Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

Research
Report
RR-90-01

Terminological Cycles in KL-ONE-based Knowledge Representation Languages

Franz Baader

Januar 1990

Deutsches Forschungszentrum für Künstliche Intelligenz GmbH

Postfach 20 80
D-6750 Kaiserslautern
Tel.: (+49 631) 205-3211/13
Fax: (+49 631) 205-3210

Stuhlsatzenhausweg 3
D-6600 Saarbrücken 11
Tel.: (+49 681) 302-5252
Fax: (+49 681) 302-4161

Deutsches Forschungszentrum
für
Künstliche Intelligenz

The German Research Center for Artificial Intelligence (Deutsches Forschungszentrum für Künstliche Intelligenz, DFKI) with sites in Kaiserslautern und Saarbrücken is a non-profit organization which was founded in 1988 by the shareholder companies ADV/Orga, AEG, IBM, Insiders, Fraunhofer Gesellschaft, GMD, Krupp-Atlas, Mannesmann-Kienzle, Nixdorf, Philips and Siemens. Research projects conducted at the DFKI are funded by the German Ministry for Research and Technology, by the shareholder companies, or by other industrial contracts.

The DFKI supports application-oriented basic research in the field of artificial intelligence and other related subfields of computer science. The overall goal is to provide "Intelligent Special Purpose Systems" which - by using AI methods - implement a problem solution for a selected application area. Currently, there are the following research areas at the DFKI:

- ❑ Utilisation of technical knowledge for expert systems
- ❑ Knowledge-based information presentation
- ❑ Foundations of knowledge representation and knowledge processing
- ❑ Automatic document analysis
- ❑ Planbased help-systems, and
- ❑ Natural-language dialogue systems for cooperating agents.

The DFKI strives at making its research results available to the scientific community. There is a network of contacts to domestic and foreign research institutions, both in academy and industry. The DFKI hosts technology transfer workshops for shareholders and other interested groups in order to inform about the current state of research.

From its beginning, the DFKI has provided an attractive working environment for AI researchers from Germany and from all over the world. The goal is to have a staff of about 100 researchers at the end of the building-up phase.

Prof. Dr. Gerhard Barth
Director

Terminological Cycles in KL-ONE-based Knowledge Representation Languages

Franz Baader

DFKI-RR-90-01

Terminological Cycles in KL-ONE-based Knowledge Representation Languages

Franz Baader
German Research Center for Artificial Intelligence
Postfach 2080
D-6750 Kaiserslautern, West Germany

Abstract

Cyclic definitions are often prohibited in terminological knowledge representation languages, because, from a theoretical point of view, their semantics is not clear and, from a practical point of view, existing inference algorithms may go astray in the presence of cycles. In this paper we consider terminological cycles in a very small KL-ONE-based language. For this language, the effect of the three types of semantics introduced by Nebel (1987, 1989, 1989a) can be completely described with the help of finite automata. These descriptions provide a rather intuitive understanding of terminologies with cyclic definitions and give insight into the essential features of the respective semantics. In addition, one obtains algorithms and complexity results for subsumption determination. The results of this paper may help to decide what kind of semantics is most appropriate for cyclic definitions, not only for this small language, but also for extended languages. As it stands, the greatest fixed-point semantics comes off best. The characterization of this semantics is easy and has an obvious intuitive interpretation. Furthermore, important constructs – such as value-restriction with respect to the transitive or reflexive-transitive closure of a role – can easily be expressed.

Table of Contents

1. Introduction

Cyclic definitions are prohibited in most terminological knowledge representation languages (e.g., in KRYPTON (Brachman et al. (1985)), NIKL (Kaczmarek et al. (1986)) or LOOM (MacGregor-Bates (1987))) for the following reasons. From a theoretical point of view, it is not obvious how to define the semantics of terminological cycles. But even if we have fixed a semantics it is not easy to obtain the corresponding inference algorithms.

On the other hand, cyclic definitions may be very useful and intuitive, e.g., if we want to express the transitive closure of roles (i.e., binary relations). For a role child, value-restrictions with respect to its transitive closure off-spring can be expressed by cyclic concept definitions if we take the appropriate semantics. For the same reason, recursive axioms are considered in data base research (see e.g., Aho-Ullman (1979), Immerman (1982), Vardi (1982), Minker-Nicolas (1983), Wu-Henschen (1988) and Vielle (1989)). Aho-Ullman (1979) showed that the transitive closure of relations cannot be expressed in the relational calculus, which is a standard relational query language. They proposed to add cyclic definitions which 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., Gurevich-Shelah (1985,1986)).

A thorough investigation of cycles in terminological knowledge representation languages can be found in Nebel (1987,1989,1989a). Nebel considered three different kinds of semantics for cyclic definitions in his language $\mathcal{NTF}$; namely, least fixed-point semantics, greatest fixed-point semantics, and what he called descriptive semantics. But, due to the fact that this language is relatively strong[1], it does not provide a deep insight into the meaning of cycles with respect to these three types of semantics. For the two fixed-point semantics, Nebel explicates 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" (Nebel (1989a), p. 124) – is treated more thoroughly. But even in this case the results are not quite satisfactory. For example, the decidability of subsumption determination is proved by an argument[2] which 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.

Before we can determine what kind of semantics is most appropriate for terminological cycles we should get 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 just be done 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 KL-ONE-based language which allows only concept conjunction and value-restrictions. For this language, the effect of the three above mentioned types of semantics can be completely described with the help of finite automata. These descriptions provide a rather intuitive understanding of terminologies with cyclic definitions and give insight into the essential features of the respective semantics. In addition, subsumption determination for each type

[1]The language allows concept and role conjunction, value-restrictions, number-restrictions and negation of primitive concepts.

[2]Roughly speaking, the argument says that it is sufficient to consider only finite interpretations to determine subsumption relations.

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 subsumption determination.

In the next section we shall recall some definitions and results concerning ordinals, fixed-points and finite automata which 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, the greatest fixed-point semantics should be preferred 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 which are defined by the automaton. Fortunately, these languages have already been investigated in the context of monadic second-order logic (see Büchi (1960)). 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 $\mathcal{FL}^-$ of Levesque-Brachman (1987). The second subsection contains results about hybrid inferences.

2. Formal Preliminaries

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 the 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 $\cup_{n\geq 0} R^n$.

In the following subsections we shall recall some definitions and results concerning ordinals, fixed-points and finite automata.

2.1 Ordinals[3]

A partial ordering $\leq$ on a set D is a *well-ordering* iff it is *linear* (i.e., for all a, b in D we have $a \leq b$ or $b \leq a$) 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 which are smaller than ω. This ordering on ordinals is well-founded and linear. Hence any set of ordinals has a

[3]See Rosenstein (1982) for the order-theoretic approach we use below. A set-theoretic definition of ordinals can be found e.g. in Halmos (1974). Some elementary properties of ordinals are also stated in Lloyd (1987), p.28–29.

least element and a least upper bound.

If α is an ordinal then the *successor* $\alpha+1$ of α is the least ordinal greater than α. An ordinal which 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 $\omega + 1$ of ω is the order typ of $\{ 0, 1, 2, ... \} \cup \{ \infty \}$ 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., $\alpha = \text{lub}(\{ \beta; \beta < \alpha \})$.

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

2.2 Fixed-Points[4]

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 $\text{glb}(C) = \text{lub}(\{ d \in D; d \text{ is a lower bound of } C \})$, and D has a least element bottom = lub($\emptyset$) and a greatest element top = lub(D).

