LTCS–Report

Aachen University of Technology
Research group for
Theoretical Computer Science

Characterizing the semantics of terminological cycles in $\mathcal{ALN}$ using finite automata

Ralf Küsters

LTCS-Report 97-04

RWTH Aachen
LuFg Theoretische Informatik
http://www-lti.informatik.rwth-aachen.de

Ahornstr. 55
52074 Aachen
Germany

Characterizing the semantics of terminological cycles in $\mathcal{ALN}$ using finite automata

Ralf Küsters

Abstract

The representation of terminological knowledge may naturally lead to terminological cycles. In addition to descriptive semantics, the meaning of cyclic terminologies can also be captured by fixed-point semantics, namely, greatest and least fixed-point semantics. To gain a more profound understanding of these semantics and to obtain inference algorithms as well as complexity results for inconsistency, subsumption, and related inference tasks, this paper provides automata theoretic characterizations of these semantics. More precisely, the already existing results for $\mathcal{FL}_0$ are extended to the language $\mathcal{ALN}$, which additionally allows for primitive negation and number-restrictions. Unlike $\mathcal{FL}_0$, the language $\mathcal{ALN}$ can express inconsistent concepts, which makes non-trivial extensions of the characterizations and algorithms necessary. Nevertheless, the complexity of reasoning does not increase when going from $\mathcal{FL}_0$ to $\mathcal{ALN}$. This distinguishes $\mathcal{ALN}$ from the very expressive languages with fixed-point operators proposed in the literature. It will be shown, however, that cyclic $\mathcal{ALN}$-terminologies are expressive enough to capture schemata in certain semantic data models.

Chapter 1

Introduction

Description logics (DLs) and the corresponding DL-systems can be used to capture the terminological knowledge of a problem domain in a formally well-defined way. In such representation formalisms the concepts of the domain are defined in terminologies by concept definitions. The definitions are complex terms constructed by atomic concepts and roles as well as concept constructors. A terminology T is a finite set of concept definitions of the form $A = C$ where A denotes an atomic concept and C a concept term. The atomic concept A is called *defined concept.* If there is no such concept definition in T for a given atomic concept, then this atomic concept is called *primitive concept.* Formally, atomic concepts are interpreted as unary relations and the roles are binary relations over the domain of the interpretation. The interpretation is extended to concept terms such that concept terms describe certain subsets of the domain. An example of a terminology consisting only of one concept definition is

$$\mathsf{Human} \quad = \quad \mathsf{Mammal} \sqcap (\geq 2\ \mathsf{parents}) \sqcap (\leq 2\ \mathsf{parents}) \sqcap \forall \mathsf{parents}.\mathsf{Human} \tag{1.1}$$

Intuitively, the defined concept Human contains all individuals which are mammals with exactly two parents all of whom are human beings. Thus, all ancestors of an individual in Human (transitive closure of the role parents) should be mammals with exactly two parents.

Intuitively, it is expected that the interpretation of a defined concept of a terminology is uniquely determined by the interpretation of the primitive concepts and roles (primitive interpretation). This is the case for acyclic terminologies since by macro extension every defined concept can be described by a concept term without defined concepts. In general, for cyclic terminologies (see, e.g., (1.1)) there are several possible extensions of given primitive interpretation to a model of the considered terminology. Therefore, beside the descriptive semantics, which is used to describe the semantics of acyclic terminologies, also fixed-point semantics, namely, greatest and least fixed-point semantics, are employed to capture the meaning of cyclic terminologies.

Cyclic terminologies were first investigated by B. Nebel [17, 18, 19], who has introduced the mentioned fixed-point semantics in addition to descriptive semantics.

	Subsumption general	Inconsistency general	Inconsistency (weak-)acyclic
descriptive	PSPACE-complete	PSPACE-complete	NP-complete
gfp	PSPACE-complete	PSPACE-complete	NP-complete
lfp	PSPACE-complete	PSPACE-complete	NP-complete

Table 1.1: Complexity of inference problems in $\mathcal{ALN}$-terminologies

Which of the semantics should be prefered depends on the particular representation problem at hand. Nebel has shown—by proving the finite model property—that for the language $\mathcal{ALN}$, which allows for concept conjunction, (universal) value-restriction, primitive negation, and unqualified number-restrictions, subsumption w.r.t. cyclic terminologies and the descriptive semantics is decidable.

In order to gain a more profound understanding of all three semantics as well as more feasible decision algorithm and complexity results for subsumption, F. Baader [4] has proposed automata theoretic characterizations of the three semantics for the small representation language $\mathcal{FL}_0$, which allows for concept conjunction and (universal) value-restrictions. Following this approach, B. Nebel [20] has given an automata theoretic characterization of equivalence of concepts w.r.t. cyclic $\mathcal{FL}_0$-terminologies. Although [20] introduces the semantics for $\mathcal{ALN}$, the results are restricted to the sub-language $\mathcal{FL}_0$.

Since $\mathcal{FL}_0$ is not expressive enough for most practical representation problems, in this paper the results for $\mathcal{FL}_0$ are extended to the language $\mathcal{ALN}$. Generalizing the results from $\mathcal{FL}_0$ to $\mathcal{ALN}$ is not trivial due to inconsistent concepts, which are expressible in $\mathcal{ALN}$, but not in $\mathcal{FL}_0$. The new complexity results presented in this paper are summarized in table 1.1 for all three semantics both for general (i.e., possibly cyclic) and (weak-)acyclic terminologies.

Terminological cycles in much more expressive extensions of $\mathcal{FL}_0$ have been investigated in [22] and [13]. K. Schild has extended the language $\mathcal{ALC}$ by the fixed-point operators of the μ-calculus to $\mu\mathcal{ALC}$and has shown—among other results—that $\mu\mathcal{ALC}$ is more expressive than general $\mathcal{ALC}$-terminologies.[1] Moreover, the language $\mu\mathcal{ALC}$ has been extended in [13] by (qualified) number-restrictions to $\mathcal{ALCQ}\mu$. Thus, the language $\mathcal{ALCQ}\mu$ contains $\mathcal{ALN}$. Consistency as well as subsumption for $\mathcal{ALCQ}\mu$-concepts is EXPTIME-complete, whereas these problems are merely PSPACE-complete for general $\mathcal{ALN}$-terminologies (see table 1.1), which justifies to consider this restricted case separately. For both $\mathcal{ALCQ}\mu$ and $\mathcal{ALN}$, the important inference problems can be decided with the help of finite automata. However, the automata for $\mathcal{ALCQ}\mu$ are of exponential size and they are tree automata reflecting certain semantic structures, while the automata for $\mathcal{ALN}$ are finite automata on words that are merely syntactic variants of $\mathcal{ALN}$-terminologies.

Cycles can increase the complexity of inference problems or even lead to undecidability. For instance, subsumption for general $\mathcal{FL}_0$-terminologies which allow

[1] To ensure the existence of least and greatest fixed-point models, recursively defined concepts must occur positively in their definition.

for feature agreements is undecidable [20], whereas this problem is decidable for acyclic terminologies.[2] Furthermore, in this paper it is shown that inconsistency w.r.t. acyclic $\mathcal{ALN}$-terminologies is NP-complete, whereas it is PSPACE-complete for general $\mathcal{ALN}$-terminologies (see table 1.1).

Nevertheless, the relevant literature points out the need for cyclic definitions of concepts (see, e.g., [16, 7, 8]). Example (1.1) reveals that cyclic terminologies can be natural descriptions of terminological knowledge. Additionally, they allow for describing properties of concepts using the transitive closure of relations (parents in the example). Furthermore, description logics can be used to describe schemas w.r.t. most semantic and object-oriented data models (e.g., [10, 6, 5]). These schemas are often cyclic, e.g., the Boss of an Employee is a Manager who is himself an Employee. Unlike terminologies, which consist of concept definitions, schemas state only necessary (rather than necessary and sufficient) conditions for concepts. In this paper, however, it is shown that the important inference problems for so-called (cyclic) $\mathcal{SL}_{dis}$-schemas, which where introduced in [7], can be reduced to corresponding problems for (cyclic) $\mathcal{ALN}$-terminologies. Hence, the decision algorithms for terminologies presented below can be used to decide inference problems for schemas. This yields new proofs for the upper bounds complexity for such schemas, and in fact, for a more expressive schema language, which allows for arbitrary number-restrictions. Conversely, existing results for lower complexity bounds for schemas will be used to derive complexity results for terminologies.

In the following chapter we will recall some definitions and results concerning ordinals, fixed-points, and automata. A formal introduction to general $\mathcal{ALN}$-terminologies is given in chapter 3, which also motivates the mentioned fixed-point semantics. The heart of this paper are the following three chapters, which deal with the characterization of the three semantics as well as the important inference problems inconsistency and subsumption. Due to inconsistency, which in $\mathcal{ALN}$ in contrast to $\mathcal{FL}_0$ also occurs w.r.t. the greatest fixed-point semantics and the descriptive semantics, the notion of "exclusion" and "exclusion sets" are introduced. These notions will turn out to be very useful for the characterization of subsumption and inconsistency as well as for deciding these inference problems. In chapter 7 the already mentioned relationship between schemas and terminologies is investigated; also (weak-)acyclic terminologies and schemas will be considered. Finally, we will summarize and discuss the results in the conclusion.

[2] One can use the technique described in [1] to obtain this result.

Chapter 2

Preliminaries

For the readers' convenience we recall some definitions and properties involving ordinals, fixed-points, and finite automata.

2.1 Ordinals

A more thorough introduction to ordinals can be found in [21].

A *linear ordering* $<$ over a domain D is a binary, transitive, total[1], and irreflexive relation over D. Such an ordering is called *well-founded* if, in addition, every non-empty sub-ordering[2] of $<$ contains a least element. An *order type* is an equivalence class of isomorphic, linear orderings. An *ordinal* denotes an order type of isomorphic, well-founded orderings. One can define a well-founded ordering on the set of ordinals. We define such an ordering $<$ as follows: $\alpha < \beta$ iff α is a proper initial segment of β. Since this defines a well-founded ordering, there is a least element for every subset of ordinals as well as a least upper bound (i.e., a least element of the set of upper bounds). Furthermore, no such subset contains an infinite decreasing chain of ordinals.

