emptyset,\emptyset) = (\emptyset,\{a_i;\ i \geq 1\})$, which is not a fixed-point since $T_J(\emptyset,\{a_i;\ i \geq 1\}) = (\{a_0\},\{a_i;\ i \geq 1\})$. ❑

In this example, the least fixed-point is reached by applying T_J once more after building the limit, i.e., $lfp(T_J) = T_J{\uparrow}^{\omega+1}$. In general, one may need even greater ordinals to obtain the least fixed-point. On the other hand, we shall now show that the greatest fixed-point can always be reached by ω-iteration of T_J.

Proposition 16
The mappings T_J are always downward ω-continuous. Conse-quently, the greatest fixed-point may be obtain as $gfp(T_J) = \cap_{i\geq 0} T_J{}^i(\text{top})$, where top denotes the greatest element of $(2^{dom(I)})^n$, i.e., top = (dom(I),...,dom(I)).

Proof
Let J be a primitive interpretation, and let $\underline{A}^{(0)} \supseteq \underline{A}^{(1)} \supseteq \underline{A}^{(2)} \supseteq ...$ be a decreasing chain in $(2^{dom(J)})^n$. We have to show that

$$\cap_{k\geq 0} T_J(\underline{A}^{(k)}) = T_J(\cap_{k\geq 0} \underline{A}^{(k)}).$$

For k ≥ 0, let I_k be the interpretation of T defined by J and $\underline{A}^{(k)}$ and let I be the interpretation defined by J and $\underline{A} := \cap_{k\geq 0} \underline{A}^{(k)}$. In the following, A_i denotes the i-th component of the tuple $\underline{A}$ and $A_i{}^{(k)}$ the i-th component of the tuple $\underline{A}^{(k)}$.By Definition 12, it is sufficient to demonstrate that, for any concept term D, we have

$$\bigcap_{k\geq 0} D^{I_k} = D^I.$$

We proceed by **induction on the size of D**.

(1) *D = P for a primitive concept P.* Then $D^I = P^J = D^{I_k}$ for all $k \geq 0$ and hence $\bigcap_{k\geq 0} D^{I_k} = P^J = D^I$.

(2) *D = C_i for a defined concept C_i.*[5] Then $D^I = A_i$, and for all $k \geq 0$, $D^{I_k} = A_i^{(k)}$.But $A_i = \bigcap_{k\geq 0} A_i^{(k)}$ by definition of $\underline{A}$.

(3) *D = E ⊓ F for concept terms E, F.* We have $D^I = E^I \cap F^I$ and by induction we get $E^I = \bigcap_{k\geq 0} E^{I_k}$ and $F^I = \bigcap_{k\geq 0} F^{I_k}$. Hence $D^I = (\bigcap_{k\geq 0} E^{I_k}) \cap (\bigcap_{k\geq 0} F^{I_k}) = \bigcap_{k\geq 0} (E^{I_k} \cap F^{I_k}) = \bigcap_{k\geq 0} D^{I_k}$.

(4) *D = ∀R:C for a role name R and a concept term C.* By Definition 8, $D^I = \{x \in \text{dom}(I);\ \forall y: ((x,y) \in R^I \rightarrow y \in C^I)\}$, and hence, by induction and the definition of I, $D^I = \{x \in \text{dom}(J);\ \forall y: ((x,y) \in R^J \rightarrow y \in \bigcap_{k\geq 0} C^{I_k})\}$. This means that we have

$$x \in D^I \text{ iff } \forall y: ((x,y) \in R^J \rightarrow \forall k: y \in C^{I_k}).$$

It is well-known (see e.g., [13], p. 305), that a formula of the form $\forall y: (A \rightarrow \forall k: B)$, where k has no free occurrence in A, is equivalent to the formula $\forall y:\forall k: (A \rightarrow B)$. If we permute the quantifiers[6] we obtain $\forall k:\forall y: (A \rightarrow B)$. This shows that

$$x \in D^I \text{ iff } \forall k:\forall y: ((x,y) \in R^J \rightarrow y \in C^{I_k}).$$

Since $\{x \in \text{dom}(J);\ \forall y: ((x,y) \in R^J \rightarrow y \in C^{I_k})\} = D^{I_k}$, we have shown that $\bigcap_{k\geq 0} D^{I_k} = D^I$. This completes the proof of the proposition. ❑

The two propositions show that, from a constructive point of view, gfp-semantics is preferable. However, if dom(J) is finite, the greatest and the least fixed-point can be reached after a finite number of applications of T_J, and as shown in [30] subsumption relationships do not change if models are restricted to be finite.

5 – Characterization of the Semantics using Finite Automata

The close connection between terminologies of $\mathcal{FL}_0$ and finite automata was first observed by B. Nebel [31]. He used this connection in the case of non-cyclic terminologies to show that subsumption for non-cyclic terminologies of $\mathcal{FL}_0$ is coNP-hard.

Before we can associate a semi-automaton $\mathcal{A}_T$ with a terminology T we must

[5]We assume that index(Ci) = i.

[6]This is the point where the proof for the least fixed-point goes wrong. In this case we would have the quantifiers "∀y:∃k:" which cannot be permuted.

transform T into some kind of normal form. It is easy to see that the concept terms $\forall R:(B \sqcap C)$ and $(\forall R:B) \sqcap (\forall R:C)$ are equivalent, i.e., they have the same extension in any interpretation. Hence, any concept term can be transformed into a finite conjunction of terms of the form $\forall R_1:\forall R_2:...\forall R_n:A$, where A is a concept name. We shall abbreviate the prefix "$\forall R_1:\forall R_2:...\forall R_n$" by "$\forall W$" where $W = R_1R_2...R_n$ is a word over $\mathbf{R_T}$, the set of role names occurring in T. In the case n = 0 we also write "$\forall \varepsilon:A$" instead of simply "A". For an interpretation I and a word $W = R_1R_2...R_n$, W^I denotes the composition $R_1^I \circ R_2^I \circ ... \circ R_n^I$ of the binary relations $R_1^I, R_2^I, ..., R_n^I$. The term ε^I denotes the identity relation, i.e., $\varepsilon^I = \{(d,d); d \in dom(I)\}$.

Definition 17
Let T be a terminology where all terms are normalized as described above. The (nondeterministic) semi-automaton with word transitions $\mathcal{A}_T$ is defined as follows: The alphabet of $\mathcal{A}_T$ is the set $\boldsymbol{R_T}$ of all role names occurring in T; the states of $\mathcal{A}_T$ are the concept names occurring in T; a terminological axiom of the form $A = \forall W_1:A_1 \sqcap ... \sqcap \forall W_k:A_k$ gives rise to k transitions, where the transition from A to A_i is labeled by the word W_i.

The next example illustrates Definition 17.

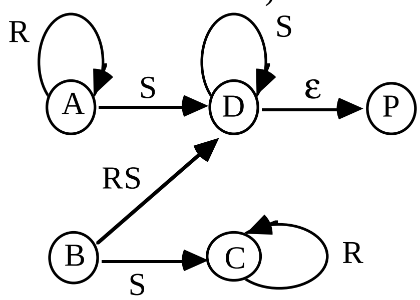
Example 18 (A normalized terminology and its semi-automaton)

$A = \forall R: A \sqcap \forall S: D$

$B = \forall RS: D \sqcap \forall S: C$

$C = \forall R: C$

$D = \forall S: D \sqcap P$



The primitive concepts are exactly those states in $\mathcal{A}_T$ that do not have successor states. The semi-automaton $\mathcal{A}_T$ can be used to characterize gfp- and descriptive semantics and, after a modification, also lfp-semantics.

5.1 – Characterization of the gfp-Semantics

Before we can show that subsumption w.r.t. gfp-semantics can be reduced to inclusion of regular languages, we need the following proposition, which describes under what conditions an individual d of a gfp-model I is in the extension A^I of a concept A.

Proposition 19
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be a gfp-model of T and let A be a concept name occurring in T. For any d ∈ dom(I) we have:

$d \in A^I$ *iff* *for all primitive concepts P, all words W ∈ L(A,P) and all individuals e ∈ dom(I): (d,e) ∈ W^I implies e ∈ P^I.*

A proof of this proposition can be found in Appendix A. For the terminology Momo = Man ⊓ ∀child: Momo of Example 10, L(Momo,Man) = child^* = {child^n; n ≥ 0}. Hence it is an immediate consequence of the proposition that this terminology—if interpreted with gfp-semantics—expresses value-restriction with respect to the reflexive-transitive closure of child. In this case, the condition of the proposition says that d ∈ Momo^I if and only if, for all n ≥ 0, and all e such that $d(\text{child}^I)^n e$, e ∈ Man^I holds. This means that for all e such that $d(\bigcup_{n\geq 0}(\text{child}^I)^n)e$, e ∈ Man^I holds. But the relation $\bigcup_{n\geq 0}(\text{child}^I)^n$ is the reflexive-transitive closure of child^I.

Proposition 19 also implies that concepts are never *inconsistent w.r.t. gfp-semantics*, i.e., for any terminology T and any concept A in T there exists a gfp-model I of T such that $A^I \neq \emptyset$. Obviously, it is enough to take the gfp-model that is defined by a primitive interpretation J satisfying P^J = dom(J) for all primitive concepts P.

The proposition can intuitively be understood as follows: The languages L(A,P) stand for the possibly infinite number of constraints of the form ∀W: P that the terminology imposes on A. An individual d is in the extension of A if and only if it satisfies all of these constraints. If a concept has to satisfy more constraints, its extension will become smaller. This motivates the following theorem, which characterizes subsumption w.r.t. gfp-semantics.

Theorem 20
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be a gfp-model of T and let A, B be concept names occurring in T. Subsumption in T can be reduced to inclusion of regular languages defined by $\mathcal{A}_T$. More precisely,

$$A \sqsubseteq_{gfp,T} B \quad \text{iff} \quad L(B,P) \subseteq L(A,P) \text{ for all primitive concepts } P.$$

Again, the proof is deferred to the appendix. In Example 18, B subsumes A w.r.t. gfp-semantics since L(B,P) = RSS^* is a subset of L(A,P) = R^*SS^*.

The theorem shows that the problem of determining subsumption w.r.t. gfp-semantics can be reduced to the inclusion problem for regular languages in polynomial time. If we want to solve the subsumption problem $A \sqsubseteq_{gfp,T} B$ for a terminology T with k primitive concepts, we have to solve k inclusion problems for regular languages that are defined by a nondeterministic semi-automaton having the same size as the terminology.

On the other hand, the inclusion problem for regular languages (given by arbitrary nondeterministic automata) can be reduced to the subsumption problem. Assume that $\mathcal{A}_1 = (\Sigma,Q_1,E_1)$ and $\mathcal{A}_2 = (\Sigma,Q_2,E_2)$ are two (nondeterministic) semi-automata defining the regular languages $L_1 = L_{\mathcal{A}_1}(p_1,q_1)$ and $L_2 = L_{\mathcal{A}_2}(p_2,q_2)$. Without loss of generality we may assume that Q_1 and Q_2 are disjoint and that $\mathcal{A}_1$ and $\mathcal{A}_2$ are trim, i.e., any state can reach the terminal state q_i and can be reached from the initial state p_i (see [12], p. 23). We consider the semi-automaton $\mathcal{A} = (\Sigma,Q_1 \cup Q_2 \cup \{t\},E)$, where t is a new state not occurring in $Q_1 \cup Q_2$ and $E = E_1 \cup E_2 \cup \{(q_1,\varepsilon,t), (q_2,\varepsilon,t)\}$. Obviously, $L_{\mathcal{A}_1}(p_1,q_1) = L_{\mathcal{A}}(p_1,t)$ and $L_{\mathcal{A}_2}(p_2,q_2) = L_{\mathcal{A}}(p_2,t)$. It is easy to see that $\mathcal{A} = \mathcal{A}_T$ for a terminology T that has the states in $Q_1 \cup Q_2$ as its defined concepts and the state t as the only[7] primitive concept. But then $L_1 \subseteq L_2$ if and only if $p_2 \sqsubseteq_{gfp,T} p_1$.

Corollary 21
The problem of determining subsumption w.r.t. gfp-semantics is PSPACE-complete.

Proof
We have seen that subsumption w.r.t. gfp-semantics can be reduced to inclusion of regular languages (defined by nondeterministic automata) in polynomial time and vice versa. It is well-known that the inclusion problem for regular languages defined by a nondeterministic automata is PSPACE-complete (see [14]). ❑

This shows that, even for our very small language, subsumption determination w.r.t. gfp-semantics is rather hard from a computational point of view. On the other hand, [31] shows that, even without cycles, this language has a coNP-complete subsumption problem.

5.2 – Characterization of the lfp-Semantics

This characterization must take into account both finite and infinite paths of the semi-automaton $\mathcal{A}_T$.

Proposition 22
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be the lfp-model of T defined by the primitive interpretation J and let A be a concept name occurring in T. For any $d_0 \in dom(I)$ we have $d_0 \in A^I$ iff the following two properties hold:
(P1) *For all primitive concepts P, all words $W \in L(A,P)$ and all individuals*

[7] Starting with trim automata guarantees this property.

$e \in dom(I)$, $(d_0,e) \in W^I$ implies $e \in P^I$.

(P2) *For all infinite paths A, W_1, C_1, W_2, C_2, W_3, C_3, ..., and all individuals d_1, d_2, d_3, ... there exists $n \geq 1$ such that $(d_{n-1},d_n) \notin W_n{}^I$.*

A proof of this proposition can be found in Appendix B. To illustrate the effect that condition P2 has, let us reconsider the "finite directed acyclic graph" example of the introduction. Thus, consider the terminology T that consists of the single axiom

$$\text{Dag} \;=\; \text{Node} \sqcap \forall \text{arc}: \text{Dag}.$$

The primitive interpretation J is defined by dom(J) := {a, b, c, d} =: Node^J, and arc^J := {(a,b), (b,b), (c,d)}. The concept Node is the only primitive concept, and since Node^J = dom(J), all elements of dom(J) satisfy condition P1. Thus, Proposition 19 yields that the greatest fixed-point model induced by J interprets Dag as the whole domain dom(J). Condition P2, however, is only satisfied for c and d, but neither for a nor for b. In fact, there is an infinite path Dag, arc, Dag, arc, Dag, arc, ... and we have an infinite sequence of individuals a, b, b, b, ... such that (a,b) $\in$ arc^J and (b,b) $\in$ arc^J. More generally, it is easy to see that conditions P2 excludes a node from the extension of Dag if, from this node, one can reach a cyclic path in the graph described by the arc-relation. The same is true for infinite instead of cyclic paths in the graph described by the arc-relation. This shows that in this case lfp-semantics is more appropriate than gfp-semantics.