Example 2.1. Consider $D = 2^S$, the set of all subsets of the set S. If the elements of D are ordered by set inclusion, then D is a complete lattice w.r.t. this ordering. Least upper bounds are obtained by set union, and greatest lower bounds by set intersection. The least element of D is $\emptyset$ and the greatest element is S. As a second example of a complete lattice, we may consider the n-fold cartesian product D^n of $D = 2^S$, which is ordered componentwise by inclusion: $(A_1,\ldots,A_n) \subseteq (B_1,\ldots,B_n)$ iff $A_1 \subseteq B_1$, ..., and $A_n \subseteq B_n$. Union and intersection are likewise defined componentwise, top = $(S,\ldots,S)$, and bottom = $(\emptyset,\ldots,\emptyset)$.

Let D be a poset and let $T: D \to D$ be a mapping. Then T is *monotonic* iff for all a, b in D, $a \le b$ implies $T(a) \le T(b)$. A *fixed-point* of T is an element $f \in D$ such that $T(f) = f$ holds. If D is a complete lattice, then any monotonic mapping $T: D \to D$ 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 $T{\uparrow}\alpha := \text{lub}(\{ T{\uparrow}\beta; \beta < \alpha \})$ and $T{\downarrow}\alpha := \text{glb}(\{ T{\downarrow}\beta; \beta < \alpha \})$.

Theorem 2.2. (least and greatest fixed-points)
Let D be a complete lattice, and let $T: D \to D$ be a monotonic mapping. Then, for any ordinal α, $T{\uparrow}\alpha \le \text{lfp}(T)$ and $T{\downarrow}\alpha \ge \text{gfp}(T)$. Furthermore, there exist ordinals β, γ such that $T{\uparrow}\beta = \text{lfp}(T)$ and $T{\downarrow}\gamma = \text{gfp}(T)$. ❑

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 $T: D \to D$ be a mapping. Then T is *upward ω-continuous* (resp. *downward ω-continuous*) iff for any

[4]See Lloyd (1987), Chapter 1, §5 and Schmidt (1986), Chapter 6. An account of the history of related fixed-point theorems can be found in Lassez-Nguyen-Sonenberg (1982).

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 2.3. (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. $gfp(T) = T\downarrow^{\omega} = glb(\{ T^n(top); n \geq 0 \})$).[5] ❑

In Section 5.3 we shall need a slightly generalized version of Theorem 2.3 for downward ω-continuous mappings.

Corollary 2.4. 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 $d \geq T(d)$. Then $d\text{-}gfp(T) := glb(\{ T^n(d); n \geq 0 \})$ is a fixed-point of T. More precisely, d-gfp(T) is the greatest fixed-point of T which is less or equal d.

Proof. Since T is downward ω-continuous and thus monotonic, $d \geq T(d)$ yields $d \geq T(d) \geq T^2(d) \geq T^3(d) \geq \ldots$. Hence $T(glb(\{ T^n(d); n \geq 0 \})) = glb(\{ T^{n+1}(d); n \geq 0 \})) = glb(\{ T^n(d); n \geq 0 \})$.[6] This shows that d-gfp(T) is a fixed-point, and obviously, $d \geq d\text{-}gfp(T)$. If f is a fixed-point with $d \geq f$ then $T(d) \geq T(f) = f$, since T is monotonic, and f is a fixed-point. Iterating this argument we obtain $T^n(d) \geq f$ for all $n \geq 0$, and hence $glb(\{ T^n(d); n \geq 0 \}) \geq f$. ❑

2.3 Automata and Words[7]

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, $W(i) := \sigma_i$ 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)... .

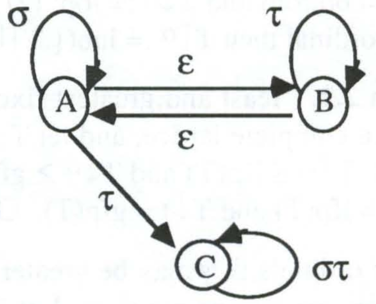
A *generalized finite automaton* $\mathcal{A} = (\Sigma, Q, E)$ consists of a finite alphabet Σ, a finite set of states Q, and a finite set of transitions (or edges) $E \subseteq Q \times \Sigma^* \times Q$. A transition connects two states of Q and is labeled by a finite word over Σ.

Example 2.5. (a generalized automaton)

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

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

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



---

[5] 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.

[6] Since $d = T^0(d) \geq T(d)$ by assumption.

[7] See e.g., Manna (1974), Hopcroft-Ullman (1979), and Eilenberg (1974).

The automaton is called "generalized" because transition labels may be arbitrary words, and not only symbols of the alphabet. However, it is well-known that any generalized finite automaton can be transformed into an equivalent finite automaton[8] (see Manna (1974) or Hopcroft-Ullman (1979)). 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 Hopcroft-Ullman (1979), 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). This transformation can be done in polynomial time.

Let $\mathcal{A}$ be a generalized finite automaton and let p, q be states of $\mathcal{A}$. A *finite path* from p to q in $\mathcal{A}$ is a sequence $p = p_0, U_1, p_1, U_2, p_2, \ldots, U_n, p_n = q$, where 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 $p = p_0, U_1, p_1, U_2, p_2, \ldots$, where 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 $p, W_0, q, W_1, q, W_2, \ldots$ 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 generalized finite automaton $\mathcal{A}$, let $L_{\mathcal{A}}(p,q)$ denote the set of all finite words which are labels of paths fom p to q. If it is clear from the context, we shall omit the index $\mathcal{A}$. In Example 2.5, $L(A,B) = (\sigma\cup\tau)^* = \Sigma^*$ and $L(A,C) = (\sigma\cup\tau)^*\tau(\sigma\tau)^* = \{ W\tau(\sigma\tau)^m;\ W \in \Sigma^*, m \geq 0 \}$. The languages L(p,q) are regular, and on the other hand, any regular language can be obtained in this way. If the regular language $L = L(\mathcal{A})$ is accepted by a set of terminal state Q_{fin}, i.e., $L = \cup_{t\in Q_{fin}} L(p,t)$, we can add a new state q_{fin} to $\mathcal{A}$, and transitions (t,ε,q_{fin}) for all $t \in Q_{fin}$. Then $L = L(p,q_{fin})$.

For a state q of the generalized finite automaton $\mathcal{A}$, let $U_{\mathcal{A}}(p)$ denote the set of all words which are labels of infinite paths starting with p. As for L, we shall often omit the index $\mathcal{A}$. Please note that U(p) may also contain finite words which 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 KL-ONE-based KR-language

In KL-ONE-based knowledge representation languages (KR-languages) we start with atomic concepts and roles and can use the language formalism to define new concepts and roles. Concepts can be considered as unary predicates which are interpreted as sets of individuals whereas roles are binary predicates which are interpreted as binary relations

[8]Accepting the same regular languages.

between individuals. The languages differ in what kind of constructs are allowed for the definition of concepts and roles. The language considered in this paper will be called $\mathcal{FL}_0$. It has only two constructs which can be used to define concepts: concept conjunction and value-restriction.

Definition 3.1. (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 ⊓ D is a concept term. (concept conjunction)

(3) Then ∀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* .[9] The following is an example of a T-box in this formalism: Let Man, Human, Male and Mos (for "*m*an who has *o*nly *s*ons") be concept names and let child be a role name. The T-box consists of the following axioms:

Man = Human ⊓ Male
Mos = Man ⊓ ∀child: Man

That means that a man is human and male. A man who 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. Assume that we want to express a concept "*m*an who has *o*nly *m*ale *o*ff-springs", for short Momo. We can't just introduce a new role name off-spring because there would be no connection between the two primitive roles child and off-spring. But the intended meaning of off-spring is that it is the transitive closure of child. It seems quite natural to use a cyclic definition for Momo: A man who has only male off-springs is himself a man, and all his children are men having only male off-springs, i.e.,

Momo = Man ⊓ ∀child: Momo.