2, 17, 42 are examples of *finite* ordinals, where, e.g., 17 denotes the order type $\{0, 1, 2, \ldots, 16\}$ for the usual ordering $<$ on non-negative integers. The ordinal ω is the least order type with an infinite number of elements, i.e., the set $\{0, 1, 2, \ldots\}$ with the usual ordering $<$ on non-negative integers. It holds $n < \omega$ for each ordinal n.

For the ordinal α the ordinal $\alpha + 1$ is called *successor* of α. An ordinal which is successor of one ordinal is a *successor ordinal*, otherwise we call it *limit ordinal*. Because $17 = 16 + 1$, e.g., 17 is a successor ordinal, whereas ω is a limit ordinal. The successor ordinal $\omega + 1$ of ω has the order type $\{0, 1, 2, \ldots\} \cup \{\infty\}$ where every element in $\{0, 1, 2, \ldots\}$ is less than ∞. One can obtain a limit ordinal α as least upper bound (*lub*) of the set of all less ordinals, i.e., $\alpha = lub(\{\beta \mid \beta < \alpha\})$.

[1] For each $a, b \in D$, $a \neq b$, it holds $a < b$ or $b < a$.

[2] A relation $<_S$ is called sub-ordering of $<$ if there is a subset S of D such that $<_S = < \cap S \times S$.

2.2 Fixed-points

The definitions and results of this section can also be found in [15]. A partial ordering[3] D is called *complete lattice*[4] if every sub-ordering C of D has a *least upper bound* $lub(C)$ in D. In this case, since $glb(C) = lub(\{d \in D \mid d \text{ is lower bound of } C\})$, there exits also a greatest lower bound for C. With that, D contains a *least element*, *bottom* $= lub(\emptyset)$, and a *greatest element*, *top* $= glb(\emptyset)$. In order to define fixed-point semantics we need the following complete lattice.

Example 1.
Let $D = 2^S \times \cdots \times 2^S$ be the n-fold cartesian product where 2^S denotes the set of all subsets of S. The set D is ordered component-wise by inclusion: $(A_1, \ldots, A_n) \subseteq (B_1, \ldots, B_n)$ iff $A_1 \subseteq B_1, \ldots, A_n \subseteq B_n$. With that, D is a complete lattice; *top* $= (S, \ldots, S)$ and *bottom* $= (\emptyset, \ldots, \emptyset)$. $\diamond$

For the partial ordering D the mapping $T : D \longrightarrow D$ is *monotonic* iff for all $a, b \in D$, $a \leq b$ implies $T(a) \leq T(b)$. An element $f \in D$ where $T(f) = f$ is called f fixed-point of T. Let D be a complete lattice and $T : D \longrightarrow D$ monotonic. Then T has a least fixed-point $lfp(T)$ and a greatest fixed-point $gfp(T)$, which are possibly identical. Every fixed-point of T lays in between the least and greatest fixed-point ([21], Proposition 5.1). Least and greatest fixed-points are expressible in terms of ordinal powers. The ordinal powers $T{\uparrow}^{\alpha}$ and $T{\downarrow}^{\alpha}$ are inductively defined:

i.) $T{\uparrow}^{0} :=$ *bottom* and $T{\downarrow}^{0} :=$ *top*;

ii.) $T{\uparrow}^{\alpha+1} := T(T{\uparrow}^{\alpha})$ and $T{\downarrow}^{\alpha+1} := T(T{\downarrow}^{\alpha})$;

iii.) if α is a limit ordinal, then $T{\uparrow}^{\alpha} := lub(\{T{\uparrow}^{\beta}; \beta < \alpha\})$ and $T{\downarrow}^{\alpha} := glb(\{T{\downarrow}^{\beta}; \beta < \alpha\})$.

Proposition 2.
Let D be a complete lattice and $T : D \longrightarrow D$ a monotonic mapping. Then for any ordinal α it holds: $T{\uparrow}^{\alpha} \leq lfp(T)$ and $T{\downarrow}^{\alpha} \geq gfp(T)$. In addition, there exist ordinals β and γ such that $T{\uparrow}^{\beta} = lfp(T)$ and $T{\downarrow}^{\gamma} = gfp(T)$.

Proof. see [15], Proposition 5.3. $\square$

The mapping T is called *upward ω-continuous* (resp., *downward ω-continuous*) iff for any increasing chain $d_0 \leq d_1 \leq d_2 \leq \cdots$ (resp., decreasing chain $d_0 \geq d_1 \geq d_2 \geq \cdots$) it holds: $T(lub(\{d_i \mid i \geq 0\}) = lub(\{T(d_i) \mid i \geq 0\})$ (resp., $T(glb(\{d_i \mid i \geq 0\}) = glb(\{T(d_i) \mid i \geq 0\}))$. It is easy to see that in this case T is monotonic. Furthermore, the ordinals β and γ in the above proposition can be chosen less or equal ω. More precisely, we have

[3] reflexive, transitive, anti-symmetric binary relation

[4] D shall denote both domain and ordering.

Proposition 3.
Let D be a complete lattice and $T : D \longrightarrow D$ a upwards ω-continuous (resp., downward ω-continuous) mapping. Then $lfp(T) = T{\uparrow}^{\omega} = lub(\{T^n(bottom) \mid n \geq 0\})$ (resp., $gfp(T) = T{\downarrow}^{\omega} = glb(\{T^n(top) \mid n \geq 0\})$).
Proof. Consequence of Proposition 4. □

The following Proposition is a generalization of Proposition 3 for downward ω-continuous mappings. An analogous statement can be proven for upward ω-continuous mappings, but this is not needed in the sequel.

Proposition 4.
Let D be a complete lattice and $T : D \longrightarrow D$ a downward ω-continuous mapping. Furthermore, let d be an element in D with $d \geq T(d)$. Then $d\text{-}gfp(T) := glb(\{T^n(d) \mid n \geq 0\})$ is the greatest fixed-point of T which is less or equal d.
Proof. see [4] □

2.3 Automata and languages

In this section we recall some notions and statements of finite automata and languages. More detailed informations can be found in [14, 12].

For a finite alphabet Σ the set of all finite words over Σ is denoted Σ^*. Let $W \in \Sigma^*$ where $W = a_0 \cdots a_{n-1}, a_i \in \Sigma$, $0 \leq i \leq n-1$, $n \in \mathbb{N}$. Then $|W| := n$ is the length of W. We denote the empty word, i.e., the word with length zero, ε. Furthermore, Σ_ε denotes the set $\Sigma \cup \{\varepsilon\}$. The finite word W can be seen as a mapping of the ordinal $n = \{0, \ldots, n-1\}$ into Σ: $W(i) := a_i$ for all $0 \leq i < n$. An *infinite word* (*ω-word*) is a mapping of the ordinal ω into Σ. The set of all infinite words over Σ is denoted Σ^ω. An ω-word W is also written $W(0)W(1)W(2)\cdots$. For the language $L \subseteq \Sigma^*$ and the letter a we define $L \cdot a := \{W \cdot a \mid W \in L\}$. The set L^ω contains exactly those ω-words over Σ which are of the form $W_1W_2W_3\cdots$ where $W_i \in L$ for all $i \geq 1$.

A *semi-automaton* (with word-transitions) is a triple $\mathcal{A} = (\Sigma, Q, E)$, which consists of a finite alphabet Σ, a finite set of states Q, and a finite set of transitions $E \subseteq Q \times \Sigma^* \times Q$. If $E \subseteq Q \times \Sigma_\varepsilon \times Q$, then $\mathcal{A}$ is called *semi-automaton without word-transitions.* If $E \subseteq Q \times \Sigma \times Q$, then $\mathcal{A}$ is called *semi-automaton with letter-transitions.*

Let $\mathcal{A}$ be a semi-automaton and p, q states of $\mathcal{A}$. There is a *finite path* of length n from p to q in $\mathcal{A}$ with label U if there are transitions (p_{i-1}, U_i, p_i) in $\mathcal{A}$ with $p_0 = p$, $p_n = q$, and $U = U_1 \cdots U_n$. For $n = 0$ this is an *empty path* labeled with $U = \varepsilon$ and $q = p$. The sequence $q_0, V_1, q_1, V_2, q_2, \ldots, V_n, q_n$ with $q_i \in Q$ for all $0 \leq i \leq n$ and $V_i \in \Sigma^*$ for all $1 \leq i \leq n$ denotes a finite path from q_0 to q_n if for all $1 \leq i \leq n$ there is a finite path from q_{i-1} to q_i with label V_i. Note, that (q_{i-1}, V_i, q_i) need not to be a transition in $\mathcal{A}$, otherwise this will be mentioned explicitly. Analogously, the sequence $q_0, V_1, q_1, V_2, q_2, V_3, \ldots$ denotes an infinite path starting from q_0 and labeled with $W = V_1V_2V_3\cdots$ if for all $i \geq 1$, there is a finite path from q_{i-1} to q_i labeled with V_i. Additionally, it is required that for an infinite number of indices $i \geq 1$ the path from q_{i-1} to q_i is non-empty. The word W is possibly finite or infinite. For a finite word W there is an index $i \geq 0$ such that q_i lays on an ε-cycle—due to the fact that Q is finite. Thus, the sequence $q_0, W, q_i, \varepsilon, q_i, \varepsilon, q_i, \ldots$ denotes an infinite path in $\mathcal{A}$ labeled with W starting from q_0. A state q of $\mathcal{A}$ lays on an ε-cycle if there is a non-empty path in $\mathcal{A}$ from q to q with label ε.

Let I and J be subsets of Q. Then $L_{\mathcal{A}}(I, J)$ (resp., $L(I, J)$ if the relationship to $\mathcal{A}$ is clear from the context) denotes the set of finite words over Σ which are labels of finite paths starting from a state in I and terminating in one state in J. Note that $L_{\mathcal{A}}(I, J)$ is a regular language. For sets $I = \{p\}$ and $J = \{q\}$ we write $L_{\mathcal{A}}(p, q)$ (resp., $L(p, q)$) instead of $L_{\mathcal{A}}(I, J)$ (resp., $L(I, J)$).

Furthermore, we will consider labels of infinite paths in the sequel. Let p be a state of $\mathcal{A}$. Then we define $U_{\mathcal{A}}(p)$ resp. $U(p) := \{W \in \Sigma^* \cup \Sigma^\omega \mid W$ is a label of an infinite path starting from $p\}$.