As a consequence of condition P2 of the proposition, ε-cycles in $\mathcal{A}_T$—i.e., non-empty paths of the form B, ε, ..., ε, B—are important for the lfp-semantics. In particular, inconsistency of concepts can be described with the help of ε-cycles. We say that the concept A of T is *inconsistent w.r.t. lfp-semantics* iff it has the empty extension in all lfp-models of T.

Corollary 23

The concept A is inconsistent w.r.t. lfp-semantics if and only if there exists a path with label ε from A to a state B that is the initial state of an ε-cycle.

Proof

(1) Assume that there is a path A, ε, ..., ε, B and a non-empty path B, ε, ..., ε, B. Thus we have an infinite path starting with A where all transitions are labeled by ε. Since $d\varepsilon^I d$ for all lfp-models I and individuals d $\in$ dom(I), property (P2) of the proposition is never satified for A and arbitrary d. Hence A is inconsistent.

(2) Assume that A is inconsistent w.r.t. lfp-semantics. We define a primitive interpretation J as follows: dom(J) := $\{d_0\}$, P^J := $\{d_0\}$ for all primitive concepts

P, and $R^J := \emptyset$ for all roles R.
Let I be the lfp-model of T defined by J. Since A is inconsistent, we have $d_0 \notin A^I$. The definition of J implies that property (P1) of Proposition 22 holds for A, d_0. Hence property (P2) cannot hold. This means that there exists an infinite path A, W_1, C_1, W_2, C_2, W_3, C_3, ..., and individuals d_1, d_2, d_3, ... such that $(d_{n-1},d_n) \in W_n^I$ for all $n \geq 1$. The definition of J implies $d_n = d_0$ and $W_n = \varepsilon$ for all $n \geq 1$. Hence there is an infinite path starting with A where all transitions are labeled by ε, and since $\mathcal{A}_T$ has only finitely many states, there is a state B that occurs infinitely often in this path. ❑

An easy consequence of this corollary is that inconsistency of concepts w.r.t. lfp-semantics can be decided in linear time. Starting from A, one has to search along ε-transitions for an ε-cycle.

Because of the role ε-cycles play for inconsistency, the semi-automaton $\mathcal{A}_T$ has to be modified before we can express subsumption w.r.t. lfp-semantics. We add a new state Q_{loop} to $\mathcal{A}_T$, a transition with label ε from Q_{loop} to Q_{loop}, and for each role R in T a transition with label R from Q_{loop} to Q_{loop}. For any state B of $\mathcal{A}_T$ lying on an ε-cycle, we add a transition with label ε from B to Q_{loop}, and for any primitive concept P we add a transition with label ε from Q_{loop} to P. This modified semi-automaton will be called $\mathcal{B}_T$.

The effect of this modification is as follows: If A is inconsistent w.r.t. lfp-semantics—i.e., by Corollary 23, there exists a path with label ε from A to a state B in $\mathcal{A}_T$ that is the initial state of an ε-cycle in $\mathcal{A}_T$—then we have $L_{\mathcal{B}_T}(A,P) = \Sigma^*$ for all primitive concept P, and $U_{\mathcal{B}_T}(A) = \Sigma^* \cup \Sigma^\omega$ in the semi-automaton $\mathcal{B}_T$. This means that, for the smallest concepts, the languages are made as large as possible.

Obviously, $L_{\mathcal{A}_T}(B,P) \subseteq L_{\mathcal{B}_T}(B,P)$ and $U_{\mathcal{A}_T}(B) \subseteq U_{\mathcal{B}_T}(B)$ for all concepts B. More precisely, $L_{\mathcal{B}_T}(B,P) = L_{\mathcal{A}_T}(B,P) \cup \{UV;$ U is a *finite* word in $U_{\mathcal{A}_T}(B)$ and $V \in \Sigma^*\}$ and $U_{\mathcal{B}_T}(B) = U_{\mathcal{A}_T}(B) \cup \{UV;$ U is a *finite* word in $U_{\mathcal{A}_T}(B)$ and $V \in \Sigma^* \cup \Sigma^\omega\}$. This is so because, obviously, U is a finite word in $U_{\mathcal{A}_T}(B)$ iff U is the label of a finite path in $\mathcal{A}_T$ from B to a concept C which lies on an ε-cycle in $\mathcal{A}_T$.

Theorem 24
Let T be a terminology and let $\mathcal{B}_T$ be the corresponding modified semi-automaton. Then $A \sqsubseteq_{lfp,T} B$ iff $U_{\mathcal{B}_T}(B) \subseteq U_{\mathcal{B}_T}(A)$ and $L_{\mathcal{B}_T}(B,P) \subseteq L_{\mathcal{B}_T}(A,P)$ for all primitive concepts P.

The proof of the theorem is given in the Appendix B. In Example 18, B does not subsumes A w.r.t. lfp-semantics since U(B) contains the infinite word SRRR..., which is not in U(A).

If we want to decide subsumption with the help of this theorem, we have to show how the inclusion "$U_{\mathcal{B}_T}(B) \subseteq U_{\mathcal{B}_T}(A)$" can be decided. It is possible to split this problem into two subproblems. Let $F_{\mathcal{B}_T}$ contain all finite words of $U_{\mathcal{B}_T}$ and let $I_{\mathcal{B}_T}$ contain all infinite words of $U_{\mathcal{B}_T}$. Obviously, $U_{\mathcal{B}_T}(B) \subseteq U_{\mathcal{B}_T}(A)$ iff $F_{\mathcal{B}_T}(B) \subseteq F_{\mathcal{B}_T}(A)$ and $I_{\mathcal{B}_T}(B) \subseteq I_{\mathcal{B}_T}(A)$.

Lemma 25
Let $\mathcal{B}$ be an arbitrary semi-automaton with word transitions. Then $F_{\mathcal{B}}(B) \subseteq F_{\mathcal{B}}(A)$ can be decided by a PSPACE-algorithm.

Proof
The semi-automaton $\mathcal{B} = (\Sigma,Q,E)$ is modified to a semi-automaton $\mathcal{C} = (\Sigma,Q \cup \{Fin\},E')$ where Fin is a new state and $E' := E \cup \{(C,\varepsilon,Fin);\ C \in Q$ and C lies on an ε-cycle$\}$. Obviously, this modification can be done in polynomial time.

Claim: For all states $A \in Q$ we have $F_{\mathcal{B}}(A) = L_{\mathcal{C}}(A,Fin)$.
Proof of the Claim. (1) Assume that $W \in F_{\mathcal{B}}(A)$. Then there exists an infinite path A, W_1, C_1, W_2, C_2, W_3, C_3, ... in $\mathcal{B}$ that has W as label. Since W is a finite word almost all labels W_i have to be empty. Let $k \geq 1$ be such that $W_i = \varepsilon$ for all $i \geq k$. Then $W = W_1 \ldots W_{k-1}$ and there exist i, j such that $k \leq i < j$ and $C_i = C_j$. This means that C_i lies on an ε-cycle and W is the label of path from A to C_i. But then $W \in L_{\mathcal{C}}(A,Fin)$.
(2) Assume that $W \in L_{\mathcal{C}}(A,Fin)$. This means that there exists a path in $\mathcal{B}$ with label W from A to a state C that lies on an ε-cycle. Now $W \in F_{\mathcal{B}}(A)$, since there is an infinite path A, W, C, ε, C, ε, … with label W. ❑ (Claim)

The problem $L_{\mathcal{C}}(B,Fin) \subseteq L_{\mathcal{C}}(A,Fin)$ is an inclusion problem for regular languages, which can be decided by a PSPACE-algorithm. ❑

Lemma 26
Let $\mathcal{B}$ be an arbitrary semi-automaton with word transitions. Then $I_{\mathcal{B}}(B) \subseteq I_{\mathcal{B}}(A)$ can be decided by a PSPACE-algorithm.

Proof
The proof proceeds in three steps.
(1) The semi-automaton with word transitions $\mathcal{B} = (\Sigma,Q,E)$ can be modified in polynomial time to a semi-automaton with letter transitions $\mathcal{A} = (\Sigma,Q_1,E_1)$ such that the following properties hold:

- **(1.1)** $Q \subseteq Q_1$;
- **(1.2)** There does not exist an infinite path in $\mathcal{A}$ using only states of $Q_1 \setminus Q$;
- **(1.3)** For all A, B in Q and all finite words $W \neq \varepsilon$:
  $W \in L_{\mathcal{B}}(A,B)$ iff $W \in L_{\mathcal{A}}(A,B)$.

The additional states in Q_1 are intermediate states that are necessary for the elimination of transitions that are labeled by words of length greater than 1. Obviously, these intermediate states cannot give rise to new infinite paths. For the elimination of ε-transitions, see [19], p. 26, Theorem 2.2.

Claim 1: For all states $A \in Q$ we have $I_{\mathcal{B}}(A) = I_{\mathcal{A}}(A)$.
Proof of the Claim. Let W be an infinite word in $I_{\mathcal{B}}(A)$, i.e., there exists an infinite path A, W_0, C_1, W_1, C_2, W_2, C_3, ... in $\mathcal{B}$ that has W as label. Since W is an infinite word, there exist infinitely many indices $0 < i_1 < i_2 < \dots$ such that the words $W_0 \dots W_{i_1-1}$, $W_{i_1} \dots W_{i_2-1}$, … are not empty. By property (1.3), $W_0 \dots W_{i_1-1} \in L_{\mathcal{A}}(A,C_{i_1})$, $W_{i_1} \dots W_{i_2-1} \in L_{\mathcal{A}}(C_{i_1},C_{i_2})$, … . This shows that there exists an infinite path from A with label W in $\mathcal{A}$, i.e., $W \in I_{\mathcal{A}}(A)$.
On the other hand, let W be an infinite word in $I_{\mathcal{A}}(A)$, i.e., there exists an infinite path A, W_0, C_1, W_1, C_2, W_2, C_3, ... in $\mathcal{A}$ that has W as label. By property (1.2), there exist infinitely many indices $0 < i_1 < i_2 < \dots$ such that C_{i_1}, C_{i_2}, … are in Q. By property (1.3), $W_0 \dots W_{i_1-1} \in L_{\mathcal{B}}(A,C_{i_1})$, $W_{i_1} \dots W_{i_2-1} \in L_{\mathcal{B}}(C_{i_1},C_{i_2})$, … . This shows that there exists an infinite path from A with label W in $\mathcal{B}$, i.e., $W \in I_{\mathcal{B}}(A)$. ❑ (Claim 1)

(2) Without loss of generality we may now assume that all states of $\mathcal{A}$ lie on some infinite path. The other states can easily be eliminated in polynomial time. For a state A of $\mathcal{A}$ we define $E_{\mathcal{A}}(A) := \cup_{C \in Q_1} L_{\mathcal{A}}(A,C)$.

Claim 2: For all states $A, B \in Q_1$ we have $I_{\mathcal{A}}(B) \subseteq I_{\mathcal{A}}(A)$ iff $E_{\mathcal{A}}(B) \subseteq E_{\mathcal{A}}(A)$.
Proof of the Claim. Assume that $W \in I_{\mathcal{A}}(B) \setminus I_{\mathcal{A}}(A)$. Then all finite initial segments U of W are in $E_{\mathcal{A}}(B)$. We cannot have all finite initial segments U of W in $E_{\mathcal{A}}(A)$ since, by König's Lemma, this would imply that $W \in I_{\mathcal{A}}(A)$.
On the other hand, assume that $U \in E_{\mathcal{A}}(B) \setminus E_{\mathcal{A}}(A)$. Since all states of $\mathcal{A}$ lie on some infinite path, the path with label U can be extended to an infinite path, i.e., U is the initial segment of some infinite word $W \in I_{\mathcal{A}}(B)$. Now $W \notin I_{\mathcal{A}}(A)$ since otherwise we would have $U \in E_{\mathcal{A}}(A)$. ❑ (Claim 2)

(3) Obviously, the languages $E_{\mathcal{A}}(A)$ are regular languages defined by $\mathcal{A}$. Hence there is a PSPACE-algorithm that decides $E_{\mathcal{A}}(B) \subseteq E_{\mathcal{A}}(A)$. ❑

The two lemmata together with the theorem show that subsumption w.r.t. lfp-semantics can be decided by a PSPACE-algorithm.

Corollary 27
The problem of determining subsumption w.r.t. lfp-semantics is PSPACE-complete.

Proof
It remains to be shown that this problem is PSPACE-hard. This will be shown

by reducing the inclusion problem for regular languages to the subsumption problem. Assume that $\mathcal{A}_1 = (\Sigma, Q_1, E_1)$ and $\mathcal{A}_2 = (\Sigma, Q_2, E_2)$ are two semi-automata[8] defining the regular languages $L_1 = L_{\mathcal{A}_1}(p_1,q_1)$ and $L_2 = L_{\mathcal{A}_2}(p_2,q_2)$. Without loss of generality we may assume that Q_1 and Q_2 are disjoint and that $\mathcal{A}_1$ and $\mathcal{A}_2$ are trim (see proof of Corollary 21). We consider the semi-automaton $\mathcal{A} = (\Sigma, Q_1 \cup Q_2 \cup \{t, f\}, E)$, where t and f are a new states not occurring in $Q_1 \cup Q_2$, and $E = E_1 \cup E_2 \cup \{(q_1,\varepsilon,t), (q_2,\varepsilon,t)\} \cup \{(p_1,\varepsilon,f), (p_2,\varepsilon,f)\} \cup \{(f,\sigma,f); \sigma \in \Sigma\}$. Obviously, $L_{\mathcal{A}_1}(p_1,q_1) = L_{\mathcal{A}}(p_1,t)$ and $L_{\mathcal{A}_2}(p_2,q_2) = L_{\mathcal{A}}(p_2,t)$. In addition, $U_{\mathcal{A}}(p_1) = \Sigma^\omega = U_{\mathcal{A}}(p_2)$.
It is easy to see that $\mathcal{A} = \mathcal{A}_T = \mathcal{B}_T$ for a terminology T that has the states in $Q_1 \cup Q_2 \cup \{f\}$ as its defined concepts and the state t as the only primitive concept.
But then $L_1 \subseteq L_2$ if and only if $p_2 \sqsubseteq_{lfp,T} p_1$. ❑

5.3 – Characterization of the Descriptive Semantics

Firstly, we need a proposition for $\underline{A}$-gfp-models (see Corollary 4) that is similar to Proposition 19 for gfp-models.