This is a very simple cyclic definition. In general, cycles in terminologies are defined as follows.

Definition 3.2. (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 3.1.

Definition 3.3. (interpretations and models)
An *interpretation* I consists of a set dom(I), the domain of the interpretation, and an

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

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

$$(C \sqcap D)^I := C^I \cap D^I,$$
$$(\forall R{:}C)^I := \{ x \in dom(I);\text{ for all } y \text{ such that } (x,y) \in R^I \text{ we have } y \in C^I \}.$$

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

$A^I = D^I$ for all terminological axioms A = D in T.

The semantics we have just defined[10] is not restricted to non-cyclic terminologies. But for cyclic terminologies this kind of semantics may seem unsatisfactory. One might think that the extension of a defined concept should be completely determined by the extensions of the primitive concepts and roles. This is the case for non-cyclic terminologies.

More precisely, let T be a T-box containing the defined concepts C_1, ..., 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 of T.[11] Of course, we are mostly interested in extensions of J which 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. Nebel (1989a), Section 3.2.4). If T contains cycles, a given primitive interpretation may have different extensions to models of T.

Example 3.4. Let R be a role name and B, P be concept names.[12] The terminology T consists of the single axiom $B = P \sqcap \forall R{:}B$.
We consider the following primitive interpretation: dom(J) := { a, b, c, d } =: P^J, and R^J := { (a,b), (c,d), (d,d) }. It is easy to see that this interpretation has two different extensions to models of T. The defined concept B may be interpreted as { a, b } or as { a, b, c, d }. Note that individuals without R^J-successors are in the extension $(\forall R{:}C)^J$ of a term $\forall R{:}C$, no matter how C may be interpreted.[13]

The example also demonstrates that, with respect to the descriptive semantics defined above, the construction $B = P \sqcap \forall R{:}B$ of the example does not express the

---

[10]This semantics will be called "descriptive semantics" in the following.

[11]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.

[12]We shall no longer use intuitive names for concepts and roles, since I agree with Brachman-Schmolze (1985), p.176, that "suggestive names can do more harm than good in semantic networks and other representation schemes." Suggestive names may seemingly exclude models which are admissible with respect to the formal semantics.

[13]This fact will be very important for the least fixed-point semantics.

value-restriction $B = \forall R^*:P$ for the reflexive-transitive closure R^* of R. This implies that our definition of the concept Momo from above is not correct w.r.t. descriptive semantics.

For these reasons 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 4.1. Let R, S be a role names and A, B, P be concept names, and let T be the terminology $A = Q \sqcap \forall S:B$, $B = P \sqcap \forall R:B$. 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 $A := Q \sqcap \forall S:B$, $B := P \sqcap \forall R:B$ 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 concepts in T.

Definition 4.2. Let T be the terminology which consists of the concept definitions $C_1 = D_1, ..., C_n = D_n$, and let J be a primitive interpretation. The mapping $T_J: (2^{dom(J)})^n \to (2^{dom(J)})^n$ 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 $T_J(Ø,Ø) = (Ø,\{ a_1 \})$ and $T_J(Ø,\{ a_1 \}) = (Ø,\{ a_1, a_2 \})$.

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 2.1) and the mappings T_J are monotonic.[14] Thus the following definition makes sense:

Definition 4.3. (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.

---

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

(2) The *least fixed-point semantics (lfp-semantics)* allows only those models of T which 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 which 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 3.4, the extension of J which interprets B as { a, b } is a lfp-model of T, and the extension which interprets B as { a, b, c, d } is a gfp-model of T. It is easy to see that, for cycle-free terminologies, lfp-, gfp- and descriptive semantics coincide (see Nebel (1989a), p.137,138).

The next question is how lfp-models (gfp-models) can be constructed from a given primitive interpretation. Nebel (1987,1989,1989a) claimed that the mappings T_J are even upward continuous, and that thus $lfp(T_J) = \cup_{i\geq 0}T_J{}^i(bottom)$, where bottom denotes the least element of $(2^{dom(J)})^n$, namely the n-tuple $(\emptyset,...,\emptyset)$. Unfortunately, this is not true.

Proposition 4.4. In general, we may have $lfp(T_J) \neq \cup_{i\geq 0}T_J{}^i(bottom)$.
Proof. We consider Example 4.1. 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 4.5. The mappings T_J are always downward ω-continuous. Consequently, the greatest fixed-point may be obtain as $gfp(T_J) = \cap_{i\geq 0}T_J{}^i(top)$, 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 $\underline{A} := \cap_{k\geq 0}\underline{A}^{(k)}$. By Definition 4.2, it is sufficient to demonstrate that, for any concept term D, we have

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

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 ≥ 0 and hence $\cap_{k\geq 0} D^{I_k} = P^J = D^I$.
(2) *D = C_i for a defined concept C_i*. Then $D^I = A_i$, and for all k ≥ 0, $D^{I_k} = A_i{}^{(k)}$.[15] But $A_i = \cap_{k\geq 0}A_i{}^{(k)}$ 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 $E^I = \cap_{k\geq 0} E^{I_k}$ and $F^I = \cap_{k\geq 0} F^{I_k}$. Hence $D^I = (\cap_{k\geq 0} E^{I_k}) \cap (\cap_{k\geq 0} F^{I_k}) = \cap_{k\geq 0} (E^{I_k} \cap F^{I_k}) = \cap_{k\geq 0} D^{I_k}$.
(4) *D = ∀R:C for a role name R and a concept term C*. By Definition 3.3, $D^I = \{ x \in dom(I);\ \forall y: ((x,y) \in R^I \rightarrow y \in C^I) \}$, and hence, by induction and the definition of I, $D^I = \{ x \in dom(J);\ \forall y: ((x,y) \in R^J \rightarrow y \in \cap_{k\geq 0} C^{I_k}) \}$. That means that we have

[15] Here A_i is the i-th component of the tuple $\underline{A}$ and $A_i{}^{(k)}$ is i-th component of the tuple $\underline{A}^{(k)}$.

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

It is well-known (see e.g., Gallier (1986), 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[16] we get $\forall k:\forall y$: (A $\rightarrow$ B). This shows that

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

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

The two propositions show that, from a constructive point of view, the gfp-semantics should be preferred. 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.

An important service terminological representation systems provide is computing the subsumption hierarchy.

Definition 4.6. (subsumption of concepts)
Let T be a terminology and let A, B be concept names.

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

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

5. Characterization of the Semantics using Finite Automata

Before we can associate a finite automaton $\mathcal{A}_T$ to a terminology T we must 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.[17] 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., ε^I = { (d,d); d $\in$ dom(I) }.

Definition 5.1. Let T be a terminology where all terms are normalized as described above. The generalized (nondeterministic) automaton $\mathcal{A}_T$ is defined as follows: The alphabet of $\mathcal{A}_T$ is the set $\mathbf{R_T}$ 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 5.1.

---

[16]This is the point where the proof for the least fixed-point goes wrong. In this case we would have the quantifiers "$\forall y:\exists k$:" which cannot be permuted.

[17]i.e., they have the same extension in any interpretation.