In order to simulate an equivalent deterministic automaton of $\mathcal{A}$ we need some more definitions. Let $\mathcal{A} = (\Sigma, Q, E)$ be a semi-automaton, then the ε-closure of $I \subseteq Q$ is defined by

$$\varepsilon\text{-}closure_{\mathcal{A}}(F) := \{q' \mid \text{there is a state } q \text{ in } F \text{ and a (possibly empty) } \varepsilon\text{-path from } q \text{ to } q' \text{ in } \mathcal{A}\}.$$

The successor set of I w.r.t. $a \in \Sigma$ is

$$next_{\mathcal{A}}(I, a) := \{q \in Q \mid \text{there is a state } q' \in I \text{ where } (q', a, q) \in E\}.$$

Finally, for $W \in \Sigma^*$ we define inductively

$$\begin{aligned} next_{\varepsilon\mathcal{A}}(I, \varepsilon) &:= \varepsilon\text{-}closure_{\mathcal{A}}(I) \text{ and} \\ next_{\varepsilon\mathcal{A}}(I, aW) &:= next_{\varepsilon\mathcal{A}}(next_{\mathcal{A}}(\varepsilon\text{-}closure_{\mathcal{A}}(I), a), W) \end{aligned}$$

the successor set of a set of states and a word. If the relation to $\mathcal{A}$ is clear from the context we write ε-$closure(I)$, $next(I, a)$ and $next_\varepsilon(I, W)$, respectively. This sets are computable in time polynomial in the size of $\mathcal{A}$ (and W).

The following lemma is easy to prove by induction on the length n of W.

Lemma 5.
Let $\mathcal{A} = (\Sigma, Q, E)$ be a semi-automaton without word-transitions, $W \in \Sigma^*$, and q, q' states in Q. Then it holds:

$$q' \in next_\varepsilon(\{q\}, W) \text{ iff } W \in L(q, q').$$

□

In Lemma 5 it is necessary to assume a semi-automaton *without* word-transitions. Consider for example a semi-automaton with the states q_0, q_1 and the transition (q_0, ab, q_1). Then $ab \in L(q_0, q_1)$, but $q_1 \notin next_\varepsilon(\{q_0\}, ab) = \emptyset$. For this purpose, $next(I, a)$ and $next_\varepsilon(I, A)$ always refer to semi-automaton without word-transitions. With respect to the introduced languages $L(I, J)$ and $U(q)$ every semi-automaton can be reduced (w.l.o.g.) to a semi-automaton without word-transitions.

Lemma 6.
Let $\mathcal{A} = (\Sigma, Q, E)$ be a semi-automaton (with word-transitions). Then there exists a semi-automaton $\mathcal{B} = (\Sigma, Q', E')$ without word-transitions and with the following properties:

i.) $Q \subseteq Q'$;

ii.) For all $q \in Q' \setminus Q$ there is no empty path from q to q which only contains states in $Q' \setminus Q$;

iii.) no state $q \in Q' \setminus Q$ lays on an ε-cycle;

iv.) for all $I, J \subseteq Q$ it holds: $L_{\mathcal{A}}(I, J) = L_{\mathcal{B}}(I, J)$;

v.) for all $q \in Q$ it holds: $U_{\mathcal{A}}(q) = U_{\mathcal{B}}(q)$.

Proof. One can construct $\mathcal{B}$ as follows: Every word-transition $(p, a_1 \cdots a_n, q) \in E$ where $n > 1$ is substituted by the transitions $(p, a_1, p_1), (p_1, a_2, p_2), \ldots, (p_{n-1}, a_n, q)$ where $p_1, \ldots, p_{n-1}$ are new states. The other transitions in E are added to E' without change. Now the above properties can easily be proven. □

For the language $U_{\mathcal{A}}(q)$ we need

Lemma 7.
Let $\mathcal{A}$ be a semi-automaton without word-transitions, q a state in $\mathcal{A}$, and $W = a_1 a_2 a_3 \cdots$ an ω-word. For $T_i := next_\varepsilon(q, a_1 \cdots a_i)$, $i \geq 0$, it holds: $W \notin U(q)$ iff there is a $k \geq 0$ with $T_k = \emptyset$; in this case it is $T_i = \emptyset$ for all $i \geq k$.

Proof. "$\Rightarrow$": Let $T_i \neq \emptyset$ for all $i \geq 0$. Thus, for all $i \geq 0$ there is a state $q_i \in T_i$ such that $W_i := a_1 \cdots a_i \in L(q, q_i)$. We consider the following tree: The root of the tree is labeled with q. Successor nodes of q are exactly those nodes labeled with q' such that there is an $\mathcal{A}$-path from q to q' with label a_1. The node labeled with q' has a successor node labeled with q'' iff there is a path from q' to q'' with label a_2. Analogously, one defines successor nodes for $a_3, a_4, a_5, \ldots$ Since Q is a finite set, the defined tree is finitely branched. Because of $W_i \in L(q, q_i)$ for all $i \geq 0$ and Lemma 5 the tree contains paths of arbitrary length. As a consequence of König's Lemma, the tree contains an infinite path. Hence, $W \in U(q)$.

"$\Leftarrow$": Let $k \geq 0$ with $T_k = \emptyset$. Assume $W \in U(q)$. Thus, there are states $q_1, q_2, q_3, \ldots$ such that $q = q_0, a_1, q_1, a_2, q_2, \ldots$ is a infinite path in $\mathcal{A}$. Consequently, $q_l \in T_l$ for all $l \geq 0$ (Lemma 5). This is a contradiction since $T_k = \emptyset$. □

Chapter 3

Cyclic $\mathcal{ALN}$-terminologies

In this chapter we formally introduce general $\mathcal{ALN}$-terminologies as well as the descriptive semantics and fixed-point semantics for such terminologies. First, we define syntax and descriptive semantics of $\mathcal{ALN}$-terminologies.

3.1 $\mathcal{ALN}$-terminologies

The syntax of $\mathcal{ALN}$-concepts and $\mathcal{ALN}$-terminologies is defined in

Definition 8 (syntax).
Let N_d, N_p, and N_r be pairwise disjoint sets. The sets N_d, N_r contain concept names and N_r contains role names. $\mathcal{ALN}$*-concepts* are inductively defined as follows:

i.) Every $C \in N_d$ is an *(atomic) concept*.

ii.) Every $P \in N_p$ is an *(atomic) concept* and $(\neg P)$ is a concept (*primitive negation*).

iii.) For every $n \in \mathbb{N}$ and $R \in N_r$ the terms $(\geq n\ R)$ (*maximum-restriction*) and $(\leq n\ R)$ (*minimum-restriction*) are concepts (*number-restrictions*).

Let C, D be concepts and $R \in N_r$. Then

iv.) $C \sqcap D$ (*concept conjunction*) and

v.) $\forall R.C$ *((universal) value-restriction)* are concepts.

A *(general)* $\mathcal{ALN}$*-terminology* T consists of a finite set of concept definitions of the form $A = D$ where A denotes an atomic concept in N_d and D an $\mathcal{ALN}$-concept. In addition, it is required that for every atomic concept $A \in N_d$ in T there is at most one concept definition with left-hand side A. We call atomic concepts appearing on the left-hand side of some concept definition in T *defined*. Atomic concepts $P \in N_p$ occurring in T are called *primitive*. For such atomic concepts there are no concept definitions in T. For an $\mathcal{ALN}$-terminology T only primitive negation is allowed, i.e., if $\neg P$ is a sub-concept in T then P is a primitive concept. $\diamond$

Beside $\mathcal{ALN}$ we also consider some sub-languages. The language $\mathcal{FL}_0$ allows for universal value-restriction and concept conjunction. The language $\mathcal{FLN}$ augment $\mathcal{FL}_0$ by unqualified number-restrictions. The language $\mathcal{AL}$ extends $\mathcal{FL}_0$ by primitive negation and allows for maximum-restrictions of the form $(\geq 1\ R)$. Analogous to $\mathcal{ALN}$ we define $\mathcal{FL}_0$-, $\mathcal{AL}$-, and $\mathcal{FLN}$-terminologies.

The following examples reveal that cyclic definitions of concepts can be a natural way to define concepts. Furthermore, they show that the (reflexive-)transitive closure of relations are expressible in cyclic terminologies. A (reflexive-)transitive closure is defined as follows: Let $R \subseteq M \times M$ be a binary relation on a set M. We define $R^0 := \{(e,e); e \in M\}$ and $R^{n+1} := R \circ R^n, n \geq 0$, where "$\circ$" denotes the composition of binary relations. Then $\bigcup_{n\geq 1} R^n$ is the *transitive closure* of R and $\bigcup_{n\geq 0} R^n$ the *reflexive-transitive closure.*

Example 9.
Let Mammal, Human, Male, Woman, and Man be atomic concepts and let parents be a role. The $\mathcal{ALN}$-terminology T is defined as follows:

$$\begin{aligned}
\mathsf{Human} &= \mathsf{Mammal} \sqcap (\geq 2\ \mathsf{parents}) \sqcap (\leq 2\ \mathsf{parents}) \sqcap \forall \mathsf{parents}.\mathsf{Human} \\
\mathsf{Man} &= \mathsf{Human} \sqcap \mathsf{Male} \\
\mathsf{Woman} &= \mathsf{Human} \sqcap \neg \mathsf{Male}
\end{aligned}$$

Intuitive, in this terminology a human being is defined as a mammal with exactly two parents all of whom are human beings, i.e., all ancestors (transitive closure of parents) of a human being are them self mammals with exactly two parents. A man is defined as male human being. A woman is defined as human being which is not male. $\diamond$

A second example:

Example 10.
Let Binary-tree, Ternary-tree, Leaves and Tree be atomic concepts and direct-successor a role. Then

$$\begin{aligned}
\mathsf{Binary\text{-}tree} &= \mathsf{Tree} \sqcap (\leq 2\ \mathsf{direct\text{-}successor}) \sqcap \\
&\quad \forall \mathsf{direct\text{-}successor}.\mathsf{Binary\text{-}tree} && (3.1) \\
\mathsf{Ternary\text{-}tree} &= \mathsf{Tree} \sqcap (\leq 3\ \mathsf{direct\text{-}successor}) \sqcap \\
&\quad \forall \mathsf{direct\text{-}successor}.\mathsf{Ternary\text{-}tree} && (3.2) \\
\mathsf{Leaves} &= \mathsf{Tree} \sqcap (\leq 0\ \mathsf{direct\text{-}successor}) && (3.3)
\end{aligned}$$

is an $\mathcal{ALN}$-terminology with defined concepts Binary-tree, Ternary-tree and Leaves. The atomic concept Tree is primitive. An individual is instance of Binary-tree if it is an instance of Tree, has at most two direct-successors, and all (direct and indirect) successors (transitive closure of direct-successors) are instances of Binary-tree. The atomic concept Ternary-tree is defined analogously. The atomic concept Leaves describes those individuals which do not have successors. $\diamond$

Until now we only refer to the intuitive semantics of terminologies. The formal semantics is defined as follows.