Proposition 28
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let J be a primitive interpretation and let $\underline{A}$ be a tuple such that $T_J(\underline{A}) \subseteq \underline{A}$. Let I be the model of T defined by J and the tuple $\underline{A}$-gfp(T_J) (see Corollary 4).
For any concept A and any individual $d \in dom(I)$ we have: $d \in A^I$ iff the following two properties hold:
(1) *For all primitive concepts P, all words $W \in L(A,P)$, and all individuals $e \in dom(I)$, $(d,e) \in W^I$ implies $e \in P^I$.*
(2) *For all defined concepts B, all words $W \in L(A,B)$, and all individuals $e \in dom(I)$, $(d,e) \in W^I$ implies $e \in (\underline{A})_j$ (where j = index(B)).*

The proof is deferred to Appendix C. Using this proposition, we can characterize subsumption w.r.t. descriptive semantics. Infinite paths are still important but it is not enough to consider just their labels. The states that are reached infinitely often by this path are also significant. An infinite path that has initial state A and reaches the state C infinitely often will be represented in the form A, U_0, C, U_1, C, U_2, C, ... where the U_i are labels of non-empty paths from A to C for i = 0 and from C to C for i > 0.

Theorem 29
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let A, B be concepts in T. Then we have $A \sqsubseteq_T B$ iff the following two properties hold:

[8]Without loss of generality the transitions are only labeled by letters of the alphabet.

(P1) *For all primitive concepts P, L(B,P) $\subseteq$ L(A,P) holds.*
(P2) *For all defined concepts C and all infinite paths of the form B, U_0, C, U_1, C, U_2, C, ..., there exists $k \geq 0$ such that $U_0 \ldots U_k \in$ L(A,C).*

Again, the proof can be found in the appendix. This theorem clearly shows that structurally identical definitions need not lead to equivalent concepts. The names chosen for defined concepts that lie on infinite paths are also relevant. For the T-box

$$\text{Momo} = \text{Man} \sqcap \forall\text{child}: \text{Momo},$$
$$\text{Mnfo} = \text{Man} \sqcap \forall\text{child}: \text{Mnfo},$$

there is an infinite path Momo, child, Momo, child, ... in the corresponding semi-automaton, but there is no k such that $\text{child}^{k+1} \in$ L(Mnfo,Momo). This shows that Mnfo is not subsumed by Momo w.r.t. descriptive semantics.

If we want to decide subsumption using this theorem, it remains to be shown how (P2) can be decided for given states A, B, C of a semi-automaton. For this problem we cannot obtain an ad hoc reduction to an inclusion problem for regular languages. In the next section we shall see that the problem can be reduced to an inclusion problem for certain languages of infinite words, which have already been considered in the context of monadic second-order logic (see [8] and [12], Chapter XIV).

One should note, however, that it might not be the best solution to decide (P2) for each state C separately. For a fixed state C, it is easy to show that deciding (P2) is PSPACE-hard. It is not yet clear whether deciding the conjunction for all C is also PSPACE-hard.

6 – Büchi Automata and Subsumption w.r.t. Descriptive Semantics

Let $\mathcal{A}$ = (Σ,Q,E) be a semi-automaton with letter transitions and let I, T be subsets of Q. Since we are interested in languages of infinite words accepted by the automaton, we call $\mathcal{A}$ together with I, T a *Büchi automaton*. The language $B_{\mathcal{A}}(I,T) \subseteq \Sigma^\omega$ accepted by this automaton is defined as $B_{\mathcal{A}}(I,T) := \{W \in \Sigma^\omega;$ W is the label of an infinite path starting from some state in I and reaching some state of T infinitely often$\}$.

Let $L \subseteq \Sigma^*$ be an arbitrary language of finite words. Then L^ω is the set of all infinite words W that can be obtained as $W = W_1W_2W_3\ldots$ where W_1, W_2, W_3, … are non-empty words in L. The languages L^ω for regular L can be used for an alternative characterization of the languages accepted by Büchi automata.

Theorem 30 (Büchi-McNaughton)

(1) *For any language $L \subseteq \Sigma^\omega$ the following two conditions are equivalent:*

(1.1) *$L = B_{\mathcal{A}}(I,T)$ for a Büchi automaton $\mathcal{A}$.*

(1.2) *L is the finite union of languages $H \cdot K^\omega$ where H and K are regular languages in Σ^*.*[9]

(2) *The class of all languages accepted by Büchi automata is closed under the Boolean operations union, intersection and complement.*

Proof

See [12], p.382, Theorem 1.4. The proof is constructive, but it takes eight pages, which shows that we are dealing with a hard problem. ❑

As an easy consequence of this theorem we obtain

Corollary 31

The inclusion problem is decidable for the class of all languages accepted by Büchi automata.

Proof

Obviously, $L_1 \subseteq L_2$ iff $L_1 \cap (\Sigma^\omega \setminus L_2) = \emptyset$. Thus the inclusion problem can be reduced to the emptiness problem since the proof of Theorem 1.4 in [12] is effective, i.e., from given Büchi automata for L_1 and L_2 one can effectively construct a Büchi automaton for $L_1 \cap (\Sigma^\omega \setminus L_2)$. Note, however, that this automaton may have a size that is exponential in the size of the initial automata (see [34,39] for size bounds for the complement automaton).

Let $L = B_{\mathcal{A}}(I,T)$ for a Büchi automaton $\mathcal{A}$. It is easy to see that $L \neq \emptyset$ iff there exists $i \in I$, $t \in T$ such that there is a path from i to t and a path from t to t. This is an easy search problem in a graph, which can be done in time polynomial in the size of $\mathcal{A}$. ❑

The argument used in the proof of Corollary 31 does not yield the complexity of the inclusion problem. However, [39] shows that equality of languages accepted by Büchi automata can be decided with a PSPACE-algorithm. Since $L_1 \subseteq L_2$ iff $L_1 \cap L_2 = L_1$, and since the automaton for the intersection can be constructed in polynomial time (see [40], proof of Lemma 1.2), we obtain a PSPACE-algorithm for the inclusion problem. On the other hand, inclusion of regular languages can be reduced to inclusion of languages accepted by Büchi automata as follows. Let L_1, L_2 be regular languages over Σ, and let # be a symbol not contained in Σ. Then $L_1 \subseteq L_2$ iff $L_1 \cdot \{\#\}^\omega \subseteq L_2 \cdot \{\#\}^\omega$. By Theorem 30, $L_1 \cdot \{\#\}^\omega$)and $L_2 \cdot \{\#\}^\omega$ are languages accepted by Büchi

[9]The language $H \cdot K^\omega$ consists of the infinite words $W_0W_1W_2W_3\ldots$ where $W_0 \in H$ and $W_1, W_2, W_3, \ldots$ are non-empty words in K.

automata. Thus we have shown:

Proposition 32
The inclusion problem for the class of all languages accepted by Büchi automata is PSPACE-complete.

It remains to be shown that our problem (P2) from Section 5.3 can be reduced to an inclusion problem for languages accepted by Büchi automata. Let $\mathcal{B} = (\Sigma,Q,E)$ be a semi-automaton with word transitions, and let A, B, C be states in Q. We want to decide whether the following property holds:

(P2) For all infinite paths of the form B, U_0, C, U_1, C, U_2, C, ..., there exists $k \geq 0$ such that $U_0 \ldots U_k \in L(A,C)$.

Let # be a new symbol not contained in Σ and let p, q be states in $\mathcal{A}$. We define the language $L_{p,q}$ over the alphabet Σ as

$$L_{p,q} := \{W;\ W \in \Sigma^* \text{ is the label of a non-empty path from p to q}\}.$$

For a language L over Σ, the language L# over $\Sigma_\# := \Sigma \cup \{\#\}$ is defined as

$$L\# := \{W\#;\ W \in L\}.$$

Obviously, the languages $L_{p,q}$ and $L_{p,q}\#$ are regular. Let $\psi: {\Sigma_\#}^* \to \Sigma^*$ be the homomorphism defined by $\psi(\sigma) = \sigma$ for $\sigma \in \Sigma$ and $\psi(\#) = \varepsilon$. Then $\psi^{-1}(L_{p,q}) := \{W \in {\Sigma_\#}^*;\ \psi(W) \in L_{p,q}\}$ and $\psi^{-1}(L_{p,q})\#$ are regular (see [19], Theorem 3.5).

Lemma 33
(P2) holds for A, B, C iff $(L_{B,C}\#)(L_{C,C}\#)^\omega \subseteq (\psi^{-1}(L_{A,C})\#)(L_{C,C}\#)^\omega$.

Proof
(1) Assume that (P2) holds. Let W be an element of $(L_{B,C}\#)(L_{C,C}\#)^\omega$, i.e., $W = U_0\#U_1\#U_2\#\ldots$, where U_0 is the label of a non-empty path from B to C and the U_i for $i \geq 1$ are labels of non-empty paths from C to C. By (P2) there exists $k \geq 0$ such that $U_0 \ldots U_k \in L(A,C)$. Hence $U_0 \ldots U_k$ is an element of $L_{A,C}$. But then $U_0\#\ldots\#U_k$ is an element of $\psi^{-1}(L_{A,C})$ and thus $W = U_0\#U_1\# \ldots U_k\#U_{k+1}\#\ldots \in (\psi^{-1}(L_{A,C})\#)(L_{C,C}\#)^\omega$.
(2) Assume that $(L_{B,C}\#)(L_{C,C}\#)^\omega \subseteq (\psi^{-1}(L_{A,C})\#)(L_{C,C}\#)^\omega$. Let B, U_0, C, U_1, C, U_2, C, ... be an infinite paths starting with B and reaching C infinitely often. Then we know that the infinite word $U_0\#U_1\#U_2\#\ldots$ is an element of $(L_{B,C}\#)(L_{C,C}\#)^\omega \subseteq (\psi^{-1}(L_{A,C})\#)(L_{C,C}\#)^\omega$. Since the last symbol of any word in in $\psi^{-1}(L_{A,C})\#$ is #, there exists $k \geq 0$ such that $U_0\#\ldots U_k\#$ is an element of $\psi^{-1}(L_{A,C})\#$. But then $U_0\#\ldots U_{k-1}\#U_k \in \psi^{-1}(L_{A,C})$, and $U_0 \ldots U_k \in L_{A,C}$. ❑

We know by Theorem 30 that $(L_{B,C}\#)(L_{C,C}\#)^\omega$ and $(\psi^{-1}(L_{A,C})\#)(L_{C,C}\#)^\omega$ are languages accepted by Büchi automata. Thus, by Proposition 32, the inclusion problem $(L_{B,C}\#)(L_{C,C}\#)^\omega \subseteq (\psi^{-1}(L_{A,C})\#)(L_{C,C}\#)^\omega$ can be decided by a PSPACE-algorithm. This yields

Corollary 34
Subsumption w.r.t. descriptive semantics can be decided with polynomial space using Büchi automata.

Büchi automata are, however, not indispensable for deciding subsumption w.r.t. gfp-semantics. Using Theorem 29 from above, B Nebel was able to characterize equivalence of concepts w.r.t. descriptive semantics with the help of deterministic finite automata. This characterization also yields PSPACE-algorithms for equivalence and for subsumption w.r.t. descriptive semantics (see [32]). It is still an open problem whether these problems are PSPACE-hard.

7 – Extensions of the Results for gfp-Semantics

We consider two extensions of the results for gfp-semantics. In the first subsection, we shall allow an additional concept forming construct, namely so-called exists-restrictions. In the second subsection, we shall introduce an assertional component into our terminological system, and consider hybrid inferences with respect to the terminological and the assertional part of the knowledge base.

7.1 – The Language $\mathcal{FL}^-$ and gfp-Semantics

In order to extend our language $\mathcal{FL}_0$ to the language $\mathcal{FL}^-$ of [23], we have to add a fourth rule to the definition of concept terms (Definition 6): Let R be a role name.

(4) Then ∃R is a concept term. *(exists-restriction)*

For example, using this new construct, the concept Father can be defined as

$$\text{Father} = \text{Man} \sqcap \exists\text{child}$$

This means that a father is a man that has a child. The semantics of the exists-restriction is defined in the obvious way, namely

$(\exists R)^I := \{d \in \text{dom}(I);$ there exists $e \in \text{dom}(I)$ such that $(d,e) \in R^I\}$.

Let T be a terminology of the language $\mathcal{FL}^-$ and let J be a primitive interpretation. The mapping T_J is defined as in Definition 12. It is easy to see that this mapping is still downward ω-continuous. Hence T_J has a greatest fixed-point, which can be obtained as $\text{gfp}(T_J) = \bigcap_{i \geq 0} T_J^i(\text{top})$.

Any concept term of $\mathcal{FL}^-$ can be transformed into a finite conjunction of terms of the form $\forall R_1{:}\forall R_2{:}...\forall R_n{:}D$, where D is a concept name or a term of the form ∃R. As in Section 5, the prefix "$\forall R_1{:}\forall R_2{:}...\forall R_n$" will be abbreviated by "∀W" where $W = R_1R_2\ ...R_n$. Let T be a terminology of $\mathcal{FL}^-$. The corresponding (nondeterministic) semi-automaton $\mathcal{A}_T$ is defined as in Definition 17. The only difference is that we also have the terms ∃R occurring in T as states of $\mathcal{A}_T$. These states are similar to the states P for primitive P in that they do not have successor states. We shall see that this similarity also extends to the characterization of gfp-semantics and of subsumption w.r.t. gfp-semantics.