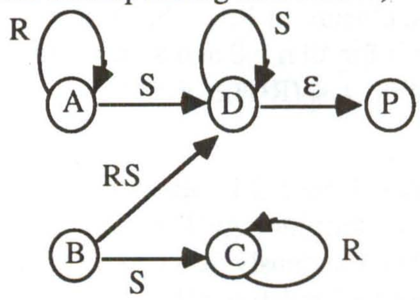
Example 5.2. (A normalized terminology and the corresponding automaton)

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

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

$C = \forall R{:}\ C$

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



The primitive concepts are exactly those states in $\mathcal{A}_T$ which don't have successor states. The 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 5.3. Let T be a terminology and let $\mathcal{A}_T$ be the corresponding 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 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 $e \in P^I$.
Proof. If A is a primitive concept, then $L(A,A) = \{ \varepsilon \}$ and $L(A,P) = \emptyset$ for $A \neq P$. Since $\varepsilon^I = \{ (d,d);\ d \in \text{dom}(I) \}$, 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 $\text{gfp}(T_J) = \bigcap_{k\geq 0} T_J{}^k(\text{top})$. The defined concept A corresponds to a component of this tuple, i.e., $A^I = (\text{gfp}(T_J))_i$ for $i = \text{index}(A)$.
(1) Assume that $d \notin A^I$. Then there exists $k \geq 0$ such that $d \notin (T_J{}^k(\text{top}))_i$. We proceed by **induction on k**.
For **k = 0,** we have $d \notin (\text{top})_i = \text{dom}(I)$, which is a contradiction.
For **k > 0** we have $d \notin (T_J(T_J{}^{k-1}(\text{top})))_i$. Let the defining axiom for A be of the form $A = \ldots \sqcap \forall W{:}\ B \sqcap \ldots$ and assume that $\forall W{:}\ B$ is responsible for $d \notin (T_J(T_J{}^{k-1}(\text{top})))_i$. That means that there exists $e \in$ dom(I) such that dW^Ie and $e \notin B^J = B^I$ (if B is a primitive concept) or $e \notin (T_J{}^{k-1}(\text{top}))_j$ (if B is a defined concept and index(B) = j). In the first case, B is a primitive concept and obviously, $W \in L(A,B)$. In the second case, we can apply the induction hypothesis to $e \notin (T_J{}^{k-1}(\text{top}))_j$. Thus there exist a primitive concept P, a word $V \in L(B,P)$ and an individual $f \in$ dom(I) such that eV^If and $f \notin P^I$. But then $WV \in L(A,P)$ and $d(WV)^If$. This completes the proof of the "if" direction.
(2) Assume that there exist a primitive concept P, a word $W \in L(A,P)$ and an individual $e \in$ dom(I) such that dW^Ie and $e \notin P^I$. 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 $A = \ldots \sqcap \forall W{:}\ P \sqcap \ldots$. Thus $d \notin (T_J(\text{top}))_i$.
For **n > 0**, we know by induction that $d_1 \notin (T^h(\text{top}))_j$ for some $h > 0$ (where index(C_1) = j). But then $d \notin (T^{h+1}(\text{top}))_i$. This completes the proof of the proposition since $A^I = (\text{gfp}(T_J))_i = \bigcap_{k\geq 0}(T_J{}^k(\text{top}))_i$. ❑

For the terminology $B = P \sqcap \forall R{:}B$ of Example 3.4, $L(B,P) = R^* = \{ R^n;\ n \geq 0 \}$. 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 R. In this case, the condition of the proposition says that $d \in B^I$ if and only if for all $n \geq 0$ and all e such that $d(R^I)^n e$, $e \in P^I$ holds. That means that for all e such that $d(\cup_{n\geq 0}(R^I)^n)e$, $e \in P^I$ holds. But $\cup_{n\geq 0}(R^I)^n$ is the reflexive-transitive closure of R^I.

Proposition 5.3 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 which 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 $\forall W$: P which 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 5.4. Let T be a terminology and let $\mathcal{A}_T$ be the corresponding 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,

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

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: $dom(J) := \{ d_0, ..., d_n \}$; $Q^J := dom(J)$ 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 5.3 that $d_0 \notin B^I$. On the other hand, assume that $d_0 \notin A^I$. By Proposition 5.3, 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 5.3 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. ❑

In Example 5.2, 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.[18] 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 non-

[18] 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 which are defined by a non-deterministic automaton having the same size as the terminology.

deterministic 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 Eilenberg (1974), p. 23). We consider the 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 which has the states in $Q_1 \cup Q_2$ as its defined concepts and the state t as the only[19] primitive concept. But then $L_1 \subseteq L_2$ if and only if $p_2 \sqsubseteq_{gfp,T} p_1$.

Corollary 5.5. 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 a nondeterministic automaton) in polynomial time and vice versa. It is well-known that the inclusion problem for regular languages defined by a nondeterministic automaton is PSPACE-complete (see Garey-Johnson (1979)). ❑

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, Nebel (1989b) has shown that, even without cycles, this languages has a co-NP-complete subsumption problem.

5.2 Characterization of the lfp-Semantics

In order to get a characterization of lfp-semantics which is similar to the characterization of gfp-semantics in Proposition 5.3, 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 generalized 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. That 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 5.6. 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 $A = \ldots \sqcap \forall W{:}\ B \sqcap \ldots$ 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 5.7. Assume that $d \in (T_J{\uparrow}\lambda)_i$, 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.

[19]In order to have this property the automata had to be trim.

(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 $dW^I e$, there are individuals d_1, ..., d_{n-1} such that $dU_0{}^I d_1 \ldots d_{n-1} U_n{}^I e$.
For $n = 0$, $W = U_0$ and the defining axiom for A is of the form $A = \ldots \sqcap \forall W: P \sqcap \ldots$. Thus $d \in (T_J(T_J\uparrow^\delta))_i$ and $dW^I e$ imply $e \in P^I$.
For $n > 0$, the defining axiom for A is of the form $A = \ldots \sqcap \forall U_0: C_1 \sqcap \ldots$, and thus $d \in (T_J(T_J\uparrow^\delta))_i$ and $dU_0{}^I d_1$ 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 5.8. Let T be a terminology and let $\mathcal{A}_T$ be the corresponding 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 $d_0 \in A^I = (\text{lfp}(T_J))_i$. 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 5.7. 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 5.6, 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,[20] and d is an individual with $d_0 W^I d$. 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 $dU^I e$. 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[21] 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).[22]
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 $B = \ldots \sqcap \forall U: C \sqcap \ldots \sqcap \forall V: P \ldots$, where P is primitive and C defined. The minimality of (W,d,B) implies that there does not exist an individual e with $dU^I e$. Assume that $dV^I e$. Since $WV \in L(A,P)$ and $d_0(WV)^I e$, property (P1) implies $e \in P^I$. This shows that $d \in (T_J(\text{bottom}))_j$. Hence we can take $\lambda = 1$.

---

[20]For A = B this may also be the empty path.
[21]See e.g., Gallier (1986), p. 9, 10, for the definition and justification of noetherian induction.
[22]Recall that α was a limit ordinal such that $\text{lfp}(T_J) = T_J\uparrow^\alpha$.

(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 $B = \forall U_1: C_1 \sqcap \ldots \sqcap \forall U_n: C_n \sqcap \ldots \sqcap \forall V: P \ldots$, where P is primitive and the C_i are all the defined concepts in the definition of B. As in (2.1) we can show for all individuals e that dV^Ie implies $e \in P^I$. Assume that $dU_i{}^Ie$ 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 $1 \leq i \leq n$ and $dU_i{}^Ie\ \}$. Then we have $\gamma \leq \alpha$ 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. ❑

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 $d_0 \in A^I$. ❑

As a consequence of 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 5.9. 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 which 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 $d\varepsilon^Id$ 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 5.8 holds for A, d_0. Hence property (P2) cannot hold. That 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 which 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 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 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 5.9, there exists a path with label ε from A to a state B in $\mathcal{A}_T$ which 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 automaton $\mathcal{B}_T$. That 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 $V \in \Sigma^* \cup \Sigma^\omega$ $\}$.[23]

Theorem 5.10. 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. **(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 $Q \neq P$; $P^J := dom(J) \setminus \{ d_n \}$; $R^J := \{ (d_i,d_{i+1}); 0 \le i \le n-1 \text{ and } R = R_{i+1} \}$ 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 5.8 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 which lies on an ε-cycle in $\mathcal{A}_T$. Since $d_0U^Id_k$ for some $k \le n$ and $d_k\varepsilon^Id_k\varepsilon^Id_k\ldots$, property (P2) of Proposition 5.8 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 5.8, (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 $k \ge 0$ such that $W_1\ldots W_k$ is a prefix of W and $W_{k+1} = W_{k+2} = \ldots = \varepsilon$. That 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, $W \in L_{\mathcal{B}_T}(A,P)$ which is a contradiction.