Definition 11 (semantics).
An *interpretation* I consists of a set $dom(I)$ of individuals and objects, *domain of* I, and a function $\cdot^I$, which maps every atomic concept A to a subset A^I of $dom(I)$ and every role R to a binary relation R^I over $dom(I)$, i.e., $R^I \subseteq dom(I) \times dom(I)$. The sets A^I and R^I are called *extensions* of A and R w.r.t. I.
The interpretation function of I is extended to concepts:

i.) $(\neg P)^I := dom(I) \setminus P^I$ for every $P \in N_p$;

ii.) for $R \in N_r$ and $d \in dom(I)$ let $R^I(d) := \{e; (d,e) \in R^I\}$ be the set of *role fillers* of d w.r.t. R and I. Then for $n \in \mathbb{N}$ the number-restrictions are interpreted $(\geq n\ R)^I := \{d \in dom(I);\ |R^I(d)| \geq n\}$ and $(\leq n\ R)^I := \{d \in dom(I);\ |R^I(d)| \leq n\}$.

Assume that for the concepts C, D the sets C^I, D^I are already defined. Let $R \in N_r$, then

iii.) $(C \sqcap D)^I := C^I \cap D^I$ and

iv.) $(\forall R.C)^I := \{d \in dom(I); R^I(d) \subseteq C^I\}$.

This defines the interpretation function of I for all concepts.

Two concepts C and D are *equivalent* if $C^I = D^I$ for all interpretations I.

An interpretation I is *model* of a terminology T (T-*model*) iff $A^I = D^I$ for all concept definitions $A = D$ in T.[1]

The *descriptive semantics* of T is defined by the set of all T-models. Two terminologies are *equivalent* w.r.t. the descriptive semantics if their descriptive semantics coincide. ◇

The characterization of the descriptive semantics will reveal that this semantics is not always appropriate to capture the intuition of a terminology. Furthermore, the interpretation of the primitive concepts and roles (the so-called primitive interpretation) may have different possible extensions for the defined concepts.

Both problems arise if the terminology at hand is cyclic. In this case also other semantics are considered.

3.2 Semantics of cyclic $\mathcal{ALN}$-terminologies

As pointed out in the preceding chapter there are some problems concerning the descriptive semantics with respect to cyclic terminologies. In this chapter, we will address this—and other—problems of the descriptive semantics in more detail. This

[1] Obviously, it is sufficient if I only interprets concepts and roles occurring in T.

leads us to alternative semantics whose advantages and disadvantages we will briefly discuss.

First, we will define the already mentioned notions "terminological cycle" and "primitive interpretation".

Definition 12 (terminological cycle).
Let T be an ($\mathcal{ALN}$-)terminology, A a defined concept, and B an atomic concept in T. The concept A 'directly uses' B if B occurs on the right-hand side of the concept definition $A = C$ in T. Let 'uses' denote the transitive closure of 'directly uses'. Then T is *cyclic* (contains a *terminological cycle*) iff there exists a defined concept A in T that uses itself; otherwise T is called *acyclic*. ◇

The terminologies in the examples 9 and 10 are cyclic.

Definition 13 (the primitive interpretation and its extension).
Let T be a terminology, $P_1, \ldots, P_m$ the primitive concepts, $R_1, \ldots, R_k$ the roles, and $A_1, \ldots, A_n$ the defined concepts in T. A *primitive interpretation* J consists of the domain $dom(J)$, and the extensions of the primitive concepts $(P_1^J, \ldots, P_m^J)$ and the extension of the roles $(R_1^J, \ldots, R_k^J)$.
An interpretation I of T *extends* J iff $dom(I) = dom(J)$, $P_1^I = P_1^J, \ldots, P_m^I = P_m^J$ and $R_1^I = R_1^J, \ldots, R_k^I = R_k^J$. Therefore, I is uniquely determined by J and the n-tuple $\underline{A} = (A_1^I, \ldots, A_n^I) \in (2^{dom(J)})^n$. We call J the *corresponding primitive interpretation* of I. For a defined concept B the position of B in the tuple $\underline{A}$ is denoted by $index(B) \in \{1, \ldots, n\}$. The ith component of tuple $\underline{A}$ is $(\underline{A})_i$. ◇

In [4] it is pointed out that involving cyclic terminologies one can in general not uniquely extend the primitive interpretation to a model of a terminology. Depending on the intuitive semantics different semantics may be prefered. For acyclic terminologies there is always a unique extension of the primitive interpretation to a model of the considered terminology.

For a model I of a terminology one can define the extensions of the defined concepts as fixed-point of a mapping, which we introduce in the sequel (see also [17, 18, 4]). This mapping will allow us to define so-called fixed-point semantics.

Definition 14.
Let T be a terminology consisting of the concept definitions $A_1 = D_1, \ldots, A_n = D_n$, and J be a primitive interpretation. The mapping $T_J : (2^{dom(J)})^n \longrightarrow (2^{dom(J)})^n$ is defined as follows: Let $\underline{A} \in (2^{dom(J)})^n$ be a tuple and I the interpretation corresponding to J and $\underline{A}$. Then $T_J(\underline{A}) := (D_1^I, \ldots, D_n^I)$. ◇

It is easy to see that an interpretation I consisting of J and $\underline{A}$ is a model of the terminology T iff $\underline{A}$ is a fixed-point of T_J. According to example 1 $(2^{dom(J)})^n$ is a complete lattice. Analogous to [4], Proposition 16 it holds that T_J is upward ω-continuous.[2] Consequently, T_J is monotonic, and thus, has at least one fixed-point.

[2]In [4] this has been proven for the language $\mathcal{FL}_0$. The proof for $\mathcal{ALN}$ is a straightforward extension.

In other words, every primitive interpretation of a terminology can be extended to a model of this terminology. With that, the following definition is well-defined.

Definition 15 (semantics of (cyclic) terminologies).
Let T be a (cyclic) terminology.

i.) The *descriptive semantics* allows all models of T as admissible models.

ii.) The *least fixed-point semantics* (*lfp-semantics*) allows only those models of T as admissible models which come from the least fixed-point of a mapping T_J (*lfp-models*).

iii.) The *greatest fixed-point semantics* (*gfp-semantics*) allows only those models of T as admissible models which come from the greatest fixed-point of a mapping T_J (*gfp-models*).

$\diamond$

As already mentioned, there is always an unique extension of a primitive interpretation J to a model of an acyclic terminology T. Therefore, T_J has exactly one fixed-point, and thus, the three semantics coincide for acyclic terminologies. For cyclic terminologies every primitive interpretation can uniquely be extended to a lfp- resp. gfp-model.

Since T_J is upward ω-continuous, we have $gfp(T_J) = \bigcap_{i\geq 0} T_J^i(top) = T_J{\downarrow}^\omega$ for $top = (dom(J))^n$ as a consequence of Proposition 3. Proposition 4 yields $\underline{A}$-$gfp(T_J) = \bigcap_{i\geq 0} T_J^i(\underline{A})$ for $\underline{A} \subseteq (dom(J))^n$ and $\underline{A} \supseteq T_J(\underline{A})$. For the least fixed-point $lfp(T_J)$ of T_J we may have $lfp(T_J) \neq \bigcup_{i\geq 0} T_J^i(bottom) = T_J{\uparrow}^\omega$. Consider for example the terminology T with concept definitions $\mathsf{A} = \mathsf{Q}\sqcap\forall\mathsf{S}.\mathsf{B}$, $\mathsf{B} = \mathsf{P}\sqcap\forall\mathsf{R}.\mathsf{B}$ [4]. Furthermore, let J be a primitive interpretation: $dom(J) := \{a_0, a_1, a_2, \ldots\}$, $\mathsf{P}^J := \{a_1, a_2, a_3, \ldots\}$, $\mathsf{Q}^J := \{a_0\}$, $\mathsf{R}^J := \{(a_{i+1}, a_i); i \geq 1\}$, and $\mathsf{S}^J := \{(a_0, a_i); i \geq 1\}$. Then it holds $lfp(T_J) = T_J{\uparrow}^{\omega+1} \neq T_J{\uparrow}^\omega$ because the equality $(T_J{\uparrow}^\omega)_2 = \mathsf{P}^I$ ($index(\mathsf{B}) = 2$) is valid for ω but not for ordinals less than ω. Thus, $(T_J{\uparrow}^{\omega+1})_1 = \mathsf{Q}^J$ ($index(\mathsf{A}) = 1$) for $\omega + 1$ but not earlier.

Now the question arises which of the three semantics is to prefer. As argued in [18, 20], none of the semantics fits best in any given representation task. For example, we will show that w.r.t. the lfp-semantics the concept Human is inconsistent. One would expect that the concept Binary-tree in example 10 is subsumed by Ternary-tree. However, as we will show using the characterization of subsumption w.r.t. the descriptive semantics, this intuition is not captured by the descriptive semantics. Finally, if the atomic concept Donkey is defined in the same way as Human in example 9, then Donkey and Human are equivalent w.r.t. the gfp-semantics which contradicts the intuition.

Nevertheless, in many cases the gfp- and descriptive semantics are the semantics of choice. Additionally, the gfp-semantics allows for a more natural automata theoretic characterization than the other two semantics.

In this paper, we not only characterize the three semantics using finite automata, but also give an automata theoretic characterization of the important inference problems inconsistency and subsumption w.r.t. all three semantics. The formal definitions of these problems are as follows:

Definition 16.
Let T be a terminology, A, B atomic concepts in T.