Proposition 35
Let T be a terminology of $\mathcal{FL}^-$, and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be a gfp-model of T, and let A be a concept name occurring in T. For any $d \in dom(I)$ we have $d \in A^I$ iff the following two properties hold:

(1) *For all primitive concepts P, all words $W \in L(A,P)$, and all individuals $e \in dom(I)$, $(d,e) \in W^I$ implies $e \in P^I$.*

(2) *For all terms ∃R in T, all words $W \in L(A,\exists R)$, and all individuals $e \in dom(I)$, $(d,e) \in W^I$ implies $e \in (\exists R)^I$, i.e., there is $f \in dom(I)$ such that $(e,f) \in R^I$.*

Proof
The proof is very similar to the proof of Proposition 19. ❑

Theorem 36
Let T be a terminology of $\mathcal{FL}^-$, and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be a gfp-model of T and let A, B be concept names occurring in T. Then we have:

$$A \sqsubseteq_{gfp,T} B \quad \text{iff} \quad L(B,P) \subseteq L(A,P) \text{ for all primitive concepts P in T, and } L(B,\exists R) \subseteq L(A,\exists R) \text{ for all terms } \exists R \text{ occurring in T.}$$

A proof of the theorem is given in Appendix D. The theorem shows that, with respect to subsumption, terms of the form ∃R behave just like primitive concepts. As a consequence, we obtain:

Corollary 37
With respect to gfp-semantics, the subsumption problem for $\mathcal{FL}^-$ can be reduced in linear time to the subsumption problem for $\mathcal{FL}_0$.

Proof
Assume that T is a T-box of $\mathcal{FL}^-$. For any role R in T let P_R be a new primitive concept. Now substitute any ∃R term in T by P_R. This yields a T-box T_0 of $\mathcal{FL}_0$, which has the same size as T. In addition, Theorems 20 and 36 imply that

$A \sqsubseteq_T B$ iff $A \sqsubseteq_{T_0} B$. ❑

Subsumption relationships w.r.t. gfp-semantics in $\mathcal{FL}^{-}$ can thus be computed by a PSPACE-algorithm. Since $\mathcal{FL}_0$ is a sublanguage of $\mathcal{FL}^{-}$, the subsumption problem w.r.t. gfp-semantics in $\mathcal{FL}^{-}$ is also PSPACE-hard.

Corollary 38
The problem of determining subsumption w.r.t. gfp-semantics in $\mathcal{FL}^{-}$ is PSPACE-complete.

The characterization of descriptive semantics for $\mathcal{FL}_0$ (Proposition 28 and Theorem 29) can be generalized to $\mathcal{FL}^{-}$ in an analogous way, i.e., the terms $\exists R$ are treated like primitive concepts as in condition (2) of Proposition 35. For lfp-semantics, one can also prove an analogous generalization of Proposition 22. For subsumption, one runs into new problems, though. The reason is that there exists an additional source of inconsistency (see Example 39 below). For this reason, an appropriate generalization of Theorem 24 probably requires a more sophisticated modification of the semi-automaton.

Example 39
Consider the terminology T: $A = \forall S{:}A$, $B = \forall R{:}B \sqcap \exists R$. The concept B has the empty extension in all lfp-models of T. In fact, assume that J is a primitive interpretation, and let λ be the least ordinal such that $(T_J\!\uparrow\!\lambda)_2 \neq \emptyset$ (where index(B) = 2). Evidently, λ is a successor ordinal, i.e., $\lambda = \alpha + 1$ for some ordinal α. Let I be the interpretation of T defined by J and $T_J\!\uparrow\!\alpha$. Now $d \in (T_J\!\uparrow\!\lambda)_2$ means that $d \in (\forall R{:}B)^I \cap (\exists R)^I$. From $d \in (\exists R)^I$ we obtain some individual e such that dR^Ie, and $d \in (\forall R{:}B)^I$ yields $e \in B^I$. This contradicts the fact that $B^I = (T_J\!\uparrow\!\alpha)_2 = \emptyset$.
Since B is inconsistent w.r.t. lfp-semantics, we know that $B \sqsubseteq_{lfp,T} A$. But $U_{\mathcal{B}_T}(A) = \{SSS\ldots\} \not\subseteq U_{\mathcal{B}_T}(B) = \{RRR\ldots\}$.

7.2 – Extending $\mathcal{FL}_0$ by an Assertional Formalism

A terminology (T-box) T restricts the number of possible worlds (from all interpretations to the models of T); a world description (A-box) A[10] describes a part of a given world. Terminological systems that allow the user to state both terminological and assertional knowledge are sometimes called hybrid systems.

Definition 40 (world descriptions, A-boxes)
*Let **C** be a set of concept names, **R** be a set of role names, and **I** be a set of*

[10]In this subsection, A will always stand for an A-box. To avoid overloading, A will no longer be used as a metavariable for concept names.

individual names. A world description (A-box) is a finite set of axioms of the form C(a) or R(a,b) where a, b are constants in ***I****, C is a concept name, and R is a role name.*

For example, let Man be a concept name, child be a role name, and WILLY and BRIAN be individual names. Then Man(WILLY) and child(WILLY,BRIAN) can be part of a world description. This means that Willy is a man, who has the child Brian.

Definition 41 (interpretations and models)
Let T be a T-box of $\mathcal{FL}_0$ and A be an A-box defined over the same sets of concept and role names. An interpretation of T (see Definition 8) can be extended to an interpretation of T ∪ A as follows: the interpretation function does not only assign subsets of dom(I) to concept names, and binary relations on dom(I) to role names, but also individuals of dom(I) to individual names, i.e., for any individual name a, a^I is an element of dom(I).
An interpretation I of T ∪ A is a model of T ∪ A iff I is a model of T and satisfies

> $a^I \in C^I$ *for all axioms C(a) in A,* $(a^I,b^I) \in R^I$ *for all axioms R(a,b) in A, and*
> $a^I \neq b^I$ *for all individual names* $a \neq b$ *in* ***I*** *(unique name assumption).*[11]

A model I of T ∪ A is a gfp-model (lfp-model) of T ∪ A iff I is a gfp-model (lfp-model) of T.

Let T be a T-box of $\mathcal{FL}_0$. If we take a primitive interpretation J with P^J = dom(J) for all primitive concepts P, and R^J = dom(J) × dom(J), then $gfp(T_J)$ = top by Proposition 19. This shows that the gfp-model of T defined by J is a model of T ∪ A for any A-box A. Thus any combination T ∪ A of a T-box of $\mathcal{FL}_0$ with an A-box is consistent w.r.t. gfp-semantics, and w.r.t. descriptive semantics. But such a combination need not have an lfp-model. In fact, if C is a concept in T that is inconsistent w.r.t lfp-semantics (see Corollary 23), and A contains an axiom C(a), then T ∪ A does not have an lfp-model.

An important inference service concerning both T-box and A-box is computing instance relationships, i.e., determining which new assertions of the form C(a) can be deduced from a given T-box and A-box.

Definition 42 (instance relationship)
Let T be a T-box of $\mathcal{FL}_0$ and A be an A-box defined over the same sets of concept and role names. Let a be an individual name in A, and C be a concept name in T. Then

[11]Note that we do not impose a closed world assumption; e.g., if D(b) is not in A, we may nevertheless have $b^I \in D^I$ in a model I of T ∪ A.

$$a \in_{T \cup A} C \text{ iff } a^I \in C^I \text{ for all models } I \text{ of } T \cup A,$$
$$a \in_{lfp, T \cup A} C \text{ iff } a^I \in C^I \text{ for all lfp-models } I \text{ of } T \cup A,$$
$$a \in_{gfp, T \cup A} C \text{ iff } a^I \in C^I \text{ for all gfp-models } I \text{ of } T \cup A.$$

In this case we say that a is an instance of C in T ∪ A w.r.t. descriptive semantics (resp. lfp-semantics, gfp-semantics).

In the following we shall only consider instance relationships with respect to gfp-semantics. We have seen that a T-box T of $\mathcal{FL}_0$ gives rise to a semi-automaton $\mathcal{A}_T$ that has the concept names of T as states, and the set of role names in T as alphabet. Without loss of generality we may assume that $\mathcal{A}_T$ is a semi-automaton with letter transitions. In fact, Proposition 19 and Theorem 20 show that, for gfp-semantics, we are only interested in regular languages of the form $L_{\mathcal{A}_T}(A,P)$. These languages do not change if we transform the semi-automaton with word transitions into a semi-automaton with letter transitions. An A-box A defines a semi-automaton (with letter transitions) $\mathcal{A}_A$ as follows: the states of $\mathcal{A}_A$ are the individual names of A; the alphabet of $\mathcal{A}_A$ are the role names occurring in A; an axiom of the form R(a,b) gives rise to a transition from a to b with label R.

We can now build the *product semi-automaton* $\mathcal{B}_{T \cup A} = \mathcal{A}_T \times \mathcal{A}_A$ of $\mathcal{A}_T$ and $\mathcal{A}_A$ (see e.g., [12], p. 17). The states of $\mathcal{B}_{T \cup A}$ are pairs (C,a) where C is a state of $\mathcal{A}_T$ and a is a state of $\mathcal{A}_A$; $\mathcal{B}_{T \cup A}$ has a transition with label R from (C,a) to (D,b) iff $\mathcal{A}_T$ has a transition from C to D with label R, and $\mathcal{A}_A$ has a transition from a to b with label R. Obviously, $W \in L_{\mathcal{B}_{T \cup A}}((C,a),(D,b))$ iff $W \in L_{\mathcal{A}_T}(C,D)$ and $W \in L_{\mathcal{A}_A}(a,b)$.

Theorem 43
Let T be a T-box of $\mathcal{FL}_0$ and A be an A-box defined over the same sets of concept and role names. Let b be an individual name in A and B be a concept name in T. Then $b \in_{gfp, T \cup A} B$ iff for all primitive concepts P, and all words $W \in L_{\mathcal{A}_T}(B,P)$ there exist concepts E, F, a word U, and an individual name f such that

(1) $W \in L_{\mathcal{A}_T}(E,P)$,
(2) $U \in L_{\mathcal{B}_{T \cup A}}((F,f),(E,b))$ and F(f) is an axiom in A.

Since, at first sight, the conditions in the theorem seem to be rather complex, we try to give an intuitive explanation what these conditions mean. A formal proof of the theorem can be found in Appendix E. First note that "$W \in L_{\mathcal{A}_T}(B,P)$" means that any element of (the extension of) B must satisfy the value restriction ∀W: P. Second, we will argue that conditions (1) and (2) of the theorem imply that (the interpretation of) b satisfies this value restriction. From condition (1) we can deduce that any element of E satisfies ∀W: P, and thus it is sufficient to convince ourself that conditions (2) implies that b is an element of E. First, note that F(f) $\in$ A implies that f is an element of F, and U $\in$ $L_{\mathcal{B}_{T\cup A}}((F,f),(E,b))$ implies that U $\in$ $L_{\mathcal{A}_T}(F,E)$, i.e., f satisfies the restriction $\forall$U: E. But if we consider U $\in$ $L_{\mathcal{B}_{T\cup A}}((F,f),(E,b))$ with respect to the second component of the product semi-automaton, then we see that b can be reached from f via U (according to the A-box), which implies that b must be in E.

It remains to be shown that the property stated on the right hand side of the theorem can be decided for given b, B. To this purpose, we define *Q(b) := {E; there exists a state (F,f) in* $\mathcal{B}_{T\cup A}$ *and a word U such that U* $\in$ $L_{\mathcal{B}_{T\cup A}}((F,f),(E,b))$ *and F(f) is an axiom in A}*. Computing Q(b) for a give individual name b is a simple search problem in a graph; this can be done in time polynomial in the size of $\mathcal{B}_{T\cup A}$.

Lemma 44
The right hand side of the theorem holds for given b, B if and only if for all primitive concepts P, $L_{\mathcal{A}_T}(B,P) \subseteq \cup_{E\in Q(b)} L_{\mathcal{A}_T}(E,P)$ *holds.*

Proof
(1) Assume that $L_{\mathcal{A}_T}(B,P) \subseteq \cup_{E\in Q(b)} L_{\mathcal{A}_T}(E,P)$ holds, and let W be an element of $L_{\mathcal{A}_T}(B,P)$. Then W $\in$ $L_{\mathcal{A}_T}(E,P)$ for some E $\in$ Q(b). The definition of Q(b) yields F, f and a word U such that (1) and (2) of the theorem hold.
(2) Assume that the right hand side of the theorem holds, and let W be an element of $L_{\mathcal{A}_T}(B,P)$ where P is primitive. Then we get E, F, U, f satisfying (1) and (2) of the theorem. This means that W $\in$ $L_{\mathcal{A}_T}(E,P)$ and E $\in$ Q(b). ❏

The lemma together with the theorem shows that there is a PSPACE-algorithm for instance testing since the instance problem "b $\in_{gfp,T\cup A}$ B?" can be reduced to an inclusion problem for regular languages in polynomial time. On the other hand, subsumption determination can be reduced to instance testing in linear time.

Lemma 45
Let T be a T-box of $\mathcal{FL}_0$*, and let C, D be concept names occurring in T. Let A be the A-box containing C(c) as the only axiom. Then we have* $c \in_{gfp,T\cup A} D$ *if and only if* $C \sqsubseteq_{gfp,T} D$.

Proof
(1) The "if" direction is trivial.
(2) Assume that $C \not\sqsubseteq_{gfp,T} D$, i.e., there exists a gfp-model I of T such that C^I is not contained in D^I. This means that there exists an individual e $\in$ dom(I) such that e $\in$ $C^I \setminus D^I$. The interpretation I of T is extended to the interpretation I of T $\cup$ A by defining c^I := e. Obviously, I is a model of T $\cup$ A, but $c^I \notin D^I$. This

shows that $c \notin_{gfp,T \cup A} D$. ❑

Since the subsumption problem w.r.t. gfp-semantics in $\mathcal{FL}_0$ is PSPACE-complete, we have thus proved:

Corollary 46
Instance testing w.r.t. gfp-semantics in $\mathcal{FL}_0$ is PSPACE-complete.

8 – Conclusion and Related Work

We have restricted our attention to the rather small terminological representation language $\mathcal{FL}_0$, because for this language the meaning of terminological cycles with respect to different types of semantics—and in particular, the subsumption problem with respect to these semantics—could completely be characterized with the help of finite automata (see Section 5). These results may help to decide what kind of semantics for cyclic definitions is most appropriate for a particular representation task, possibly not only for this small language, but also for suitably extended languages.