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

(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; $R^J := \{ (d_i,d_{i+1}); i \ge 0 \text{ and } R = R_{i+1} \}$ 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 5.8 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 which is the label of a finite path in $\mathcal{A}_T$ from B to a concept C which lies on an ε-cycle in $\mathcal{A}_T$. 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 5.8, (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 $W_1W_2W_3$... 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 $W_1W_2W_3$... of the infinite path is a finite word U, the

[23]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 $\mathcal{A}_T$.

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 $A \not\sqsubseteq_{gfp,T} B$.

(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 which 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}); $0 \le i \le n-1$ 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 5.8, 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., there exist an infinite path A, W_1, C_1, W_2, C_2, W_3, C_3, ... in $\mathcal{A}_T$ and individuals $e_0 = d_0, e_1, e_2, e_3, \ldots$ such that $(e_{m-1}, e_m) \in W_m{}^I$ for all m > 0. The definition of J implies that there exists $k \ge 0$ such that $W_1 \ldots W_k$ is a prefix of U and $W_{k+1} = W_{k+2} = \ldots = \varepsilon$. That 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 $A \not\sqsubseteq_{gfp,T} B$.

(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 5.8 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 5.8 yields $d_0 \notin A^I$, which is a contradiction. Assume that $W \in L_{\mathcal{B}_T}(A,P) \setminus L_{\mathcal{A}_T}(A,P)$. That means that W = UV, and there is a path with label U in $\mathcal{A}_T$ from A to a concept C which 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 5.8 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 $d_1, d_2, d_3, \ldots$ such that $(d_{n-1}, d_n) \in W_n{}^I$ for all n > 0.

(4.2.1) First, we assume that the label $W_1 W_2 W_3 \ldots$ 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 which 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 $W_1 W_2 W_3 \ldots$ 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 which 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. ❑

In Example 5.2, 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. This problem can be split 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 5.11. Let $\mathcal{B}$ be an arbitrary generalized automaton. Then $F_{\mathcal{B}}(B) \subseteq F_{\mathcal{B}}(A)$ can be decided by a PSPACE-algorithm.

Proof. The generalized automaton $\mathcal{B} = (\Sigma,Q,E)$ is modified to a generalized 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 $\mathcal{B}$ which 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 $C_i = C_j$. That 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 $W \in L_{\mathcal{C}}(A,Fin)$. That means that there exists a path in $\mathcal{B}$ with label W from A to a state C which 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. ❑

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 5.12. Let $\mathcal{B}$ be an arbitrary generalized automaton. 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 generalized automaton $\mathcal{B} = (\Sigma,Q,E)$ can be modified in polynomial time to an ordinary finite automaton[24] $\mathcal{A} = (\Sigma,Q_1,E_1)$ 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 $W \in L_{\mathcal{A}}(A,B)$.[25]

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 $\mathcal{B}$ which has W as label. Since W is an infinite word, there exist infinitely many indices $0 < i_1 < i_2 < \ldots$ such that the words $W_0 \ldots W_{i_1-1}$, $W_{i_1} \ldots W_{i_2-1}$, ... are not empty. By property (1.3), $W_0 \ldots W_{i_1-1} \in L_{\mathcal{A}}(A,C_{i_1})$, $W_{i_1} \ldots W_{i_2-1} \in L_{\mathcal{A}}(C_{i_1},C_{i_2})$, ... 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 $\mathcal{A}$ which has W as label. By property (1.2), there exist infinitely many indices $0 < i_1 < i_2 < \ldots$ such that C_{i_1}, C_{i_2}, ... are in Q. By property (1.3), $W_0 \ldots W_{i_1-1} \in L_{\mathcal{B}}(A,C_{i_1})$, $W_{i_1} \ldots W_{i_2-1} \in L_{\mathcal{B}}(C_{i_1},C_{i_2})$, ... This shows that there exists an infinite path from A with label W in $\mathcal{B}$, i.e., $W \in I_{\mathcal{B}}(A)$. ❑

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

[24]Where transitions are only labeled by symbols of the alphabet.

[25]The additional states in Q_1 are intermediate states which are needed for the elimination of transitions which 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 Hopcroft-Ullman (1979), p. 26, Theorem 2.2.

$\mathcal{A}$ 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 $U \in E_{\mathcal{A}}(A)$. ❑

(3) Obviously, the languages $E_{\mathcal{A}}(A)$ are regular languages defined by $\mathcal{A}$. Hence there is a PSPACE-algorithm which 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 5.13. 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 nondeterministic automata[26] 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 5.5). We consider the 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 which 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 shall prove a proposition for $\underline{A}$-gfp-models (see Corollary 2.4) which is similar to Proposition 5.3 for gfp-models.

Proposition 5.14. Let T be a terminology and let $\mathcal{A}_T$ be the corresponding 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 2.4).

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 5.3). 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

[26]Without loss of generality the transitions are only labeled by symbols of the alphabet.

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 $d \notin (T_J(T_J^{k-1}(\underline{A})))_i$. That means that there exists $e \in dom(I)$ such that $dW^I e$ 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 $eV^I f$ 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 $d(WV)^I f$. 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 5.3. ❑

We can now characterize subsumption w.r.t. descriptive semantics. Infinite paths are still important but it is not enough to consider just their labels. The states which are reached infinitely often by this path are also significant. An infinite path which 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 5.15. Let T be a terminology and let $\mathcal{A}_T$ be the corresponding 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 5.14, $d \notin B^I$ 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 5.14 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, ..., 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 5.14, $e_k \notin C_k^I$ 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, ... with the above described properties. But then there is a concept C such that $C = C_i$ for infinitely many indices i. That 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 5.14 yields $d \notin A^I$.

(2) Assume that $A \sqsubseteq_T B$. This implies $A \sqsubseteq_{gfp,T} B$ and thus, by Theorem 5.4, 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.[27] 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 $D = \ldots \sqcap \forall V_1\colon D' \sqcap \ldots$. Let m' be the index of D'. The definition of $\underline{A}$ yields $e' \notin (\underline{A})_{m'}$ and thus $e \notin (T_J(\underline{A}))_m$. ❑

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 $d_0\varepsilon^Jd_0$, $d_0U_1{}^Jd_{j_1}$ and $\varepsilon \in L(B,B)$, $U_1 \in L(B,C)$. 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 5.14 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. ❑

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 generalized automaton.[28] For this problem we can't get an ad hoc reduction to an inclusion problem for regular languages. But 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 Büchi (1960) and Eilenberg (1974), Chapter XIV).

---

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

[28] However, it may not be the best way 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 $\mathcal{A} = (\Sigma,Q,E)$ be a (nondeterministic) finite automaton[29] and let I, T be subsets of Q. 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 which can be obtained as $W = W_1W_2W_3\ldots$ where $W_1, W_2, W_3, \ldots$ 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 6.1. (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 $H(K^\omega)$ where H and K are regular languages in Σ^*.[30]

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

Proof. See Eilenberg (1974), 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 get

Corollary 6.2. The inclusion problem is decidable for the class of all languages accepted by Büchi automata.