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

In order to refer to the terminology we will say that A is T-*subsumed* by B w.r.t. descriptive (lfp-, gfp-)semantics. The concept A is *inconsistent* (T-*inconsistent*) w.r.t. the descriptive (lfp-, gfp-)semantics iff for all (lfp-, gfp-)models I of T it holds: $A^I = \emptyset$. $\diamond$

Chapter 4

Semi-automata and gfp-semantics

In this chapter we characterize the gfp-semantics for $\mathcal{ALN}$-terminologies. We will see that this semantics allows for a very natural description using finite automata. With the help of this characterization we prove automata theoretic characterizations of inconsistency and subsumption which leads to decision algorithms and complexity results for these problems. With that, the already existing results for $\mathcal{FL}_0$ [4] are extended by number-restrictions and primitive negation.

Unlike $\mathcal{FL}_0$, in $\mathcal{ALN}$ inconsistent concepts are expressible. It has turned out that this fact prohibits a straightforward extension of the results for subsumption w.r.t. $\mathcal{FL}_0$. Therefore, the notion "*exclusion of concepts*" will be introduced. Furthermore, words which exclude concepts will be described by so-called "*exclusion sets*" in order to formulate decision algorithms not only for subsumption but also for inconsistency. Both "exclusion" and "exclusion sets" will be important even for the other two semantics, which we deal with in the following chapters. However, for these semantics the notions have to be modified appropriately.

Since $\mathcal{ALN}$ allows for number-restrictions, primitive negation can be dispensed with using the technique proposed in [3]: The concepts $\neg P$ and P in a terminology can respectively be replaced by $(\leq 0\ R_P)$ and $(\geq 1\ R_P)$ for a new role name R_P. Without loss of generality, we will thus restrict our attention to $\mathcal{FLN}$ in the sequel. In addition, to simplify the formal definition of the notion "exclusion" we may (without loss of generality) assume that an $\mathcal{FLN}$-terminology contains no minimum-restrictions of the form $(\leq 0\ R)$ since such a term can be substituted by $\forall R.\bot$.[1] For the characterization of subsumption we therefore consider only terminologies without $(\leq 0\ R)$, which we call $\mathcal{FLN}^r$-terminologies. Since these reductions of $\mathcal{ALN}$ to $\mathcal{FLN}$ and $\mathcal{FLN}^r$ are polynomial, the here presented complexity results for these languages are also valid for $\mathcal{ALN}$.

[1]The symbol $\bot$ denotes the empty concept, which can be expressed by $(\geq 2\ R) \sqcap (\leq 1\ R)$.

4.1 The semi-automaton $\mathcal{A}_T$

In order to state automata theoretic characterizations we associate a semi-automaton $\mathcal{A}_T$ to a $\mathcal{FLN}$-terminology T. First, we need a further definition and additionally have to normalize T:

For a word $W = R_1 \cdots R_n$ and an interpretation I the composition $R_1^I \circ \cdots \circ R_n^I$ of the relations R_i^I, $1 \leq i \leq n$, is denoted by W^I, where for the empty word ε^I denotes the identity relation.

For $\mathcal{FLN}$-concepts C, D and a role R the concepts $\forall R.(C \sqcap D)$ and $\forall R.C \sqcap \forall R.D$ are obviously equivalent. Thus, every $\mathcal{FLN}$-concept is equivalent to a conjunction of concepts of the form $\forall R_1.\forall R_2.\ldots.\forall R_n.C$ where C denotes an atomic concept or a number-restriction. According to the above definition $\forall R_1.\forall R_2.\ldots.\forall R_n.C$ is equivalent to $\forall W.C$ where $W = R_1 \cdots R_n$; note that $\forall \varepsilon.C$ is equivalent to C. We call a $\mathcal{FLN}$-terminology T *normalized* if the right-hand side of the concept definitions in T are conjunctions of concepts of the form $\forall W.C$ where C and W are defined as above.

Definition 17 (The semi-automaton $\mathcal{A}_T$).
Let T (w.o.l.g.) be a normalized $\mathcal{FLN}$-terminology. The corresponding semi-automaton (with word-transitions) $\mathcal{A}_T = (\Sigma, Q, E)$ is defined as follows: The alphabet Σ of $\mathcal{A}_T$ is the set of all role names occurring in T; the states Q of $\mathcal{A}_T$ are the atomic concepts and number-restrictions in T; every concept definition of the form $A = \forall W_1.A_1 \sqcap \cdots \sqcap \forall W_n.A_n$ in T gives rise to the transitions $(A, W_1, A_1), \ldots,$ $(A, W_n, A_n) \in E$. $\diamond$

Obviously, one can construct $\mathcal{A}_T$ in time polynomial in the size of T.

Remark 18.
By Lemma 6 one can construct— with linear complexity—a semi-automaton $\mathcal{A}_T{}'$ without word-transitions such that the introduced languages for $\mathcal{A}_T$ and $\mathcal{A}_T{}'$ (see Lemma 6) coincide. Thus, we can (w.l.o.g.) assume that $\mathcal{A}_T$ is a semi-automaton without word-transitions. This is needed if decision algorithms for subsumption are considered.

Let T' be a terminology corresponding to $\mathcal{A}_T{}'$. Then, T' can directly be constructed out of T as follows: Every sub-concept of the form $\forall RW.C$ in T ($R \in \Sigma$ and $W \in \Sigma^+$) is replaced by $\forall R.A$ where A is a new introduced concept with concept definition $A = \forall W.C$. This substitution can be iterated until there are only sub-concepts of the form $\forall W.C$ for $W = \varepsilon$ or $W = R$, $R \in \Sigma$. $\diamond$

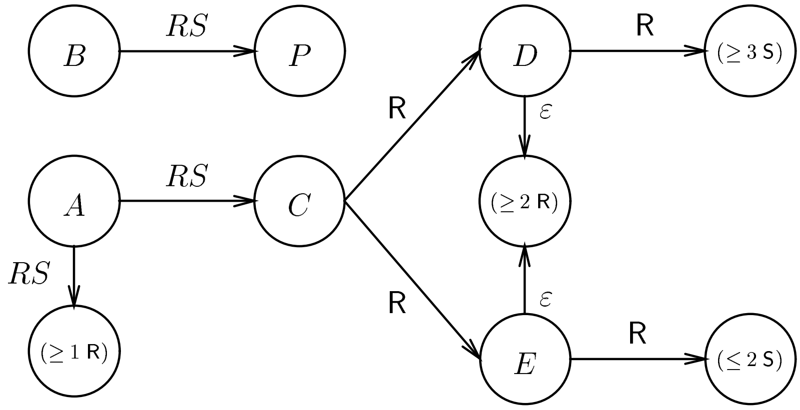
The following is an example of a normalized $\mathcal{FLN}$-terminology with corresponding semi-automaton $\mathcal{A}_T$.

Example 19.
Let T be defined by the following concept definitions:

$$\mathsf{A} = \forall \mathsf{RS.C} \sqcap \forall \mathsf{RS.}(\geq 1\ \mathsf{R})$$

$$\begin{aligned}
\mathsf{C} &= \forall \mathsf{R.D} \sqcap \forall \mathsf{R.E} \\
\mathsf{D} &= \forall \varepsilon.(\geq 2\ \mathsf{R}) \sqcap \forall \mathsf{R}.(\geq 3\ \mathsf{S}) \\
\mathsf{E} &= \forall \varepsilon.(\geq 2\ \mathsf{R}) \sqcap \forall \mathsf{R}.(\leq 3\ \mathsf{S}) \\
\mathsf{B} &= \forall \mathsf{RS.P}
\end{aligned}$$

The corresponding semi-automaton $\mathcal{A}_T$ is given by the following graph:



◇

In order to characterize inconsistency and subsumption, the notion "*requiring*" is very useful. Due to number-restrictions, atomic concepts can "require" chains of role successors for every instance of such a concept.

Definition 20 (require).
Let T be an $\mathcal{FLN}$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton and A an atomic concept in T. Let $W = R_1 \cdots R_n$ be a finite word and $V = R_1 \cdots R_m$ a prefix of W, i.e. $m \leq n$. The word W is *required by A starting from V* iff for all i, $m \leq i < n$, there are numbers $m_{i+1} \geq 1$ such that $VR_{m+1} \cdots R_i \in L(A, (\geq m_{i+1}\ R_{i+1}))$. Let $W \in \Sigma^\omega$, $W = R_1R_2R_3 \cdots$, be an infinite word and $V = R_1 \cdots R_m$ an finite prefix of W. Then W is *required by A starting from V* iff every finite prefix of W which also has V as prefix is required by A starting from V, i.e.: for all $i \geq m$ there are numbers $m_{i+1} \geq 1$ such that $VR_{m+1} \cdots R_i \in L(A, (\geq m_{i+1}\ R_{i+1}))$ for all $i \geq m$.
If $V = \varepsilon$ we say "W is required by A" instead of "W is required by A starting from ε". ◇

The important property of this notion is captured by

Lemma 21.
In addition to the above denotation let I be a model of T and d, e_m individuals in $dom(I)$ such that $dV^I e_m$ and $d \in A^I$. We distinguish two cases. (a) W is a finite word. Then there is a $f \in dom(I)$ such that $dV^I e_m (R_{m+1} \cdots R_n)^I f$. (b) W is an infinite word. Then there are individuals $e_{m+1}, e_{m+2}, e_{m+3}, \ldots \in dom(I)$ such that $dV^I e_m R^I_{m+1} e_{m+1} R^I_{m+2} e_{m+2} \cdots$.

Proof. We only consider the case $W \in \Sigma^\omega$; the case $W \in \Sigma^*$ can be shown analogously. The existence of the individuals e_i, $i \geq m$ such that $dV^I e_m (R_{m+1} \cdots R_i)^I e_i$ is proven by induction over i starting from m:

For $i = m$ there nothing is to show. For $i \geq m$ the word $VR_{m+1} \cdots R_i$ is an element of $L(A, (\geq m_{i+1}\ R_{i+1}))$. We have $dV^I e_m (R_{m+1} \cdots R_i)^I e_i$ as induction hypothesis. The characterizations of the semantics (see Theorems 22, 42 resp. 51)[2] yield $e_i \in (\geq m_{i+1}\ R_{i+1})^I$ because of $d \in A^I$. Since $m_{i+1} \geq 1$, there is an R_{i+1}-successor e_{i+1} of e_i. □

Because of $\mathsf{parents}^j \in L_{\mathcal{A}_T}(\mathsf{Human}, (\geq 2\ \mathsf{parents}))$ for all $j \geq 0$, every word $\mathsf{parents}^j$ is required by Human (example 9). As a consequence of Lemma 21, every instance of Human has an infinite chain of ancestors.