We have seen that the results for subsumption in $\mathcal{FL}_0$ can be generalized in two directions. First, they have been extended to cyclic definitions in a larger language: we have shown that our automata-theoretic approach also applies to subsumption w.r.t. gfp-semantics in the language $\mathcal{FL}^{-}$ of [23]. It is, however, not clear how this approach could be extended to languages allowing both conjunction and disjunction of concepts. Also note that in the presence of negation of concepts, greatest and least fixed-point need no longer exist. As a second way of extending the results of Section 5, we have shown that, for gfp-semantics, hybrid inferences such as “instance testing” can also be treated by our automata-theoretic approach.

Since the first publication of a preliminary version of this work in [2,3], several other papers on cyclic definitions in terminological representation languages have appeared. Dionne, Mays, and Oles [10,11] give an intensional semantics for cyclic definitions in roughly the same language we have considered here. This is done by mapping concept descriptions to (possibly non-well-founded) sets that embody the “abstract structure” of the descriptions. In [11] it is shown, however, that the “structural” subsumption relation obtained this way coincides with subsumption w.r.t. gfp-semantics (as defined above).

As pointed out in the introduction, one motivation for using cyclic definitions in terminologies is that they can be used to express transitive closure of roles. Alternatively, one could allow for transitive closure as role constructor. In [4], the role constructors union, composition, and transitive closure are added to terminological representation languages. For $\mathcal{FL}_0$, it is shown that this extension has the same expressive power as $\mathcal{FL}_0$ with cyclic definitions interpreted by gfp-semantics. For a considerably larger language, called $\mathcal{ALC}$, subsumption in the extended language, called $\mathcal{ALC}_{\text{trans}}$, is still decidable. Interestingly, subsumption w.r.t. descriptive semantics for cyclic $\mathcal{ALC}$-terminologies can be reduced to the subsumption problem for $\mathcal{ALC}_{\text{trans}}$. Similar results have independently been obtained by K. Schild as byproducts of the correspondence he exhibits between $\mathcal{ALC}_{\text{trans}}$ and propositional dynamic logics (see [36]).

Employing a similar correspondence between $\mathcal{ALC}$ with cyclic definitions and the propositional mu-calculus, Schild [37] and Giacomo and Lenzerini [15] introduce a more flexible treatment of cyclic definitions in $\mathcal{ALC}$, where lfp-, gfp- and descriptive semantics coexist. [9] proposes to split the T-box into a "schema" and a "view" terminology, where the schema terminology is interpreted with descriptive semantics, and the view terminology is interpreted with an appropriate fixed-point semantics. The idea is that the terminological axioms in the view terminology are really seen as definitions (and thus must yield a unique extension for the defined concepts), whereas the axioms in the schema terminology function as integrity constraints (restricting the possible extensions of the defined concepts).

9 – References

[1] Aho, A.V., Ullman, J.D., Universality of data retrieval languages, in *Proceedings of the 6th ACM Symposium on Principles of Programming Languages*, 1979.

[2] Baader, F., Terminological cycles in KL-ONE-based knowledge representation languages, in *Proceedings of the 8th National Conference of the American Association for Artificial Intelligence*, AAAI-90, Boston, 1990.

[3] Baader, F., Terminological cycles in KL-ONE-based knowledge representation languages, DFKI Research Report RR-90-01, DFKI Kaiserslautern, 1990.

[4] Baader, F., Augmenting concept languages by transitive closure of roles: An alternative to terminological cycles, in *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, IJCAI-91, Sydney, 1991.

[5] Baader, F., Hollunder, B., KRIS: Knowledge representation and inference system, SIGART Bulletin, 2 (1991).

[6] Brachman, R.J., Schmolze, J.G., An overview of the KL-ONE knowledge representation system, Cognitive Science, 16 (1985).

[7] Brachman, R.J., Pigman-Gilbert, V., Levesque, H.J., An essential hybrid reasoning system: Knowledge and symbol level accounts in KRYPTON, in *Proceedings of the 9th International Joint Conference on Artificial Intelligence*, Los Angeles, 1985.

[8] Büchi, J.R., On a decision method in restricted second order arithmetic, in *Proceedings of the 1960 Congress on Logic, Methodology and Philosophy of Science,* Stanford, 1960.

[9] Buchheit, M., Donini, F.M., Nutt, W., Schaerf, A., Refining the structure of terminological systems: Terminology = Schema + Views, in *Proceedings of the 12th National Conference of the American Association for Artificial Intelligence*, AAAI-94, Seattle, 1994.

[10] Dionne, R., Mays, E., Oles, F.J., A non-well-founded approach to terminological cycles, in *Proceedings of the 10th National Conference of the American Association for Artificial Intelligence*, AAAI-92, San Jose, 1992.

[11] Dionne, R., Mays, E., Oles, F.J., The equivalence of model-theoretic and structural subsumption in description logics, in *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, IJCAI-93, Chambery, 1993.

[12] Eilenberg, S., *Automata, Languages and Machines, Vol. A*, Academic Press, New York/London, 1974.

[13] Gallier, J.H., *Logic for Computer Science, Foundations of Automated Theorem Proving,* Harper&Row, New York, 1986.

[14] Garey, M.R., Johnson, D.S., *Computers and Intractability: A Guide to the Theory of NP-Completeness,* Freeman, San Francisco, 1979.

[15] Giacomo, G., Lenzerini, M., Concept language with number restrictions and fixpoints, and its relationship with mu-calculus, in *Proceedings of the 11th European Conference on Artificial Intelligence*, ECAI-94, Amsterdam, 1994.

[16] Gurevich, Y., Shelah, S., Fixed-point extensions of first-order logic, in *Proceedings of the 26th Annual Symposium on Foundations of Computer Science,* IEEE Computer Society Press, 1985.

[17] Gurevich, Y., Shelah, S., Fixed-point extensions of first-order logic, Annals of Pure and Applied Logic, 32 (1986).

[18] Halmos, P.R., *Naive Set Theory*, Springer Verlag, New York, 1974.

[19] Hopcroft, J.E., Ullman, J.D., *Introduction to Automata Theory, Languages, and Computation,* Addison-Wesley, Reading, 1979.

[20] Immerman, N., Relational queries computable in polynomial time, Information and Control, 68 (1986).

[21] Kaczmarek, T.S., Mark, W., Sondheimer, K., Recent developments in NIKL, in *Proceedings of the 5th National Conference of the American Association for Artificial Intelligence*, AAAI-86, Philadelphia, 1986.

[22] Lassez, J.-L., Nguyen, V.L., Sonenberg, E.A., Fixed point theorems and semantics: A folk tale, Information Processing Letters, 14 (1982).

[23] Levesque, H.J., Brachman, R.J., Expressiveness and tractability in knowledge representation and reasoning, Computational Intelligence, 3 (1987).

[24] Lloyd, J.W., *Logic Programming, Second, Extended Edition,* Springer Verlag, Berlin,

1987.

[25] McNaughton, R., Testing and generating infinite sequences by a finite automaton, Information and Control, 9 (1966).

[26] MacGregor, R., Bates, R., The Loom knowledge representation language, Technical Report ISI/RS-87-188, University of Southern California, Information Science Institute, 1987.

[27] Manna, Z., *Mathematical Theory of Computation*, McGraw-Hill, New York, 1974.

[28] Minker, J., Nicolas, J.M., On recursive axioms in deductive databases, Information Systems, 8 (1983).

[29] Nebel, B., On terminological cycles, KIT Report 58, Technische Universität Berlin, 1987.

[30] Nebel, B., *Reasoning and Revision in Hybrid Representation Systems*, Lecture Notes in Computer Science 422, Springer Verlag, Berlin, 1990.

[31] Nebel, B., Terminological reasoning is inherently intractable, Artificial Intelligence, 43 (1990).

[32] Nebel, B., Terminological cycles: Semantics and computational properties, in *Formal Aspects of Semantic Networks*, Sowa, J. ed., Morgan Kaufmann, San Mateo, 1991.

[33] Patel-Schneider, P., McGuinness, D.L., Brachman, R.J., Alperin Resnick, L., Borgida, A., The CLASSIC knowledge representation system: Guiding principles and implementation issues, SIGART Bulletin, 2 (1991).

[34] Pecuchet, J.P., On the complementation of Büchi automata, Theoretical Computer Science, 47 (1986).

[35] Rosenstein, J.G., *Linear Ordering*, Academic Press, New York, 1982.

[36] Schild, K., A correspondence theory for terminological logics, in *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, IJCAI-91, Sydney, 1991.

[37] Schild, K., Terminological cycles and the propositional μ-calculus, in *Proceedings of the 4th International Conference on Principles of Knowledge Representation and Reasoning*, KR-94, Bonn, 1994.

[38] Schmidt, D.A., *Denotational Semantics: A Methodology for Language Development*, Allyn and Bacon, Boston, 1986.

[39] Sistla, A.P., Vardi, M.Y., Wolper, P., The complementation problem for Büchi automata with applications to temporal logic, Theoretical Computer Science, 49 (1987).

[40] Thomas, W., Automata on infinite objects, in *Handbook of Theoretical Computer Science, Vol. B*, Leeuwen, J. ed., Elsevier Science Publishers, Amsterdam, 1990.

[41] Vardi, M., Complexity of relational query languages, in *Proceedings of the 4th ACM Symposium on the Theory of Computing*, 1982.

[42] Vielle, L., Recursive query processing: The power of logic, Theoretical Computer Science, 69 (1989).

[43] Wu, C.S., Henschen, L.J., Answering linear recursive queries in cyclic databases, in

Proceedings of the International Conference on Fifth Generation Computer Systems, Tokyo, 1988.

Appendix A: Proofs of the Results of Section 5.1

Proposition 19

Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be a gfp-model of T and let A be a concept name occurring in T. For any d ∈ dom(I) we have:

d ∈ A^I iff for all primitive concepts P, all words W ∈ L(A,P) and all individuals e ∈ dom(I): (d,e) ∈ W^I implies e ∈ P^I.

Proof

If A is a primitive concept, then L(A,A) = {ε} and L(A,P) = Ø for A ≠ P. Since $ε^I$ = {(d,d); d ∈ dom(I)}, the proposition follows immediately.

Assume that A is a defined concept. The gfp-model I is given by a primitive interpretation J and the tuple gfp(T_J) = $\cap_{k\geq 0}T_J{}^k$(top). The defined concept A corresponds to a component of this tuple, i.e., A^I = $(\text{gfp}(T_J))_i$ for i = index(A).

(1) Assume that d ∉ A^I. Then there exists k ≥ 0 such that d ∉ $(T_J{}^k(\text{top}))_i$. We proceed by **induction on k**.

For **k = 0,** we have d ∉ $(\text{top})_i$ = dom(I), which is a contradiction.

For **k > 0** we have d ∉ $(T_J(T_J{}^{k-1}(\text{top})))_i$. Let the defining axiom for A be of the form A = … ⊓ ∀W: B ⊓ … and assume that ∀W: B is responsible for d ∉ $(T_J(T_J{}^{k-1}(\text{top})))_i$. This means that there exists e ∈ dom(I) such that dW^Ie and e ∉ B^J = B^I (if B is a primitive concept) or e ∉ $(T_J{}^{k-1}(\text{top}))_j$ (if B is a defined concept and index(B) = j). In the first case, B is a primitive concept and obviously, W ∈ L(A,B). In the second case, we can apply the induction hypothesis to e ∉ $(T_J{}^{k-1}(\text{top}))_j$. Thus there exist a primitive concept P, a word V ∈ L(B,P) and an individual f ∈ dom(I) such that eV^If and f ∉ P^I. But then WV ∈ L(A,P) and $d(WV)^If$. This completes the proof of the “if” direction.

(2) Assume that there exist a primitive concept P, a word W ∈ L(A,P) and an individual e ∈ dom(I) such that dW^Ie and e ∉ P^I. Let W be the label of the (non-empty) path A, U_0, C_1, …, C_{n-1}, U_n, P. Since W = $U_0 \ldots U_n$ and dW^Ie, there are individuals d_1, …, d_{n-1} such that $dU_0{}^Id_1 \ldots d_{n-1}U_n{}^Ie$. We proceed by **induction on n**.

For **n = 0**, W = U_0 and the defining axiom for A is of the form A = … ⊓ ∀W: P ⊓ … . Thus d ∉ $(T_J(\text{top}))_i$.

For **n > 0**, we know by induction that d_1 ∉ $(T^h(\text{top}))_j$ for some h > 0 (where index(C_1) = j). But then d ∉ $(T^{h+1}(\text{top}))_i$. This completes the proof of the proposition since A^I = $(\text{gfp}(T_J))_i$ = $\cap_{k\geq 0}(T_J{}^k(\text{top}))_i$. ❑

As an easy consequence of this proposition one obtains a characterization of subsumption w.r.t gfp-semantics.