Proof. Obviously, $L_1 \subseteq L_2$ iff $L_1 \cap (\Sigma^\omega \setminus L_2) \neq \emptyset$. Thus the inclusion problem can be reduced to the emptiness problem since the proof of Theorem 1.4 in Eilenberg (1974) is effective, i.e., from given Büchi automata for L_1 and L_2 one can effectively construct a Büchi automaton for $L_1 \cap (\Sigma^\omega \setminus L_2)$.[31]

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 6.2 does not yield the complexity of the inclusion problem. However, Sistla-Vardi-Wolper (1987) have shown 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 Thomas (1989), proof of Lemma 1.2), we obtain a PSACE-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 $L_1(\{\#\}^\omega) \subseteq L_2(\{\#\}^\omega)$. By Theorem 6.1, $L_1(\{\#\}^\omega)$ and $L_2(\{\#\}^\omega)$ are languages accepted by Büchi automata. Thus we have

---

[29] $\mathcal{A}$ is not generalized, i.e. $E \subseteq Q \times \Sigma \times Q$.

[30] The language $H(K^\omega)$ 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.

[31] However, this automaton may have a size which is exponential in the size of the initial automata (see Pécuchet (1986) and Sistla-Vardi-Wolper (1987) for size bounds for the complememt automaton).

Proposition 6.3. 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 generalized automaton 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

$L_{p,q} := \{ W; W \in \Sigma^*$ 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 Hopcroft-Ullman (1979), Theorem 3.5).

Lemma 6.4. (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 $U_0\#U_1\#U_2\#\ldots$ 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 6.1 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 6.3, 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 6.5. Subsumption w.r.t. descriptive semantics can be decided with polynomial space using Büchi automata.[32] ❑

7. Extensions of the Results for the gfp-Semantics

We shall now consider two extensions of the results for the gfp-semantics. In the first subsection, we shall allow an additional concept forming construct, namely exists-

[32]Using Theorem 5.15 above, Nebel recently was able to characterize equivalence of concepts w.r.t. descriptive semantics with the help of deterministic automata. This characterization also yields PSPACE-algorithms for equivalence and for subsumption w.r.t. descriptive semantics (see Nebel (1990)). However, it is still open whether these problems are PSPACE-hard.

restriction. In the second subsection, we shall introduce an assertional component into our KR-system, and consider hybrid inferences.

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

In order to extend our language $\mathcal{FL}_0$ to the language $\mathcal{FL}^-$ of Levesque-Brachman (1987), we have to add a fourth rule to the definition of concept terms (Definition 3.1):
Let R be a role name.

(4) Then $\exists R$ is a concept term. (exists-restriction)

For example, the concept Father can be defined as

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

That means that a father is a man who 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 4.2. It is easy to see that these mappings are 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 $\exists R$. As in Section 5, the prefix "$\forall R_1{:}\forall R_2{:}...\forall R_n$" will be abbreviated by "$\forall W$" where $W = R_1R_2 ...R_n$. Let T be a terminology of $\mathcal{FL}^-$. The corresponding generalized (nondeterministic) automaton $\mathcal{A}_T$ is defined as in Definition 5.1. The only difference is that we also have the terms $\exists R$ occurring in T as states of $\mathcal{A}_T$. These states are similar to the states P for primitive P in that they don't 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 7.1. Let T be a terminology of $\mathcal{FL}^-$, and let $\mathcal{A}_T$ be the corresponding automaton. Let I be a gfp-model of T, and let A be a concept name occurring in T. For any $d \in \text{dom}(I)$ 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 $e \in \text{dom}(I)$, $(d,e) \in W^I$ implies $e \in P^I$.
(2) For all terms $\exists R$ in T, all words $W \in L(A,\exists R)$, and all individuals $e \in \text{dom}(I)$, $(d,e) \in W^I$ implies $e \in (\exists R)^I$, i.e., there is $f \in \text{dom}(I)$ such that $(e,f) \in R^I$.
Proof. The proof is very similar to the proof of Proposition 5.3. ❑

Theorem 7.2. Let T be a terminology of $\mathcal{FL}^-$, and let $\mathcal{A}_T$ be the corresponding 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: $\text{dom}(J) := \{ d_0, \ldots, d_n, e \}$; $Q^J := \text{dom}(J)$ for all primitive concepts $Q \neq P$; $P^J := \text{dom}(J) \setminus \{ d_n \}$; $R^J := \{ (d_i,d_{i+1});\ 0 \leq i \leq n-1 \text{ and } R = R_{i+1} \} \cup \{ (d_i,e);\ 0 \leq i \leq n \} \cup \{ (e,e) \}$ for all roles R. The definition of the role extensions implies that $d_0V^Jd_n$ iff $V = W$, and that $(\exists R)^J = \text{dom}(J)$ for all roles R.

Let I be the gfp-model defined by J. As in part (1) of the proof of Theorem 5.4 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; $S^J := \{ (d_i,d_{i+1}); 0 \leq i \leq n-1$ and $S = R_{i+1} \} \cup \{ (d_i,e); 0 \leq i \leq n \} \cup \{ (e,e) \}$ for all roles $S \neq R$; $R^J := \{ (d_i,d_{i+1}); 0 \leq i \leq n-1$ and $S = R_{i+1} \} \cup \{ (d_i,e); 0 \leq i \leq n-1 \} \cup \{ (e,e) \}$. 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 7.1 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 7.1 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 5.4. ❑

The theorem shows that, with respect to subsumption, terms of the form $\exists R$ behave just like primitive concepts. As a consequence, we obtain

Corollary 7.3. Subsumption determination in $\mathcal{FL}^-$ can be reduced in linear time to subsumption determination in $\mathcal{FL}_0$.
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 $\exists R$ 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, Theorem 7.2 implies that $A \sqsubseteq_T B$ iff $A \sqsubseteq_{T_0} B$. ❑

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

Corollary 7.4. 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 5.14 and Theorem 5.15) can be generalized to $\mathcal{FL}^-$ in an analogous way.[33] For the lfp-semantics, one can also prove an analogous generalization of Proposition 5.8. But for subsumption one runs into new problems. The reason is that there is an additional source of inconsistency.

Example 7.5. 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 get 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 $U_{\mathcal{B}_T}(A) = \{ SSS\ldots \} \not\subseteq U_{\mathcal{B}_T}(B) = \{ RRR\ldots \}$.

---

[33] i.e., the terms $\exists R$ are treated like primitive concepts as in condition (2) of Proposition 7.1.

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 describes a part of some world. KR-systems which allow both T-boxes and A-boxes are sometimes called hybrid systems.

Definition 7.6. (world descriptions)
Let **C** be a set of concept names, **R** be a set of role names, and **I** be a set of 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. That means that Willy is a man who has the child Brian.

Definition 7.7. (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 3.3) can be extended to an interpretation of T $\cup$ 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 $\cup$ A is a model of T $\cup$ 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).[34]

A model I of T $\cup$ A is a gfp-model (lfp-model) of T $\cup$ 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 $R^J = dom(J) \times dom(J)$, then $gfp(T_J) = top$ by Proposition 5.3. This shows that the gfp-model of T defined by J is a model of T $\cup$ A for any A-box A. Thus any combination T $\cup$ 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 which is inconsistent w.r.t lfp-semantics (see Corollary 5.9), and A contains an axiom C(a), then T $\cup$ A does not have an lfp-model.

An important service hybrid representation systems provide is computing instance relationships.

Definition 7.8. (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.

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

[34] 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 $\cup$ A.