With that, we are prepared to characterize the gfp-semantics as well as inconsistency and subsumption.

4.2 Characterizing the gfp-semantics

The characterization of the gfp-semantics w.r.t. $\mathcal{FLN}$-terminologies is a straightforward extension of the characterization for $\mathcal{FL}_0$-terminologies [4].

Theorem 22 (Characterizing the gfp-semantics w.r.t. $\mathcal{FLN}$).
Let T be an $\mathcal{FLN}$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton, I a gfp-model of T, and A an atomic concept occurring in T. For every $d \in dom(I)$ we have $d \in A^I$ iff the following properties hold:

(P1) for all primitive concepts P in T and words $W \in L(A, P)$ it holds $d \in (\forall W.P)^I$;

(P2) for all maximum-restrictions $(\geq n\ R)$ in T and all words $W \in L(A, (\geq n\ R))$ it holds $d \in (\forall W.(\geq n\ R))^I$; and

(P3) for all minimum-restrictions $(\leq n\ R)$ in T and all words $W \in L(A, (\leq n\ R))$ it holds $d \in (\forall W.(\leq n\ R))^I$.

Proof. Analogous to the proof of [4], Proposition 19 □

Remark 23.
As a consequence of the reduction from $\mathcal{ALN}$ to $\mathcal{FLN}$ for the characterization w.r.t. $\mathcal{ALN}$-terminologies only the following property has to be added in Theorem 22: For all terms of the form $\neg P$ in T and words $W \in L(A, \neg P)$ it holds $d \in (\forall W.\neg P)^I$. ◇

In example 9 we have $L(\mathsf{Human}, (\geq 2\ \mathsf{parents})) = L(\mathsf{Human}, (\leq 2\ \mathsf{parents})) = \mathsf{parents}^*$. Thus, the ancestors of the individuals in Human^I must have exactly two parents. This reveals, that cyclic definitions allow to expressive the reflexive-transitive closure of relations.

[2]To proof Lemma 21 these Theorems are not necessary.

4.3 Inconsistency w.r.t. gfp-semantics

If the definition of the concept Human in example 9 is extended by the value-restriction $\forall$parents parents parents.(≥ 3 parents) (the grand-grandparents of a human being have at least three parents) then Human is inconsistent because of conflicting number-restrictions. Formally, number-restrictions $(\geq l\ R)$ and $(\leq r\ R)$ are *conflicting* if $l > r$.

To prove a characterization for inconsistency we define a canonical model $I = I(A, d_0)$ for an atomic concept A and an individual d_0 such that $d_0 \in A^I$ if A is consistent. The primitive interpretation of this model can be seen as a tree with root d_0. The edges of this tree are defined by the extensions of the roles. The idea is to satisfy property (P2) in Theorem 22. If there are no conflicting number-restrictions for A then it can be shown that property (P3) is also satisfied.

Definition 24 (canonical gfp-model w.r.t. $\mathcal{FLN}$).
Let T be an $\mathcal{FLN}$-terminology, $\mathcal{A}_T = (\Sigma, Q, E)$ the corresponding semi-automaton, and A an atomic concept in T. The primitive canonical interpretation $J = J(A, d_0)$ for A and the individual d_0 is defined as follows:
Let $W_0, W_1, W_2, W_3, \ldots$ be the infinite (w.o.l.g., the finite case can be dealt with analogously) enumeration of the words required by A such that $W_i \neq W_j$ for all $i \neq j$, and for all $i, j \geq 0$, $i < j$, it holds: $|W_i| \leq |W_j|$. Note that if one word is required by A then all prefixes of this word are also required. Consequently, $W_0 = \varepsilon$ and $|W_i| < |W_{i+1}|$ implies $|W_{i+1}| = |W_i| + 1$. We define J inductively:

$\boldsymbol{J}_0$: $dom(J_0) := \{d_0\}$; $R^{J_0} := \emptyset$ for all roles R in T; the extensions of the primitive concepts are only defined for J. Obviously, J_0 only consists of a finite set of individuals (basis step) .

$\boldsymbol{J}_{i+1}$: For W_{i+1} there is exactly one $j < i+1$ and one role R in T such that $W_{i+1} = W_j R$. Let $m \geq 1$ maximal such that $W_j \in L(A, (\geq m\ R))$, i.e., there is no $m' > m$ with $W_j \in L(A, (\geq m'\ R))$. (Such a positive integer m exists, because W_{i+1} is required by A and T contains only a finite number of number-restrictions. According to the induction hypothesis $dom(J_i)$ is a finite set. Thus, the set $K_{i+1} := \{d_1, \ldots, d_r\} := \{d \in dom(J_i); d_0 W_j^{J_i} d\}$ ($r \geq 0$ size of K_{i+1}) of W_j-successors of d_0 in J_i is finite. Let $e_1^1, \ldots, e_m^1, \ldots, e_1^r, \ldots, e_m^r$ be $(r{\cdot}m)$ new individuals. We extend J_i to J_{i+1}:
$dom(J_{i+1}) := dom(J_i) \dot\cup \{e_1^1, \ldots, e_m^1, \ldots, e_1^r, \ldots, e_m^r\}$; $S^{J_{i+1}} := S^{J_i}$ for all roles $S \neq R$ and $R^{J_{i+1}} := R^{J_i} \dot\cup \{(d_k, e_l^k); 1 \leq k \leq r,\ 1 \leq l \leq m\}$.
Obviously, $dom(J_{i+1})$ is finite (induction step). With that, we define $\boldsymbol{J}$:
$dom(J) := \bigcup_{i \geq 0} dom(J_i)$; $S^J := \bigcup_{i \geq 0} S^{J_i}$ for all roles S in T; for all primitive concepts P and individuals $d \in dom(J)$ let: $d \in P^J$ iff there exists a word $W \in \Sigma^*$ such that $W \in L(A, P)$ and $d_0 W^J d$.
The *canonical gfp-model* $I = I(A, d_0)$ for A and d_0 is defined as the gfp-model given by $J(A, d_0)$ and T. $\diamond$

In definition 24 every J_i can be seen as a finite tree with root d_0. The nodes are given by $dom(J_i)$ and the edges are defined by the extensions of the roles. The tree

J_0 consist only of the root d_0. The tree J_i is extended to J_{i+1} by adding m (see above) new R-successors to paths labeled with W_j (see above for the definition of j). The primitive interpretation J is the "limes" of these trees. The nodes in J belonging to the primitive concept P are labeled with P. For an infinite sequence $W_0, W_1, W_2, \ldots$ of words required by A the tree described by J has at least one infinite path (König's Lemma).

To conclude $d_0 \in A^I$ for the canonical model $I = I(A, d_0)$ the following condition is sufficient:[3]

> There is no word $W \in \Sigma^*$ and there are no conflicting number-restriction $(\geq l\ R)$ and $(\leq r\ R)$, $l > r$, such that W is required by A and $W \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$. (4.1)

The following lemma summarizes some properties of J and I:

Lemma 25.
Using the denotations and conditions of definition 24 it holds:

1.) The primitive canonical interpretation J is a tree with root d_0, i.e., for every individual (node) $e \in dom(J)$ there is an unique finite word W such that $d_0 W^J e$, and every node $e \in dom(J)$, $e \neq d_0$ has a unique predecessor, i.e., there is a unique role $S \in \Sigma$ and an unique node $d \in dom(J)$ such that $dS^J e$; there is no predecessor for d_0. The tree J is finitely branched. Every word W with $d_0 W^J e$ is required by A.

2.) For $V \in L(A, (\geq m\ R))$, m maximal with this property and $d_0 V^J d$, $d \in dom(J)$, it holds: $|R^J(d)| = m$.

3.) For every $W \in \Sigma^*$ required by A there is (at least) one individual $e \in dom(J)$ satisfying $d_0 W^J e$.

4.) Let V be a finite word and d be an individual. Then $d_0 V^J d$ implies: $d \in P^J$ iff $V \in L(A, P)$.

5.) Assuming that condition (4.1) is satisfied the properties (P1), (P2), and (P3) in Theorem 22 hold for A, d_0, T and J. Since I is a gfp-model of T it follows from Theorem 22, $d_0 \in A^I$.

Proof. 1.) This is can easily be proven using the construction of J and induction over i.

2.) Statement 1.) and $d_0 V^J d$ imply that V is required by A. For $m \geq 1$ (m maximal) and $V \in L(A, (\geq m\ R))$ the word VR is also required by A. Thus, there are non-negative integers j, i, $0 \leq j < i + 1$, such that $W_j = V$ and $W_{i+1} = VR$. According to the construction exactly m R-successors of d are generated in J_{i+1}. Since in no J_k, $k > i + 1$, R-successors of d are generated, we have $|R^J(d)| = m$.

[3] and following Theorem 29 also necessary

For $m = 0$ the word VR is not required by A, thus, 1.) implies that d has no R-successors, otherwise VR would be required by A.

3.) Is $W \in \Sigma^*$ required by A then there is (exactly) one $i \geq 0$ such that $W = W_i$. We show the existence of an individual $e \in dom(J)$, $d_0 W_i^J e$, by induction over i: For $i = 0$ we have $W = \varepsilon$. Thus, defining $e = d_0$ yields $d_0 W_i^J e$. The induction step is a consequence of statement 2.): For W_{i+1} there is a non-negative integer j, $j < i+1$, and a role R such that $W_{i+1} = W_j R$. Now let $V := W_j$. As a consequence of the induction hypothesis there is an individual d for V which satisfies $d_0 V^J d$. Since W_{i+1} is required by A there is a $m \geq 1$ such that $V \in L(A, (\geq m\ R))$. Statement 2.) implies the existence of an individual $e \in dom(J)$ satisfying $d_0 W_j^J d R^J e$.

4.) According to 1.) there is exactly one $V \in \Sigma^*$ satisfying $d_0 V^J d$. Using the definition of the extensions of the primitive concepts the statement follows immediately.

5.) Assume that condition (4.1) holds. Property (P1) w.r.t. A, d_0, T and J of Theorem 22 is a consequence of 4.).