Theorem 20
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be a gfp-model of T and let A, B be concept names occurring in T. Subsumption in T can be reduced to inclusion of regular languages defined by $\mathcal{A}_T$. More precisely,

$$A \sqsubseteq_{gfp,T} B \quad \text{iff} \quad L(B,P) \subseteq L(A,P) \text{ for all primitive concepts } P.$$

Proof
(1) Assume that $L(B,P) \not\subseteq L(A,P)$ for some primitive concept P, i.e., there is a word W such that $W \in L(B,P) \setminus L(A,P)$. Let $W = R_1R_2...R_n$ for n (not necessarily different) role names $R_1, R_2, ..., R_n$. We define the primitive interpretation J as follows: $dom(J) := \{d_0, \ldots, d_n\}$; $Q^J := dom(J)$ for all primitive concepts $Q \neq P$; $P^J := dom(J) \setminus \{d_n\}$; $R^J := \{(d_i,d_{i+1});\ 0 \leq i \leq n-1 \text{ and } R = R_{i+1}\}$ for all roles R. The definition of the role extensions implies that $d_0V^Jd_n$ iff $V = W$.
Let I be the gfp-model defined by J. Since $W \in L(B,P)$, $d_0W^Id_n$ and $d_n \notin P^I$, we know by Proposition 19 that $d_0 \notin B^I$. On the other hand, assume that $d_0 \notin A^I$. By Proposition 19, there exists a primitive concept Q, a word $V \in L(A,Q)$ and an individual $f \in dom(I)$ such that d_0V^If and $f \notin Q^I$. The definition of J implies that $Q = P$ and $f = d_n$. But then $d_0V^Id_n$ yields $V = W$. This contradicts our assumption that $W \notin L(A,P)$. Hence we have shown that $d_0 \in A^I \setminus B^I$, which implies that $A \not\sqsubseteq_{gfp,T} B$.
(2) Now assume that $A \not\sqsubseteq_{gfp,T} B$, i.e., there exists a gfp-model I and an individual $d \in dom(I)$ such that $d \in A^I \setminus B^I$. Assume that $L(B,P) \subseteq L(A,P)$ for all primitive concepts P. Since $d \notin B^I$, Proposition 19 says that there exists a primitive concept P, a word $W \in L(B,P)$ and an individual $e \in dom(I)$ such that dW^Ie and $e \notin P^I$. But then $L(B,P) \subseteq L(A,P)$ yields $W \in L(A,P)$ and thus $d \notin A^I$, which is a contradiction. ❑

Appendix B: Proofs of the Results of Section 5.2

In order to obtain a characterization of lfp-semantics that is similar to the characterization of gfp-semantics in Proposition 19, we need two lemmata.

Let J be a primitive interpretation of the terminology T, let A, B be defined concepts in T, and let $\mathcal{A}_T$ be the semi-automaton corresponding to T. The least fixed-point of T_J can be obtained as $lfp(T_J) = T_J\uparrow^{\alpha}$ for some ordinal α. Without loss of generality we may assume that α is a limit ordinal. This means that $lfp(T_J) = \cup_{\lambda<\alpha}T_J\uparrow^{\lambda}$. Let I be the lfp-model of T defined by J. Assume that index(A) = i and index(B) = j, i.e., $A^I = (lfp(T_J))_i$ and $B^I = (lfp(T_J))_j$. For an individual $d \in dom(I)$ we have $d \in A^I$ if and only if there exists $\lambda < \alpha$ such that $d \in (T_J{\uparrow}\lambda)_i$.

Lemma 47
Let ind(A) = i and ind(B) = j, and assume that $d \in (T_J{\uparrow}\lambda)_i$, dW^Ie and that (A,W,B) is a transition of $\mathcal{A}_T$. Then there exists $\gamma < \lambda$ such that $e \in (T_J{\uparrow}\gamma)_j$.

Proof
The lemma is proved by transfinite induction on λ.
(1) For $\lambda = 0$, $(T_J{\uparrow}\lambda)_i = (bottom)_i = \emptyset$. Hence there is no such individual d.
(2) For $\lambda = \delta + 1$, $T_J{\uparrow}\lambda = T_J(T_J{\uparrow}\delta)$. The definition of A in T is of the form A = … ⊓ ∀W: B ⊓ … and we have $d \in (T_J(T_J{\uparrow}\delta))_i$ and dW^Ie. Thus e must be an element of $(T_J{\uparrow}\delta)_j$ and we can take $\gamma = \delta$.
(3) Let λ be a *limit ordinal*. Then $T_J{\uparrow}\lambda = \cup_{\delta<\lambda} T_J{\uparrow}\delta$, and thus $d \in (T_J{\uparrow}\lambda)_i$ iff there exists $\delta < \lambda$ such that $d \in (T_J{\uparrow}\delta)_i$. If we apply the induction hypothesis to δ, we get $\gamma < \delta < \lambda$ such that $e \in (T_J{\uparrow}\gamma)_j$. ❑

Lemma 48
Assume that $d \in (T_J{\uparrow}\lambda)_i$, that dW^Ie, and that $W \in L(A,P)$. Then we have $e \in P^I$.

Proof
The lemma is proved by transfinite induction on λ.
(1) For $\lambda = 0$, there is no such individual d.
(2) For $\lambda = \delta + 1$, $T_J{\uparrow}\lambda = T_J(T_J{\uparrow}\delta)$. Let W be the label of the (non-empty) path A, U_0, C_1, …, C_{n-1}, U_n, P. Since $W = U_0 \ldots U_n$ and dW^Ie, there are individuals $d_1, \ldots, d_{n-1}$ such that $dU_0{}^Id_1 \ldots d_{n-1}U_n{}^Ie$.
For $n = 0$, $W = U_0$ and the defining axiom for A is of the form A = … ⊓ ∀W: P ⊓ … . Thus $d \in (T_J(T_J{\uparrow}\delta))_i$ and dW^Ie imply $e \in P^I$.
For $n > 0$, the defining axiom for A is of the form A = … ⊓ $\forall U_0$: C_1 ⊓ …, and thus $d \in (T_J(T_J{\uparrow}\delta))_i$ and $dU_0{}^Id_1$ imply $d_1 \in (T_J{\uparrow}\delta)_k$ (where the defined concept C_1 has index(C_1) = k). The induction hypothesis for δ yields $e \in P^I$.
(3) Let λ be a *limit ordinal*. Then $T_J{\uparrow}\lambda = \cup_{\delta<\lambda} T_J{\uparrow}\delta$ and thus $d \in (T_J{\uparrow}\lambda)_i$ iff there exists $\delta < \lambda$ such that $d \in (T_J{\uparrow}\delta)_i$. If we apply the induction hypothesis to δ we get $e \in P^I$. ❑

We can now characterize lfp-semantics with the help of finite and infinite paths in the automaton $\mathcal{A}_T$.

Proposition 22
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be the lfp-model of T defined by the primitive interpretation J and let A be a concept

name occurring in T. For any $d_0 \in dom(I)$ we have $d_0 \in A^I$ iff the following two properties hold:

(P1) *For all primitive concepts P, all words $W \in L(A,P)$ and all individuals $e \in dom(I)$, $(d_0,e) \in W^I$ implies $e \in P^I$.*

(P2) *For all infinite paths A, W_1, C_1, W_2, C_2, W_3, C_3, ..., and all individuals d_1, d_2, d_3, ... there exists $n \geq 1$ such that $(d_{n-1},d_n) \notin W_n^I$.*

Proof

The case where A is a primitive concept is trivial. In the following, let A be a defined concept.

(1) Assume that $d_0 \in A^I = (lfp(T_J))_i$. Then there exists an ordinal λ such that $d_0 \in (T_J{\uparrow}^{\lambda})_i$, and thus property (P1) is an immediate consequence of Lemma 48. If (P2) does not hold then there exists an infinite path A, W_1, C_1, W_2, C_2, W_3, C_3, ..., and individuals d_1, d_2, d_3, ... such that $(d_{n-1},d_n) \in W_n^I$ for all $n \geq 1$. By Lemma 47, there exist ordinals $\lambda > \lambda_1 > \lambda_2 > \lambda_3 > \ldots$ such that $d_n \in (T_J{\uparrow}^{\lambda_n})_{j_n}$ (for all $n \geq 1$ and appropriate indices j_n). But there can be no such infinitely decreasing chain of ordinals since the ordering of ordinals is well-founded.

(2) Assume that (P1) and (P2) hold. We define an ordering ">" on 3-tuples of the form (W,d,B) where B is a defined concept, W is the label of a path from A to B,[12] and d is an individual with d_0W^Id. Let $\mathcal{P}$ be the set of all such tuples and let (V,d,B) and (W,e,C) be two elements of $\mathcal{P}$. Then (V,d,B) > (W,e,C) iff W = VU where U is the label of a *non-empty* path from B to C and dU^Ie. Obviously, ">" is a strict partial ordering, and property (P2) ensures that this ordering is well-founded. The following claim will be proved by Noetherian induction[13] on ">".

Claim: For any $(W,d,B) \in \mathcal{P}$ there exists an ordinal $\lambda < \alpha$ such that $d \in (T_J{\uparrow}^{\lambda})_j$ (where index(B) = j).[14]

Proof of the claim. (2.1) Let (W,d,B) be a minimal element of $\mathcal{P}$. Let the defining axiom of B be of the form $B = \ldots \sqcap \forall U\colon C \sqcap \ldots \sqcap \forall V\colon P \ldots$, where P is primitive and C defined. The minimality of (W,d,B) implies that there does not exist an individual e with dU^Ie. Assume that dV^Ie. Since $WV \in L(A,P)$ and $d_0(WV)^Ie$, property (P1) implies $e \in P^I$. This shows that $d \in (T_J(bottom))_j$. Hence we can take $\lambda = 1$.

(2.2) Assume that (W,d,B) is not a minimal element of $\mathcal{P}$. Let the defining axiom of B be of the form $B = \forall U_1\colon C_1 \sqcap \ldots \sqcap \forall U_n\colon C_n \sqcap \ldots \sqcap \forall V\colon P \ldots$, where P is primitive and the C_i are all the defined concepts in the definition of B.

[12]For A = B this may also be the empty path.

[13]See e.g., [13], p. 9, 10, for the definition and justification of Noetherian induction.

[14]Recall that α was a limit ordinal such that $lfp(T_J) = T_J{\uparrow}^{\alpha}$.

As in (2.1) we can show for all individuals e that $dV^I e$ implies $e \in P^I$. Assume that $dU_i^I e$ and index(C_i) = k. We have $(WU_i,e,C_i) \in \mathcal{P}$ and $(W,d,B) > (WU_i,e,C_i)$. Hence, by the induction hypothesis, there is an ordinal $\lambda(i,e) < \alpha$ such that $e \in (T_J\uparrow^{\lambda(i,e)})_k$. We define $\gamma := \sup\{\lambda(i,e);$ where $1 \le i \le n$ and $dU_i^I e\}$. Then we have $\gamma \le \alpha$ and it is easy to see that $d \in (T_J\uparrow^{\gamma+1})_j$. But then $d \in (T_J\uparrow^{\alpha+1})_j$ and since $T_J\uparrow^{\alpha}$ is the fixed-point of T_J, $d \in (T_J\uparrow^{\alpha})_j$. Since α is a limit ordinal, this means that there exists $\lambda < \alpha$ such that we have $d \in (T_J\uparrow^{\lambda})_j$. This completes the proof of the claim. ❑ (Claim)

If we apply the claim to (ε,d_0,A), we get $d_0 \in (T_J\uparrow^{\lambda})_i$ for some $\lambda < \alpha$, and thus $d_0 \in A^I$. ❑

To take the role of ε-cycles into account, the semi-automaton $\mathcal{A}_T$ is modified as follows. We add a new state Q_{loop} to $\mathcal{A}_T$, a transition with label ε from Q_{loop} to Q_{loop}, and for each role R in T a transition with label R from Q_{loop} to Q_{loop}. For any state B of $\mathcal{A}_T$ lying on an ε-cycle, we add a transition with label ε from B to Q_{loop}, and for any primitive concept P we add a transition with label ε from Q_{loop} to P. This modified semi-automaton is called $\mathcal{B}_T$.

For all concepts B we thus have $L_{\mathcal{B}_T}(B,P) = L_{\mathcal{A}_T}(B,P) \cup \{UV;$ U is a *finite* word in $U_{\mathcal{A}_T}(B)$ and $V \in \Sigma^*\}$ and $U_{\mathcal{B}_T}(B) = U_{\mathcal{A}_T}(B) \cup \{UV;$ U is a *finite* word in $U_{\mathcal{A}_T}(B)$ and $V \in \Sigma^* \cup \Sigma^\omega\}$.

Theorem 24
Let T be a terminology and let $\mathcal{B}_T$ be the corresponding modified automaton. Then $A \sqsubseteq_{lfp,T} B$ iff $U_{\mathcal{B}_T}(B) \subseteq U_{\mathcal{B}_T}(A)$ and $L_{\mathcal{B}_T}(B,P) \subseteq L_{\mathcal{B}_T}(A,P)$ for all primitive concepts P.

Proof
The proof is structured as follows: In part (1) we show that $L_{\mathcal{B}_T}(B,P) \not\subseteq L_{\mathcal{B}_T}(A,P)$ implies $A \not\sqsubseteq_{lfp,T} B$. In part (2) and (3), the same is shown for the case $U_{\mathcal{B}_T}(B) \not\subseteq U_{\mathcal{B}_T}(A)$. In part (2) we assume that an infinite word is responsible for $U_{\mathcal{B}_T}(B) \not\subseteq U_{\mathcal{B}_T}(A)$, and in part (3) we assume that this is due to a finite word. This will establish the "only if" direction of the theorem. Part (4) of the proof is devoted to the "if" direction.

(1) Assume that $L_{\mathcal{B}_T}(B,P) \not\subseteq L_{\mathcal{B}_T}(A,P)$, i.e., there is a word $W = R_1\ldots R_n$ such that $W \in L_{\mathcal{B}_T}(B,P) \setminus L_{\mathcal{B}_T}(A,P)$. The primitive interpretation J is defined as follows: dom(J) := $\{d_0, \ldots, d_n\}$; Q^J := dom(J) for all primitive concepts $Q \neq P$; P^J := dom(J) \ $\{d_n\}$; $R^J := \{(d_i,d_{i+1}); 0 \le i \le n-1$ and $R = R_{i+1}\}$ for all roles R. The definition of the roles implies that $d_0V^Jd_n$ iff V = W. Let I be the lfp-model defined by J.

(1.1) If $W \in L_{\mathcal{A}_T}(B,P)$, then $d_0W^Id_n$ and $d_n \notin P^I$ imply that $d_0 \notin B^I$ because

(P1) of Proposition 22 is not satisfied. If $W \in L_{\mathcal{B}_T}(B,P) \setminus L_{\mathcal{A}_T}(B,P)$, then $W = UV$ where $U \in U_{\mathcal{A}_T}(B) \cap \Sigma^*$ is the label of a path in $\mathcal{A}_T$ from B to a concept C that lies on an ε-cycle in $\mathcal{A}_T$. Since $d_0U^Id_k$ for some $k \leq n$ and $d_k\varepsilon^Id_k\varepsilon^Id_k\ldots$, property (P2) of Proposition 22 is not satisfied, which yields $d_0 \notin B^I$.