In this case we say that a is an *instance* of C in $T \cup A$ 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 generalized automaton $\mathcal{A}_T$ which 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 the transitions of $\mathcal{A}_T$ are labeled by symbols of the alphabet.[35] An A-box A defines an automaton $\mathcal{A}_A$ 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 automaton* $\mathcal{B}_{T\cup A} = \mathcal{A}_T \times \mathcal{A}_A$ of $\mathcal{A}_T$ and $\mathcal{A}_A$ (see e.g., Eilenberg (1974), 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 7.9. Let T be a T-box of $\mathcal{FL}_0$ and A be an A-box defined over the same sets of concept an role names. Let b be an individual name in A and C 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 5.3, 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 $U = U_1U_2$, $f^I U_1{}^I e_0 U_2{}^I e$ 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 =

---

[35]Proposition 5.3 and Theorem 5.4 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 generalized automaton into an ordinary automaton.

$U_1W \in L_{\mathcal{A}_T}(F,P)$, 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 5.3, 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 $f^I U^I b^I$.[36] 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. ❑

We shall now show how the property stated on the right hand side of the theorem can be decided for given b, B.

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 7.10. 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 7.11. 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. That 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 subsumption determination w.r.t gfp-semantics in $\mathcal{FL}_0$ is PSPACE-complete we have thus proved

Corollary 7.12. Instance testing w.r.t. gfp-semantics is PSPACE-complete. ❑

---

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

8. Conclusion

We have considered a small terminological language because for this language the meaning of terminological cycles with respect to different kinds of semantics, and in particular, the important subsumption relation could be characterized with the help of finite automata. These results may help to decide what kind of semantics is most appropriate for cyclic definitions, not only for this small language, but also for suitably extended languages.

As it stands, the gfp-semantics comes off best. The characterizations given in Proposition 5.3 and Theorem 5.4 are easy, and have an obvious intuitive interpretation. Furthermore, important constructs – such as value-restriction with respect to the reflexive-transitive closure of a role – can easily be expressed. The lfp-semantics is less constructive, and the modifications of the automaton which are necessary to characterize subsumption are not obvious. For the descriptive semantics one has to consider certain languages of infinite words which are more difficult and less intuitive than the regular languages which occur in the context of gfp-semantics.

This research can be continued in two directions. First, one may try to extend the results to cyclic definitions in larger languages. As a first step in this direction, the results for gfp-semantics were extended in Section 7.1 to cycles in the language $\mathcal{FL}^-$ of Levesque-Brachman (1987). Hybrid inferences such as "instance testing" can also be handled for gfp-semantics, as shown in Section 7.2.

Secondly, one can use a larger language, but restrict cycles to the small language. One idea in this direction is to extend a given language by value-restrictions of the form $\forall L{:}P$ where L is a regular language over the alphabet of role names. In accordance with part (1) of Proposition 5.3, the semantics of this construct should be defined as $(\forall L{:}P)^I := \{ d \in dom(I)$; for all words $W \in L$ and all individuals $e \in dom(I)$, $(d,e) \in W^I$ implies $e \in P^I \}$. For example, $\forall RR^*{:}P$ would express value-restriction with respect to the transitive closure of the role R (RR^* is the regular language $\{ R^n; n \geq 1 \}$).

9. References

Aho, A.V., Ullman, J.D. (1979). Universality of Data Retrieval Languages. Proceedings of the *6th ACM Symposium on Principles of Programming Languages.*

Brachman, R.J., Schmolze, J.G. (1985). An Overview of the KL-ONE Knowledge Representation System. *Cognitive Science* **16.**

Brachman, R.J., Pigman-Gilbert, V., Levesque, H.J. (1985). An Essential Hybrid Reasoning System: Knowledge and Symbol Level Accounts in KRYPTON. Proceedings of the *9th International Joint Conference on Artificial Intelligence,* Los Angeles.

Büchi, J.R. (1960). On a Decision Method in Restricted Second Order Arithmetic. Proceedings of the *1960 Congress on Logic, Methodology and Philosophy of Science,* Stanford.

Eilenberg, S. (1974). *Automata, Languages and Machines,* Vol. A. Academic Press, New York/London.

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

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

Gurevich, Y., Shelah, S. (1985). Fixed-point Extensions of First-Order Logic. Proceedings of the *26th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society Press.

Gurevich, Y., Shelah, S. (1986). Fixed-point Extensions of First-Order Logic. *Annals of Pure and Applied Logic* **32**.

Halmos, P.R. (1974). *Naive Set Theory*. Springer Verlag, New York.

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

Immerman, N. (1982). Relational Queries Computable in Polynomial Time. Proceedings of the *4th ACM Symposium on the Theory of Computing*.

Kaczmarek, T.S., Mark, W., Sondheimer, K. (1986). Recent Developments in NIKL. Proceedings of the *5th National Conference of the American Association for Artificial Intelligence*, Philadelphia.

Lassez, J.-L., Nguyen, V.L., Sonenberg, E.A. (1982). Fixed Point Theorems and Semantics: A Folk Tale. *Information Processing Letters* **14**.

Levesque, H.J., Brachman, R.J. (1987). Expressiveness and Tractability in Knowledge Representation and Reasoning. *Computational Intelligence* **3**.

Lloyd, J.W. (1987). *Logic Programming*, Second, Extended Edition. Springer Verlag, Berlin.

McNaughton, R. (1966). Testing and Generating Infinite Sequences by a Finite Automaton. *Information and Control* **9**.

MacGregor, R., Bates, R. (1987). The Loom Knowledge Representation Language. Technical Report ISI/RS-87-188, University of Southern California, Information Science Institute.

Manna, Z. (1974). *Mathematical Theory of Computation*. McGraw-Hill, New York.

Minker, J., Nicolas, J.M. (1983). On Recursive Axioms in Deductive Databases. *Information Systems* **8**.

Nebel, B. (1987). On Terminological Cycles. *KIT Report* **58**, Technische Universität Berlin.

Nebel, B. (1989). On Terminological Cycles. Preprints of the *Workshop on Formal Aspects of Semantic Networks*, Two Harbors.

Nebel, B. (1989a). *Reasoning and Revision in Hybrid Representation Systems*. PhD Thesis, Universität des Saarlandes, Saarbrücken, West Germany.

Nebel, B. (1989b). Terminological Reasoning is Inherently Intractable. IWBS Report 82, IWBS, IBM Stuttgart.

Nebel, B. (1990). Terminological Cycles: Semantics and Computational Properties. To appear in J. Sowa (ed.). *Formal Aspects of Semantic Networks*.

Pecuchet, J.P. (1986). On the Complementation of Büchi Automata. *Theoretical Computer Science* **47**.

Rosenstein, J.G. (1982). *Linear Ordering*. Academic Press, New York.

Schmidt, D.A. (1986). *Denotational Semantics: A Methodology for Language Development*. Allyn and Bacon, Boston.

Sistla, A.P., Vardi, M.Y., Wolper, P. (1987). The Complementation Problem for Büchi Automata with Applications to Temporal Logic. *Theoretical Computer Science* **49**.

Thomas, W. (1989). Automata on Infinite Objects. Proceedings of the *École de Printemps 1989*, Ramatuelle.

Vardi, M. (1982). Complexity of Relational Query Languages. Proceedings of the *4th ACM Symposium on the Theory of Computing*.

Vielle, L. (1989). Recursive Query Processing: The Power of Logic. *Theoretical Computer Science* **69**.

Wu, C.S., Henschen, L.J. (1988). Answering Linear Recursive Queries in Cyclic Databases. Proceedings of the *International Conference on Fifth Generation Computer Systems*, Tokyo.

Deutsches
Forschungszentrum
für Künstliche
Intelligenz GmbH

DFKI Publikationen

Die folgenden DFKI Veröffentlichungen können bezogen werden von

DFKI
-Bibliothek-
Postfach 2080
6750 Kaiserslautern

DFKI Publications

The following DFKI publications can be ordered from

DFKI -Library-
P.O. Box 2080
6750 Kaiserslautern
Federal Republic of Germany

DFKI Research Reports

RR-90-01
Franz Baader
Terminological Cycles in KL-ONE-based Knowledge Representation Languages
33 pp.