Let $(\geq n\ R)$ be a maximum-restriction of T, $W \in L(A, (\geq n\ R))$ and $e \in dom(J)$, where $(d_0, e) \in W^J$. If $n \geq 1$, then there is a maximal non-negative integer $m \geq 1$ such hat $W \in L(A, (\geq m\ R))$. Since 2.) implies $|R^J(e)| = m$ and since $m \geq n$ it follows: $e \in (\geq n\ R)^J$. Hence, (P2) is valid. For $n = 0$ we have $e \in (\geq n\ R)^J$ since $(\geq 0\ R)^J = dom(J)$.

Let $(\leq n\ R)$ be a minimum-restriction in T, $W \in L(A, (\leq n\ R))$, and $e \in dom(J)$, where $(d_0, e) \in W^J$. In particular, 1.) implies that W is required by A. The assumption yields that there is no $m > n$ such that $W \in L(A, (\geq m\ R))$ (no conflicting number-restrictions). Now 2.) implies $|R^J(e)| \leq n$, and hence, $e \in (\leq n\ R)^J$. Thus, (P3) is valid. □

Remark 26.
For $\mathcal{ALN}$-terminologies also conflicts caused by P and $\neg P$ (analogous to conflicting number-restrictions) have to be taken into account. With that, one can define a similar canonical model $I = I(A, d_0)$ satisfying $d_0 \in A^I$ if A is consistent. ◇

For an algorithmic characterization of inconsistency and subsumption we need the following

Definition 27 (exclusion set w.r.t. gfp-semantics in $\mathcal{FLN}$).
Let T be a terminology and $\mathcal{A}_T = (\Sigma, Q, E)$ the corresponding semi-automaton without word-transitions (see Remark 18). The set $F_0 \subseteq Q$ is called *exclusion set* w.r.t. $\mathcal{A}_T$ (and to the gfp-semantics in $\mathcal{FLN}$) if the following holds: There is a non-negative integer n, a word $R_1 \cdots R_n \in \Sigma^*$, conflicting number-restrictions $(\geq l\ R)$ and $(\leq r\ R)$, $l > r$, and for all $1 \leq i \leq n$ there are integers $m_i \geq 1$ such that $F_i := next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq n$, implies $(\geq m_i\ R_i) \in F_{i-1}$ for all $1 \leq i \leq n$ and $(\geq l\ R), (\leq r\ R) \in F_n$. ◇

For $\mathcal{ALN}$-terminologies the set $F_0 \subseteq Q$ is also called exclusion set if F_n contains P and $\neg P$ for a primitive concept P.

To decide inconsistency and subsumption of concepts we formulate a NPSPACE-decision algorithm for deciding the set of exclusion sets. In the following algorithm the statements output "yes" and "no", respectively, not only lead to the corresponding output but also terminates the computation.

Algorithm 28.

Input: Semi-automaton $\mathcal{A}_T = (\Sigma, Q, E)$ without word-transitions corresponding to the $\mathcal{FLN}$-terminology T; $F_0 \subseteq Q$.

Output: There exists a computation with output "yes" iff F_0 is an exclusion set.

$F := F_0$;
$z := 0$;
while $z < 2^{|Q|}$ do
    if "there are conflicting number-restrictions $(\geq l\ R), (\leq r\ R) \in F$, $l > r$"
        then output "yes";
    if "there exists an integer $m \geq 1$ and a maximum-restriction $(\geq m\ R) \in F$"
        then "Guess (non-det.) $(\geq m\ R) \in F$ where $m \geq 1$"
        else output "no";
    $F := next_\varepsilon(F, R)$;
    $z := z + 1$
end;
output "no". △

Obviously, this algorithm terminates on every input. The correctness is easy to see considering the definition of exclusion sets. For the completeness of the above algorithm it is to show that if there is a word $R_1 \cdots R_n$ such that F_n (see definition 27) contains conflicting number-restrictions and $n \geq 2^{|Q|}$, then there is also a word satisfying $n < 2^{|Q|}$ such that for this word F_n contains conflicting number-restrictions. Using a "pumping-lemma" argument this is not hard to prove. Thus, the set of exclusion sets is decidable using polynomial space.

Theorem 29 (Characterizing inconsistency w.r.t. gfp-semantics).
Let T be an $\mathcal{FLN}$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton without word-transitions,[4] and A be an atomic concept in T. The following statements are equivalent:

1.) A is T-inconsistent w.r.t. the gfp-semantics in $\mathcal{FLN}$.

2.) There exists a word W required by A and conflicting number-restrictions $(\geq l\ R)$ and $(\leq r\ R)$, $l > r$, where $W \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$.

3.) ε-$closure(\{A\})$ is an exclusion set.

[4] The equivalence of 1.) and 2.) is valid even for an arbitrary semi-automaton $\mathcal{A}_T$.

Proof. Equivalence of 1.) and 2.):

"1. $\Leftarrow$ 2.": Assume that 2.) is valid and let I be a gfp-model of T where $A^I \neq \emptyset$, i.e., there is an individual $d \in A^I$. Lemma 21 implies the existence of an individual $e \in dom(I)$ such that $dW^I e$. As a consequence of (P2) and (P3) of Theorem 22 we have $e \in (\geq l\ R)^I$ and $e \in (\leq r\ R)^I$ which is a contradiction because of $l > r$.

"1. $\Rightarrow$ 2.": Assume that 2.) is not valid. Then Lemma 25, 5.) implies the existence of the canonical gfp-model $I = I(A, d_0)$ such that $d_0 \in A^I$. Hence, A is consistent.

Equivalence of 2.) and 3.):

"2. $\Rightarrow$ 3.": Let $W \in \Sigma^*$ be the word $R_1 \cdots R_n$ which is required by A and which satisfies $W \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$, $l > r$. Let $F_i := next_\varepsilon(A, R_1 \cdots R_i)$ for all $0 \leq i \leq n$. Since W is required by A there are maximum-restrictions $(\geq m_i\ R_i)$, $m_i \geq 1$, where $R_1 \cdots R_i \in L(A, (\geq m_{i+1}\ R_{i+1}))$ for all $0 \leq i < n$. Thus, Lemma 5 implies $(\geq m_{i+1}\ R_{i+1}) \in F_i$ for all $0 \leq i < n$. Additionally, since $W \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$ the number-restrictions $(\geq l\ R)$ and $(\leq r\ R)$ are contained in F_n. Hence, $F_0 = \varepsilon\text{-}closure(\{A\})$ is an exclusion set.

"2. $\Leftarrow$ 3.": Let $F_0 := \varepsilon\text{-}closure(\{A\})$ be an exclusion set. Consequently there is a word $W = R_1 \cdots R_n \in \Sigma^*$, and maximum-restrictions $(\geq m_i\ R_i)$, $m_i \geq 1$, for all $1 \leq i \leq n$ such that for $F_i := next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq n$, it holds: $(\geq m_i\ R_i) \in F_{i-1}$ for all $1 \leq i \leq n$. In addition, F_n contains conflicting number-restrictions $(\geq l\ R)$ and $(\leq r\ R)$, $l > r$. Thus, Lemma 5 implies: $R_1 \cdots R_i \in L(A, (\geq m_{i+1}\ R_{i+1}))$ for all $0 \leq i < n$, i.e., W is required by A, and $W \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$. □

Remark 30.
For $\mathcal{ALN}$-terminologies also conflicts of the form $W \in L(A, P) \cap L(A, \neg P)$ have to be considered in 2.). Furthermore, the definition of exclusion sets has to be modified according to the remark on page 21. ◇

Since $\varepsilon\text{-}closure(\{A\})$ is computable in time polynomial in the size of T it follows as a consequence of the above theorem that inconsistency is decidable using polynomial space. We also have

Theorem 31 (Inconsistency w.r.t. gfp-semantics).
Inconsistency w.r.t. gfp-semantics for $\mathcal{ALN}$- and $\mathcal{FLN}$-terminologies is PSPACE-complete and NP-complete for (weak-)acyclic $\mathcal{ALN}$- and $\mathcal{FLN}$-terminologies (see Definition 78).

Proof. As already shown, inconsistency for $\mathcal{ALN}$- and $\mathcal{FLN}$-terminologies is in PSPACE. D. Calvanese [8] has shown that consistency w.r.t. $\mathcal{AL}$-schemas (see chapter 7) and descriptive semantics is PSPACE-complete. Since a concept is consistent w.r.t. descriptive semantics iff it is consistent w.r.t. gfp-semantics, consistency involving $\mathcal{AL}$-schemas is PSPACE-complete even for gfp-semantics. Thus, Theorem 75 implies the PSPACE-completeness for general $\mathcal{ALN}$-($\mathcal{FLN}$-)terminologies. Note, that $\mathcal{SLN}$-schemas introduced in chapter 7 comprise $\mathcal{AL}$-schemas.

NP-completeness for (weak-)acyclic terminologies is shown in Theorem 79. □

On page 19 we have extend the definition of the defined concept Human by a condition for grand-grandparents. For this concept the word parents parents parents is required by Human. As a consequence of Theorem 29 parents parents parents $\in L(\mathsf{Human}, (\leq 2\ \mathsf{parents})) \cap L(\mathsf{Human}, (\geq 3\ \mathsf{parents}))$ implies the inconsistency of Human.