(1.2) On the other hand, assume that $d_0 \notin A^I$. By Proposition 22, (P1) or (P2) is not satisfied. In the first case, there exist a primitive concept Q, a word $V \in L_{\mathcal{A}_T}(A,Q)$ and an individual $f \in dom(I)$ such that d_0V^If and $f \notin Q^I$. The definition of J implies that Q = P and $f = d_n$. But then $d_0V^Id_n$ yields V = W. This contradicts our assumption that $W \notin L_{\mathcal{B}_T}(A,P)$ since $L_{\mathcal{A}_T}(A,P) \subseteq L_{\mathcal{B}_T}(A,P)$. In the second case, there exists an infinite path A, W_1, C_1, W_2, C_2, W_3, C_3, ... in $\mathcal{A}_T$ and individuals $e_0 = d_0$, e_1, e_2, e_3, ... such that $(e_{m-1},e_m) \in W_m{}^I$ for all m > 0. The definition of J implies that there exists $k \geq 0$ such that $W_1\ldots W_k$ is a prefix of W and $W_{k+1} = W_{k+2} = \ldots = \varepsilon$. This means that C_k is inconsistent, and thus by the definition of $\mathcal{B}_T$, $W_1\ldots W_kU$ is in $L_{\mathcal{B}_T}(A,P)$ for all words U. In particular, this yields $W \in L_{\mathcal{B}_T}(A,P)$, which is a contradiction.

Hence we have shown that $d_0 \in A^I \setminus B^I$, which implies that $A \not\sqsubseteq_{lfp,T} B$.

(2) Assume that $U_{\mathcal{B}_T}(B) \not\subseteq U_{\mathcal{B}_T}(A)$ because there exists an infinite word $W = R_1R_2R_3\ldots$ such that $W \in U_{\mathcal{B}_T}(B) \setminus U_{\mathcal{B}_T}(A)$. The primitive interpretation J is defined as follows: dom(J) := $\{d_0, d_1, d_2, \ldots\}$; P^J := dom(J) for all primitive concepts P; $R^J := \{(d_i,d_{i+1}); i \geq 0 \text{ and } R = R_{i+1}\}$ for all roles R. Let I be the lfp-model defined by J.

(2.1) If $W \in U_{\mathcal{A}_T}(B)$, then it is the label of an infinite path B, W_1, C_1, W_2, C_2, W_3, C_3, ... in $\mathcal{A}_T$. Obviously, (P2) of Proposition 22 is not satisfied for d_0 and B, which yields $d_0 \notin B^I$. If $W \in U_{\mathcal{B}_T}(B) \setminus U_{\mathcal{A}_T}(B)$, then W has a finite initial segment U that is the label of a finite path in $\mathcal{A}_T$ from B to a concept C that lies on an ε-cycle in $\mathcal{A}_T$. As in part (1.1) of the proof, we can deduce $d_0 \notin B^I$.

(2.2) On the other hand, assume that $d_0 \notin A^I$. By Proposition 22, (P1) or (P2) is not satisfied. Since we have defined P^J := dom(J) for all primitive concepts P, (P1) is always satisfied. Thus (P2) does not hold, i.e., there exist an infinite path A, W_1, C_1, W_2, C_2, W_3, C_3, ... in $\mathcal{A}_T$ and individuals $e_0 = d_0$, e_1, e_2, e_3, ... such that $(e_{n-1},e_n) \in W_n{}^I$ for all n > 0. If the label $W_1W_2W_3\ldots$ of this infinite path is an infinite word, the definition of J implies that it is equal to W. Hence $W \in U_{\mathcal{A}_T}(A)$, which contradicts our assumption that $W \notin U_{\mathcal{B}_T}(A)$. If the label $W_1W_2W_3\ldots$ of the infinite path is a finite word U, the definition of J implies that U is a finite initial segment of W. By the definition of $\mathcal{B}_T$, $UV \in U_{\mathcal{B}_T}(A)$ for all infinite words $V \in \Sigma^\omega$. Hence $W \in U_{\mathcal{B}_T}(A)$, which is a contradiction.

Thus we have shown that $d_0 \in A^I \setminus B^I$, which implies that $A \not\sqsubseteq_{lfp,T} B$.

(3) Assume that $U_{\mathcal{B}_T}(B) \not\subseteq U_{\mathcal{B}_T}(A)$ because there exists a finite word W such that $W \in U_{\mathcal{B}_T}(B) \setminus U_{\mathcal{B}_T}(A)$. From $W \in U_{\mathcal{B}_T}(B)$ we can deduce that there is a prefix $U = R_1\ldots R_n$ of W and a path with label U in $\mathcal{A}_T$ from B to a concept C that lies on an ε-cycle in $\mathcal{A}_T$. The primitive interpretation J is defined as follows: $dom(J) := \{d_0, d_2, \ldots, d_n\}$; $P^J := dom(J)$ for all primitive concepts P; $R^J := \{(d_i,d_{i+1}); 0 \leq i \leq n-1$ and $R = R_{i+1}\}$ for all roles R. Let I be the lfp-model defined by J.

(3.1) Obviously, the pair d_0, B doesn't satisfy (P2) of Proposition 22, and thus $d_0 \notin B^I$.

(3.2) On the other hand, assume that $d_0 \notin A^I$. As in part (2.2) of the proof we can deduce that (P2) does not hold, i.e., that there there exist an infinite path A, W_1, C_1, W_2, C_2, W_3, C_3, ... in $\mathcal{A}_T$ and individuals $e_0 = d_0$, e_1, e_2, e_3, ... such that $(e_{m-1},e_m) \in W_m{}^I$ for all $m > 0$. The definition of J implies that there exists $k \geq 0$ such that $W_1 \ldots W_k$ is a prefix of U and $W_{k+1} = W_{k+2} = \ldots = \varepsilon$. This means that C_k is inconsistent, and thus by the definition of $\mathcal{B}_T$, $W_1 \ldots W_k V$ is in $U_{\mathcal{B}_T}(A)$ for all words $V \in \Sigma^*$. In particular, $W \in U_{\mathcal{B}_T}(A)$, which is a contradiction.

Thus we have shown that $d_0 \in A^I \setminus B^I$, which implies that $A \not\sqsubseteq_{lfp,T} B$.

(4) Let $U_{\mathcal{B}_T}(B) \subseteq U_{\mathcal{B}_T}(A)$, and $L_{\mathcal{B}_T}(B,P) \subseteq L_{\mathcal{B}_T}(A,P)$ for all primitive concepts P. Assume that $A \not\sqsubseteq_{gfp,T} B$, i.e., there exist a lfp-model I of T and an individual $d_0 \in dom(I)$ such that $d_0 \in A^I \setminus B^I$. Now $d_0 \notin B^I$ implies that (P1) or (P2) of Proposition 22 does not hold for d_0, B.

(4.1) If (P1) does not hold, then there exist a primitive concept P, a word $W \in L_{\mathcal{A}_T}(B,P)$, and an individual $e \in dom(I)$ such that $d_0 W^I e$ and $e \notin P^I$. Since $L_{\mathcal{A}_T}(B,P) \subseteq L_{\mathcal{B}_T}(B,P) \subseteq L_{\mathcal{B}_T}(A,P)$, we have $W \in L_{\mathcal{B}_T}(A,P)$. For $W \in L_{\mathcal{A}_T}(A,P)$, Proposition 22 yields $d_0 \notin A^I$, which is a contradiction. Assume that $W \in L_{\mathcal{B}_T}(A,P) \setminus L_{\mathcal{A}_T}(A,P)$. This means that $W = UV$, and there is a path with label U in $\mathcal{A}_T$ from A to a concept C that lies on an ε-cycle. Now $d_0 W^I e$ implies that there exists an individual f such that $d_0 U^I f$. Since $f \varepsilon^I f \varepsilon^I f \ldots$, property (P2) of Proposition 22 is not satisfied. This yields $d_0 \notin A^I$, which is a contradiction.

(4.2) If (P2) does not hold, then there exist an infinite path B, W_1, C_1, W_2, C_2, W_3, C_3, ... in $\mathcal{A}_T$ and individuals d_1, d_2, d_3, ... such that $(d_{n-1},d_n) \in W_n{}^I$ for all $n > 0$.

(4.2.1) First, we assume that the label $W_1W_2W_3$... of this path is an infinite word W. Then we have $W \in U_{\mathcal{B}_T}(B) \subseteq U_{\mathcal{B}_T}(A)$. If $W \in U_{\mathcal{A}_T}(A)$, we immediately get $d_0 \notin A^I$, which is a contradiction. If $W \in U_{\mathcal{B}_T}(A) \setminus U_{\mathcal{A}_T}(A)$, then there exists a finite initial segment U of W such that there is a path with label U in $\mathcal{A}_T$ from A to a concept C that lies on an ε-cycle. As in (4.1) this implies $d_0 \notin A^I$. This contradicts our assumption.

(4.2.2) Assume that the label $W_1W_2W_3$... of the infinite path B, W_1, C_1, W_2, C_2, W_3, C_3, ... is a finite word W. We have $W \in U_{\mathcal{B}_T}(B) \subseteq U_{\mathcal{B}_T}(A)$. But $W \in U_{\mathcal{B}_T}(A)$ means that there exists a prefix U of W such that there is a path with label U in $\mathcal{A}_T$ from A to a concept C that lies on an ε-cycle. As in (4.1) this

implies $d_0 \notin A^I$, which is a contradiction.
This completes the proof of the theorem. ❑

Appendix C: Proofs of the Results of Section 5.3

Proposition 28
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let J be a primitive interpretation and let $\underline{A}$ be a tuple such that $T_J(\underline{A}) \subseteq \underline{A}$. Let I be the model of T defined by J and the tuple $\underline{A}$-gfp(T_J) (see Corollary 4).
For any concept A and any individual $d \in dom(I)$ we have: $d \in A^I$ iff the following two properties hold:
(1) *For all primitive concepts P, all words $W \in L(A,P)$, and all individuals $e \in dom(I)$, $(d,e) \in W^I$ implies $e \in P^I$.*
(2) *For all defined concepts B, all words $W \in L(A,B)$, and all individuals $e \in dom(I)$, $(d,e) \in W^I$ implies $e \in (\underline{A})_j$ (where j = index(B)).*

Proof
The case where A is a primitive concept is trivial (see the proof of Proposition 19). Let A be a defined concept and let i = index(A), i.e., $A^I = (\underline{A}\text{-gfp}(T_J))_i$. We know that $\underline{A}\text{-gfp}(T_J) = \cap_{k\geq 0} T_J^k(\underline{A})$.
(1) Assume that $d \notin A^I$. Then there exists $k \geq 0$ such that $d \notin (T_J^k(\underline{A}))_i$. We proceed by **induction on k**.
For **k = 0** we have $d \notin (\underline{A})_i$, $d\varepsilon^I d$ and $\varepsilon \in L(A,A)$.
For **k > 0** we have $d \notin (T_J(T_J^{k-1}(\underline{A})))_i$. Let the defining axiom for A be of the form $A = \ldots \sqcap \forall W{:}\ C \sqcap \ldots$, and assume that $\forall W{:}\ C$ is responsible for $d \notin (T_J(T_J^{k-1}(\underline{A})))_i$. This means that there exists $e \in dom(I)$ such that dW^Ie and $e \notin C^J = C^I$ (if C is a primitive concept) or $e \notin (T_J^{k-1}(\underline{A}))_m$ (if C is a defined concept and index(C) = m). In the first case, C is a primitive concept, and obviously $W \in L(A,C)$. In the second case, we can apply the induction hypothesis to $e \notin (T_J^{k-1}(\underline{A}))_m$. Thus there exist a primitive concept P (resp. a defined concept B with index j), a word $V \in L(C,P)$ (resp. $V \in L(C,B)$) and an individual $f \in dom(I)$ such that eV^If and $f \notin P^I$ (resp. $f \notin (\underline{A})_j$). But then $WV \in L(A,P)$ (resp. $WV \in L(A,B)$) and $d(WV)^If$. This completes the proof of the “if” direction.
(2) Assume that (1) or (2) does not hold. Then $d \notin A^I$ follows as in the proof of Proposition 19. ❑

For subsumption with respect to descriptive semantics, the not only the labels of infinite paths are important, but also the states that are reached infinitely often.

Theorem 29
Let T be a terminology and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let A, B be concepts in T. Then we have $A \sqsubseteq_T B$ iff the following two properties hold:
(P1) *For all primitive concepts P, $L(B,P) \subseteq L(A,P)$ holds.*
(P2) *For all defined concepts C and all infinite paths of the form B, U_0, C, U_1, C, U_2, C, ..., there exists $k \geq 0$ such that $U_0 \ldots U_k \in L(A,C)$.*

Proof
(1) Assume that (P1) and (P2) hold. Let I be a model of T defined by the primitive interpretation J and a fixed-point $\underline{A}$ of T_J. Obviously, $T_J(\underline{A}) \subseteq \underline{A}$ and $\underline{A} = \underline{A}\text{-gfp}(T_J)$. Let d be an individual such that $d \notin B^I$. We have to show that $d \notin A^I$. By Proposition 28, $d \notin B^I$ means that (1) or (2) of the proposition does not hold.
(1.1) Let P be a primitive concept, $W \in L(B,P)$ be a word and let $e \in dom(I)$ be a individual such that $(d,e) \in W^I$ and $e \notin P^I$. By (P1), $W \in L(A,P)$ and thus Proposition 28 yields $d \notin A^I$.
(1.2) Let C_1 be a defined concept, $W_1 \in L(B,C_1)$ be a word and let $e_1 \in dom(I)$ be a individual such that $(d,e_1) \in W_1^I$ and $e_1 \notin (\underline{A})_{i_1}$ (where $i_1 = index(C_1)$). Since I is the model defined by J and $\underline{A}$, $(\underline{A})_{i_1} = C_1^I$ and we can proceed with C_1 in place of A.
Assume that we have already obtained a sequence $C_1, W_1, e_1, \ldots, C_k, W_k, e_k$ such that $e_i \notin C_i^I$, $e_{i-1}W_i^I e_i$ and $W_i \in L(C_{i-1},C_i)$ for $1 \leq i \leq n$ (where $e_0 := d$ and $C_0 := B$). By Proposition 28, $e_k \notin C_k^I$ means that (1) or (2) of the proposition does not hold.
If (1) does not hold we get a primitive concept, a word $W \in L(C_k,P)$ and an individual $e \in dom(I)$ such that $(e_k,e) \in W^I$ and $e \notin P^I$. But then $W_1 \ldots W_k W \in L(B,P) \subseteq L(A,P)$, $e \notin P^I$ and $d(W_1 \ldots W_k W)^I e$ imply $d \notin A^I$.
If (2) does not hold we get e_{k+1}, C_{k+1} such that $e_{k+1} \notin C_{k+1}^I$, $e_k W_{k+1}^I e_{k+1}$ and $W_{k+1} \in L(C_k,C_{k+1})$.
If this second case holds for all k we get an infinite path B, W_1, C_1, W_2, C_2, W_3, C_3, ... and corresponding individuals $e_1, e_2, e_3, \ldots$ with the above described properties. But then there is a concept C such that $C = C_i$ for infinitely many indices i. This means that the above path is of the form B, U_0, C, U_1, C, U_2, C, ... By property (P2), there exists $k \geq 0$ such that $U_0 \ldots U_k \in L(A,C)$. In addition, we know that there is an individual e_m such that $d(U_0 \ldots U_k)^I e_m$ and $e_m \notin C^I = (\underline{A})_j$ (where j = index(C)). Thus Proposition 28 yields $d \notin A^I$.
(2) Assume that $A \sqsubseteq_T B$. This implies $A \sqsubseteq_{gfp,T} B$ and thus, by Theorem 20, property (P1) holds. Now assume that (P2) does not hold, i.e., there exists an infinite path of the form B, U_0, C, U_1, C, U_2, C, ... such that $U_0 \ldots U_k \notin L(A,C)$ for all $k \geq 0$.