Abstract: Cyclic definitions are often prohibited in terminological knowledge representation languages, because, from a theoretical point of view, their semantics is not clear and, from a practical point of view, existing inference algorithms may go astray in the presence of cycles. In this paper we consider terminological cycles in a very small KL-ONE-based language. For this language, the effect of the three types of semantics introduced by Nebel (1987, 1989, 1989a) can be completely described with the help of finite automata. These descriptions provide a rather intuitive understanding of terminologies with cyclic definitions and give insight into the essential features of the respective semantics. In addition, one obtains algorithms and complexity results for subsumption determination. The results of this paper may help to decide what kind of semantics is most appropriate for cyclic definitions, not only for this small language, but also for extended languages. As it stands, the greatest fixed-point semantics comes off best. The characterization of this semantics is easy and has an obvious intuitive interpretation. Furthermore, important constructs – such as value-restriction with respect to the transitive or reflexive-transitive closure of a role – can easily be expressed.

RR-90-02
Som Bandyopadhyay
Towards an Understanding of Coherence in Multimodal Discourse
18 pp.

Abstract: An understanding of coherence is attempted in a multimodal framework where the presentation of information is composed of both text and picture segments (or, audio-visuals in general). Coherence is characterised at three levels: coherence at the syntactic level which concerns the linking mechanism of the adjacent discourse segments at the surface level in order to make the presentation valid; coherence at the semantic level which concerns the linking of discourse segments through some semantic ties in order to generate a wellformed thematic organisation; and, coherence at the pragmatic level which concerns effective presentation through the linking of the discourse with the addressees´ preexisting conceptual framework by making it compatible with the addressees´ interpretive ability, and linking the discourse with the purpose and situation by selecting a proper discourse typology. A set of generalised coherence relations are defined and explained in the context of picture-sequence and multimodal presentation of information.

RR-90-03
Hans-Jürgen Bürckert
A Resolution Principle for Clauses with Constraints
25 pages

Abstract: We introduce a general scheme for handling clauses whose variables are constrained by an underlying constraint theory. In general, constraints can be seen as quantifier restrictions as they filter out the values that can be assigned to the variables of a clause (or an arbitrary formulae with restricted universal or existential quantifier) in any of the models of the constraint theory. We present a resolution principle for clauses with constraints, where unification is replaced by testing constraints for satisfiability over the constraint theory. We show that this constrained resolution is sound and complete in that a set of clauses with constraints is unsatisfiable over the constraint theory iff we can deduce a constrained empty clause for each model of the constraint theory, such that the empty clauses constraint is satisfiable in that model. We show also that we cannot require a better result in general, but we discuss certain tractable cases, where we need at most finitely many such empty clauses or even better only one of them as it is known in classical resolution, sorted resolution or resolution with theory unification.

RR-90-04
Andreas Dengel & Nelson M. Mattos
Integration of Document Representation, Processing and Management
18 pages

Abstract: We introduce a general scheme for handling clauses whose variables are constrained by an underlying constraint theory. In general, constraints can be seen as quantifier restrictions as they filter out the values that can be assigned to the variables of a clause (or an arbitrary formulae with restricted universal or existential quantifier) in any of the models of the constraint theory. We present a resolution principle for clauses with constraints, where unification is replaced by testing constraints for satisfiability over the constraint theory. We show that this constrained resolution is sound and complete in that a set of clauses with constraints is unsatisfiable over the constraint theory iff we can deduce a constrained empty clause for each model of the constraint theory, such that the empty clauses constraint is satisfiable in that model. We show also that we cannot require a better result in general, but we discuss certain tractable cases, where we need at most finitely many such empty clauses or even better only one of them as it is known in classical resolution, sorted resolution or resolution with theory unification.

DFKI Technical Memos

TM-89-01
Susan Holbach-Weber
Connectionist Models and Figurative Speech
27 pages

Abstract: In my three month stay at DFKI as a guest researcher, from July 1989 to August 1989, I presented a one hour seminar on my thesis research, a five hour tutorial ("Kompaktkurs") introduction to connectionist models, and a one hour lecture at the University of Stuttgart. I also arranged for DFKI to acquire the Rochester Connectionist Simulator, and worked on a paper describing my thesis work for publication as a chapter in a book on connectionist approaches to metaphor. This report describes first the content of the Kompaktkurs, then the content of the paper on my thesis work.

TM-89-02 (Reprint)
Harold Boley
Expert System Shells: Very-High-Level Languages For Artificial Intelligence
17 pages

Abstract: Expert-system shells are discussed as very-high-level programming languages for knowledge engineering. Based on a category/domain distinction for expert systems the concept of expert-system shells is

explained using seven classifications. A proposal for a shell-development policy is sketched. The conclusions express concern about over-emphasis on shell surfaces.

TM-90-01 (Preprint)
Elisabeth André, Thomas Rist
Ein planbasierter Ansatz zur Synthese illustrierter Dokumente
13 Seiten

Zusammenfassung: Obwohl die Erzeugung illustrierter Dokumente in der KI-Forschung zunehmendes Interesse findet, werden in den meisten Systemen Text und Graphik weitgehend unabhängig voneinander aufgebaut und stehen daher beziehungslos nebeneinander. In dieser Arbeit wird von der Überlegung ausgegangen, daß nicht nur die Erzeugung von Texten, sondern auch die Synthese illustrierter Dokumente als kommunikative Handlung zur Erreichung von Zielen aufgefaßt werden kann. Für die Realisierung eines Systems, das selbstständig illustrierte Dokumente erstellt, bietet sich daher ein planbasierter Ansatz an. Es wird zunächst gezeigt, daß die in der Textlinguistik gebräuchliche Unterscheidung zwischen Haupt- und Neben-handlungen auch für Text-Bild-Kombinationen geeignet ist. Von dieser Unterscheidung ausgehend werden Strategien formuliert, die sich sowohl auf die Erzeugung von Text als auch auf den Aufbau von Bildern beziehen. Die gemeinsame Planung von Text und Bild wird als grundlegende Voraussetzung angesehen, die beiden Modi in einem Dokument aufeinander abzustimmen.

TM-90-02 (Preprint)
Thomas Rist, Elisabeth André
Wissensbasierte Perspektivenwahl für die automatische Erzeugung von 3D-Objektdarstellungen
9 Seiten

Zusammenfassung: Aus welcher Perspektive ein Objekt gezeigt werden soll, ist eine der elementaren Fragen, die sich bei der automatischen Erzeugung von 3D-Darstellungen stellt, die aber in den wenigen Systemen, die graphische Objektdarstellungen selbständig planen, bisher vernachlässigt wurde. Ziel der vorliegenden Arbeit ist es, aufzuzeigen, wie sich Wissen über Objekte und Darstellungstechniken verwenden läßt, um die Menge der möglichen Perspektiven, aus denen ein Objekt gesehen und gezeigt werden kann, sinnvoll einzuschränken. Als Grundlage zur Perspektivenwahl schlagen wir ein Bezugssystem vor, das eine Einteilung der Perspektiven in 26 Klassen nahelegt und das darüberhinaus Vorteile bietet, wenn gewählte Perspektiven natürlichsprachlich zu beschreiben sind. Anschließend führen wir einige für die Perspektivenwahl relevanten Kriterien an. Diese Kriterien werden dann zur Formulierung von Regeln herangezogen, die wir dazu verwenden, um in einer konkreten Präsentationssituation eine geeignete Perspektive zu bestimmen.

Terminological Cycles in KL-ONE-based Knowledge Representation Languages

Franz Baader

Research Report

RR-90-01