The primitive interpretation J is defined as follows: If U := $U_0U_1U_2\ldots$ is an infinite word $R_1R_2R_3\ldots$, then dom(J) := $\{d_0, d_1, d_2, \ldots\}$; P^J := dom(J) for all primitive concepts P; $R^J := \{(d_{i-1},d_i);\ i \geq 1$ and $R = R_i\}$ for all roles R. If U := $U_0U_1U_2\ldots$ is a finite word $R_1R_2\ldots R_s$ then dom(J) := $\{d_0, d_1, \ldots, d_s\}$; P^J := dom(J) for all primitive concepts P; $R^J := \{(d_{i-1},d_i);\ 1 \leq i \leq s$ and $R = R_i\}$ for all roles R.

Let $j_1 \leq j_2 \leq \ldots$ be the indices such that $d_0U_0{}^Jd_{j_1}U_1{}^Jd_{j_2}U_2{}^J\ldots$ .The tuple $\underline{A}$ is defined as follows: Let D be a defined concept in T and m = index(D). Then

$(\underline{A})_m$:= dom(J) \ {e; There exist finite words W, V and an index $k \geq 0$ such that $WV = U_0\ldots U_k$, $W \in L(B,D)$, $V \in L(D,C)$, d_0W^Je and $eV^Jd_{j_{k+1}}$}.

Claim: $T_J(\underline{A}) \subseteq \underline{A}$.

Proof of the claim. Let D be a defined concept in T and m = index(D). Assume that $e \notin (\underline{A})_m$. We have to show that $e \notin (T_J(\underline{A}))_m$.

By the definition of $\underline{A}$, $e \notin (\underline{A})_m$ means that there exist finite words W, V and an index $k \geq 0$ such that $WV = U_0\ldots U_k$, $W \in L(B,D)$, $V \in L(D,C)$, d_0W^Je and $eV^Jd_{j_{k+1}}$. Without loss of generality we may assume that the path from D to C is not empty.[15] Thus $V = V_1V_2$, there exists an individual e' with $eV_1{}^Je'$ and $e'V_2{}^Jd_{j_{k+1}}$, and the defining axiom for D is of the form $D = \ldots \sqcap \forall V_1{:}\ D' \sqcap \ldots$ Let m' be the index of D'. The definition of $\underline{A}$ yields $e' \notin (\underline{A})_{m'}$ and thus $e \notin (T_J(\underline{A}))_m$. ❑ (Claim)

Let I be the model of T defined by J and $\underline{A}$-gfp(T_J). Let j be the index of B, i.e., $B^I = (\underline{A}\text{-gfp}(T_J))_j$. We have $d_0\varepsilon^Jd_0$, $d_0U_0{}^Jd_{j_1}$ and $\varepsilon \in L(B,B)$, $U_0 \in L(B,C)$. This shows that $d_0 \notin (\underline{A})_j$ and thus $d_0 \notin (\underline{A}\text{-gfp}(T_J))_j = B^I$.

Assume that $d_0 \notin A^I$. Because all primitive concepts have dom(I) as extension, Proposition 28 implies that there exist a defined concepts D, a word $U \in L(A,D)$ and an individual $e \in$ dom(I) such that d_0U^Ie and $e \notin (\underline{A})_m$ (where m = index(C)). Thus, by definition of $\underline{A}$, there are finite words W, V and an index $k \geq 0$ such that $WV = U_0\ldots U_k$, $W \in L(B,D)$, $V \in L(D,C)$, d_0W^Je and $eV^Jd_{j_{k+1}}$. But d_0U^Je and d_0W^Je imply U = W (by the definition of the role extensions in J). This shows that $UV = WV = U_0\ldots U_k$ is an element of L(A,C). This contradicts our assumption that (P2) does not hold. ❑

[15]Otherwise we could take $U_0...U_{k+1}$ instead of $U_0...U_k$.

Appendix D: Proof of Theorem 36

Theorem 36
Let T be a terminology of $\mathcal{FL}^{-}$, and let $\mathcal{A}_T$ be the corresponding semi-automaton. Let I be a gfp-model of T and let A, B be concept names occurring in T. Then we have:

$A \sqsubseteq_{gfp,T} B$ *iff* $L(B,P) \subseteq L(A,P)$ *for all primitive concepts P in T, and* $L(B,\exists R) \subseteq L(A,\exists R)$ *for all terms* $\exists R$ *occurring in T.*

Proof
(1) Assume that $L(B,P) \not\subseteq L(A,P)$ for some primitive concept P, i.e., there is a word W such that $W \in L(B,P) \setminus L(A,P)$. Let $W = R_1R_2...R_n$ for n (not necessarily different) role names $R_1, R_2, ..., R_n$. We define the primitive interpretation J as follows: $dom(J) := \{d_0, \ldots, d_n, e\}$; $Q^J := dom(J)$ for all primitive concepts $Q \neq P$; $P^J := dom(J) \setminus \{d_n\}$; $R^J := \{(d_i,d_{i+1});\ 0 \leq i \leq n{-}1 \text{ and } R = R_{i+1}\} \cup \{(d_i,e);\ 0 \leq i \leq n\} \cup \{(e,e)\}$ for all roles R. The definition of the role extensions implies that $d_0V^Jd_n$ iff V = W, and that $(\exists R)^J = dom(J)$ for all roles R.
Let I be the gfp-model defined by J. As in part (1) of the proof of Theorem 20 one can show that $d_0 \in A^I \setminus B^I$. This implies that $A \not\sqsubseteq_{gfp,T} B$.
(2) Assume that $L(B,\exists R) \not\subseteq L(A,\exists R)$ for some term $\exists R$ in T, i.e., there is a word W such that $W \in L(B,\exists R) \setminus L(A,\exists R)$. Let $W = R_1R_2...R_n$ for n (not necessarily different) role names $R_1, R_2, ..., R_n$. We define the primitive interpretation J as follows: $dom(J) := \{d_0, \ldots, d_n, e\}$; $P^J := dom(J)$ for all primitive concepts P; $S^J := \{(d_i,d_{i+1});\ 0 \leq i \leq n{-}1 \text{ and } S = R_{i+1}\} \cup \{(d_i,e);\ 0 \leq i \leq n\} \cup \{(e,e)\}$ for all roles $S \neq R$; $R^J := \{(d_i,d_{i+1});\ 0 \leq i \leq n{-}1 \text{ and } R = R_{i+1}\} \cup \{(d_i,e);\ 0 \leq i \leq n{-}1\} \cup \{(e,e)\}$. The definition of the role extensions implies that $d_0V^Jd_n$ iff V = W, that $(\exists S)^J = dom(J)$ for all roles $S \neq R$, and that $(\exists R)^J = dom(J) \setminus \{d_n\}$.
Let I be the gfp-model defined by J. Since $W \in L(B,\exists R)$, $d_0W^Id_n$ and $d_n \notin (\exists R)^I$, we know by Proposition 35 that $d_0 \notin B^I$. On the other hand, assume that $d_0 \notin A^I$. Since $P^I = dom(I)$ for all primitive concepts P, and $(\exists S)^J = dom(J)$ for all roles $S \neq R$, Proposition 35 implies that there exists a word $V \in L(A,\exists R)$, and an individual $f \in dom(I)$ such that d_0V^If and $f \notin (\exists R)^I$. By definition of J, we get $f = d_n$, and thus V = W. This contradicts our assumption that $W \notin L(A,\exists R)$. Hence we have shown that $d_0 \in A^I \setminus B^I$, which implies that $A \not\sqsubseteq_{gfp,T} B$.
(3) The proof of the "if" direction is similar to part (2) of the proof of Theorem 20. ❑

Appendix E: Proof of Theorem 43

Theorem 43
Let T be a T-box of $\mathcal{FL}_0$ and A be an A-box defined over the same sets of concept and role names. Let b be an individual name in A and B be a concept name in T. Then $b \in_{gfp,T\cup A} B$ iff for all primitive concepts P, and all words $W \in L_{\mathcal{A}_T}(B,P)$ there exist concepts E, F, a word U, and an individual name f such that

(1) $W \in L_{\mathcal{A}_T}(E,P)$,

(2) $U \in L_{\mathcal{B}_{T\cup A}}((F,f),(E,b))$ and F(f) is an axiom in A.

Proof

(1) Assume that there is a primitive concept P and a word $W = R_1 \ldots R_k \in L_{\mathcal{A}_T}(B,P)$ such that there do not exist E, F, U, f satisfying (1) and (2) of the theorem. Let M be a gfp-model of $T \cup A$, and $b^M =: e_0 \in dom(M)$. We want to construct a gfp-model I of $T \cup A$ such that $b^I \notin B^I$.

(1.1) Without loss of generality we may assume that $R^M = \{(c^M,d^M); R(c,d) \in A\}$ for all roles R. This is true because making role extensions smaller only makes concept extensions larger w.r.t. gfp-semantics. Hence all axioms of the form C(e) remain satisfied if we restrict the role extensions to $\{(c^M,d^M); R(c,d) \in A\}$.

(1.2) The primitive interpretation J is defined as follows: $dom(J) := dom(M) \cup \{e_1, \ldots, e_k\}$ where $e_1, \ldots, e_k$ are new individuals; $R^J := R^M \cup \{(e_{i-1},e_i); 1 \le i \le k \text{ and } R = R_i\}$ for all roles R; $Q^J := Q^M \cup \{e_1, \ldots, e_k\}$ for all primitive concepts $Q \neq P$; $P^J := P^M \cup \{e_1, \ldots, e_{k-1}\}$. Let I be the gfp-model of T defined by J. The interpretation I of T is extended to an interpretation I of $T \cup A$ by defining $c^I := c^M$ for all individual names c.

Obviously, $b^I W^I e_k$, $W \in L_{\mathcal{A}_T}(B,P)$, and $e_k \notin P^I$ imply $e_0 = b^I \notin B^I$.

(1.3) It remains to be shown that I is in fact a gfp-model of $T \cup A$. Obviously, $(c^I,d^I) \in R^I$ for all axioms R(c,d) in A. Assume that F(f) is an axiom of A, but $f^I \notin F^I$. By Proposition 19, there exist a primitive concept Q, a word $U \in L_{\mathcal{A}_T}(F,Q)$, and an individual e such that $f^I U^I e$ and $e \notin Q^I$.

If $f^I U^I e$ does not use some e_i ($i \ge 1$) as intermediate individual, then we also have $f^M U^M e$ and $e \notin Q^M$. Hence $f^M \notin F^M$, which contradicts our assumption that M is a model of $T \cup A$.

Otherwise, the definition of the role extensions implies that $U = U_1 U_2$, $f^I U_1{}^I e_0 U_2{}^I e$ and $e = e_i$ for some $i \ge 1$. But now $e \notin Q^I$ yields $Q = P$, $e = e_k$, and $U_2 = W$. Because $U = U_1 W \in L_{\mathcal{A}_T}(F,P)$, there exists a state E of $\mathcal{A}_T$ such that $U_1 \in L_{\mathcal{A}_T}(F,E)$ and $W \in L_{\mathcal{A}_T}(E,P)$. In addition, $f^I U_1{}^I e_0$ implies $f^M U_1{}^M e_0 = b^I$, and thus, by (1.1), we have $U_1 \in L_{\mathcal{A}_A}(f,b)$. This shows that $U_1 \in L_{\mathcal{B}_{T\cup A}}((F,f),(E,b))$. But then E, F, U_1, f satisfy (1) and (2) of the theorem. This

contradicts our assumption.
(2) Assume that $b \notin_{gfp,T\cup A} B$, but the right hand side of the theorem holds. Let I be a gfp-model of $T \cup A$ such that $b^I \notin B^I$. By Proposition 19, there exist a primitive concept P, a word $W \in L_{\mathcal{A}_T}(B,P)$, and an individual e such that $b^I W^I e$ and $e \notin P^I$. For $W \in L_{\mathcal{A}_T}(B,P)$ there exist concepts E, F, a word U, and an individual name f satisfying (1) and (2) of the theorem. But then $U \in L_{\mathcal{B}_{T\cup A}}((F,f),(E,b))$ and $W \in L_{\mathcal{A}_T}(E,P)$ yield $UW \in L_{\mathcal{A}_T}(F,P)$ and $f^I U^I b^I$.[16] Thus we have $UW \in L_{\mathcal{A}_T}(F,P)$, $f^I (UW)^I e$, and $e \notin P^I$. This means that $f^I \notin F^I$, which contradicts our assumption that I was model of $T \cup A$ since F(f) is an axiom in A. ❑

[16]Since I is a model of $T \cup A$, $U \in L_{\mathcal{A}_A}(f,b)$ implies $f^I U^I b^I$.