4.4 Subsumption w.r.t. gfp-semantics

For the language $\mathcal{FL}_0$ subsumption has been characterized in [4] via inclusion of regular languages. With respect to gfp-semantics this characterization is as follows:

$$A \sqsubseteq_{gfp,T} B \quad \text{iff} \quad L(B,P) \subseteq L(A,P) \text{ for all primitive concepts } P \text{ in } T. \tag{4.2}$$

The straightforward extension of this characterization to $\mathcal{FLN}$-terminologies would be to add inclusions $L(B, (\geq n\ R)) \subseteq L(A, (\geq n\ R))$ as well as $L(B, (\leq n\ R)) \subseteq L(A, (\leq n\ R))$ for the number-restrictions $(\geq n\ R)$, $(\leq n\ R)$ in T. Using Theorem 22 it is easy to see that this condition is sufficient for $A \sqsubseteq_{gfp,T} B$. However, the example 19 shows that the straightforward extension is not necessary for subsumption:

Although, $\mathsf{RS} \in L(\mathsf{B},\mathsf{P})$ and $\mathsf{RS} \notin L(\mathsf{A},\mathsf{P})$, hence $L(\mathsf{B},\mathsf{P}) \not\subseteq L(\mathsf{A},\mathsf{P})$, it holds $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$ in example 19. Proof: Let I be a gfp-model of T and d an individual where $d \in \mathsf{A}^I$. Since $\mathsf{RS} \in L(\mathsf{A}, (\geq 1\ \mathsf{R}))$ and $\mathsf{RSR} \in L(\mathsf{A}, (\geq 2\ \mathsf{R}))$ the word RSRR is required by A starting from RS. Is there a RS-successor of d then Lemma 21 implies the existence of an individual e such that $d(\mathsf{RSRR})^I e$. Because of $\mathsf{RSRR} \in L(\mathsf{A}, (\geq 3\ \mathsf{S})) \cap L(\mathsf{A}, (\leq 2\ \mathsf{S}))$ Theorem 22 yields the contradiction $e \in (\geq 3\ \mathsf{S})^I$ and $e \in (\leq 2\ \mathsf{S})^I$. Consequently, d has no RS-successors. Hence, Theorem 22 implies $d \in \mathsf{B}^I$, and thus, $A \sqsubseteq_{gfp,T} B$.

Formally, we describe the property of the word RS in the following definitions. According to this definition the word RS "excludes" the atomic concept A. Subsumption relations $B \sqsubseteq_{gfp,T} A$ may be valid, although excluding words violet inclusions like $L(B,P) \subseteq L(A,P)$ (in the example $\{\mathsf{RS} \in L(B,P) \setminus L(A,P))$. The notion "exclusion" will be important for the other two semantics as well, even though with adapted definitions.

Definition 32 (exclusion).
Let T be an $\mathcal{FLN}^r$-terminology, $\mathcal{A}_T = (\Sigma, Q, E)$ the corresponding semi-automaton, and A an atomic concept in T. The word $W \in \Sigma^* \cup \Sigma^\omega$ *excludes* A iff there exists a finite prefix $V \in \Sigma^*$ of W, a word $V' \in \Sigma^*$ as well as conflicting number-restrictions $(\geq l\ R)$, $(\leq r\ R)$, $l > r$, such that $VV' \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$ and VV' is required by A starting from V. The set E_A denotes the set of finite words excluding A.[5] ◇

[5]The corresponding terminology and semantics will be clear from the context. Infinite words $W \in \Sigma^\omega$ will be crucial for the descriptive semantics.

If A is inconsistent, then Theorem 29 implies that the empty word ε excludes A, thus, $E_A = \Sigma^*$ since for every word in E_A all words with prefix W are contained in E_A as well.

Additionally to the denotations and conditions in definition 32 let I be a gfp-model of T and d, e individuals such that $dV^I e$. By Lemma 21 there is an individual f where $dV^I eV'^I f$ since VV' is required by A starting from V. Theorem 22 and $d \in A^I$ imply the contradiction: $f \in (\geq l\ R)^I$ and $f \in (\leq r\ R)^I$. Thus, we have

Lemma 33.
Let $W \in \Sigma^* \cup \Sigma^\omega$ be an A-excluding word and V defined as in definition 32 such that $dV^I e$ for individuals d, e and a gfp-model I. Then it holds: $d \notin A^I$. □

Note, that we only address $\mathcal{FLN}^r$-terminologies in definition 32. The reason for this is the following: Let $V \in \Sigma^*$, $R \in \Sigma$ such that VR is a prefix of W and $V \in L(A, (\leq 0\ R))$. Let d, e be individuals where $dW^I e$. Hence, there is an f, $dV^I f$. For $d \in A^I$ Theorem 22 implies $f \in (\leq 0\ R)^I$, in contradiction to the fact that f has an R-successor. Thus, $d \notin A^I$. In terms of Lemma 33 the word W excludes A, i.e., $dW^I e$ implies that d cannot belong to the extension of A. Thus, minimum-restrictions of the form $(\leq 0\ R)$ state, additional to conflicting number-restrictions, further reasons for the exclusion of concepts. It turns out that this would prohibit to characterize the set E_A only using exclusion sets. Therefore, for the sake of simplicity we restrict our attention (w.l.o.g., see page 15) to $\mathcal{FLN}^r$-terminologies.

Before characterizing subsumption formally, we motivate the characterization looking at some examples: Similar to the inclusions in (4.2) we will formulate inclusions for number-restrictions. As we have already seen, the set E_A of A-excluding words must be taken into account. For primitive concepts P we have $L(B, P) \subseteq L(A, P) \cup E_A$ instead of $L(B, P) \subseteq L(A, P)$. For number-restrictions more sophisticated conditions are needed. Let T consists of the concept definitions $\mathsf{A} = \forall\mathsf{R}.(\geq 3\ \mathsf{R})$ and $\mathsf{B} = \forall\mathsf{R}.(\geq 2\ \mathsf{R})$. Although, $\mathsf{R} \in L(\mathsf{B}, (\geq 2\ \mathsf{R})) \setminus (L(\mathsf{A}, (\geq 2\ \mathsf{R})) \cup E_A)$ it holds $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$, since for an individual in A it is required that there are at least three R-successors for all R-successors of this individual. On the other hand for individuals in B only two R-successors are required for R-successors. In fact, it will turn out that not $L(B, (\geq l\ R)) \subseteq (L(A, (\geq l\ R)) \cup E_A)$, but $L(B, (\geq l\ R)) \subseteq (\bigcup_{r \geq l} L(A, (\geq r\ R)) \cup E_A)$ is necessary for $A \sqsubseteq_{gfp,T} B$. For minimum-restrictions a further extension is needed. For a word $W \in L(B, (\leq l\ R)) \setminus (\bigcup_{r \leq l} L(A, (\leq r\ R)) \cup E_A)$ it should be possible to define a gfp-model I such that $d \in A^I \setminus B^I$ for an individual d. For this purpose, there should be an individual $e \in dom(I)$ where $dW^I e$ and $e \notin (\leq l\ R)^I$ such that one can derive $d \notin B^I$ using Theorem 22. This requires that e has at least $(l+1)$ R-successors. Is, on the other hand, A excluded by WR the condition $d \in A^I$ implies that R-successors for e are not allowed. Hence, we should also assume that A is not excluded by WR. Example:

For the concept definitions $\mathsf{A} = \forall\mathsf{RR}.(\leq 1\ \mathsf{R}) \sqcap \forall\mathsf{RR}.(\geq 2\ \mathsf{R})$, $\mathsf{B} = \forall\mathsf{R}.(\leq 1\ \mathsf{R})$ it holds $\mathsf{W} = \mathsf{R} \in L(\mathsf{B}, (\leq 1\ \mathsf{R})) \setminus (\bigcup_{r \leq 1} L(\mathsf{A}, (\leq r\ \mathsf{R})) \cup E_A)$. Let I be a gfp-model and d an individual where $d \in \mathsf{A}^I$. The individual d has no (WR=)RR-successor, since

RR excludes A. Thus, for d there is no R-successor e such that e has at least two R-successors and therefore is not an element of $(\leq 1\ \mathsf{R})^I$. In fact, it holds $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$.

We will now define an extended canonical gfp-model I such that $d_0 \in A^I \setminus B^I$ if one of the necessary inclusions—briefly discussed above—for the subsumption relation $A \sqsubseteq_{gfp,T} B$ does not hold. For this purpose, the canonical model for A and d_0 will be extended as follows: If an inclusion is invalid because of the word W, i.e., this word is contained on the left-hand side but not on the right-hand side of the inclusion relation, then the tree of the canonical model with root d_0 is extended such that there is a path in the tree starting at the root labeled with W. If W violates the inclusion relation of a minimum-restriction $(\leq l\ R)$, then the path with label W additionally has to be extended by $(l+1)$ R-successors; more precisely by $(l+1)$ minus the already existing R-successors. Finally, this tree has to be completed such that the conditions formulated in (P1), (P2), and (P3) are satisfied w.r.t. A and d_0. Because of the extension of the canonical model, these conditions need not be satisfied anymore. Formally the extended model is defined as follows:

Definition 34 (extended canonical gfp-model w.r.t. $\mathcal{FLN}^r$).
Let T be an $\mathcal{FLN}^r$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton, A an atomic concept in T, W a word in Σ^*, $r \in \mathbb{N}$ and $R \in \Sigma$. For $r = 0$ we denote the *extended primitive canonical interpretation* by $J' = J(A, d_0, W)$; for $r > 0$ it is denoted by $J' = J(A, d_0, W, R, r)$. Let $U_1, U_2, U_3, \ldots$ be an enumeration of all words in Σ^* where $U_i \neq U_j$ for $i \neq j$ and $|U_i| \leq |U_j|$ for all $i < j$. Let $J = J(A, d_0)$ be the primitive canonical interpretation (see Definition 24) and $I = I(A, d_0)$ the corresponding gfp-model of T. We define J' inductively as follows:

$\boldsymbol{J}_0$: If W is required by A, then Lemma 25, 3.) implies the existence of $d_1 \in dom(J)$ such that $d_0 W^I d_1$. Let $k := r - |R^J(d_1)|$ (if greater or equal0 and $k := 0$ otherwise) and $f_1, \ldots, f_k$ be new, pairwise distinct individuals. For the sake of an uniformed denotation (also see the case in which W is not required by A) let $U := W$. We define:
$dom(J_0) := dom(J) \dot\cup \{f_1, \ldots, f_k\}$; $S^{J_0} := S^J$ for all roles $S \neq R$; $R^{J_0} := R^J \dot\cup \{(d_1, f_i);\ 1 \leq i \leq k\}$.

If W is not required by A, then there is a prefix U of W of maximal length which is required by A. Consequently, there is a $d_1 \in dom(J)$ such that $d_0 U^J d_1$. Furthermore, since W is not required by A, there is a $V \in \Sigma^+$, $V = R_1 \cdots R_n$, where $W = U R_1 \cdots R_n$. Let $k := r$, and $d_2, d_3, \ldots, d_{n+1}, f_1, \ldots, f_k$ be new, pairwise distinct individuals. We define:
$dom(J_0) := dom(J) \mathbin{\dot\cup} \{d_2, \ldots, d_{n+1}, f_1, \ldots, f_k\}$; $S^{J_0} := S^J \mathbin{\dot\cup} \{(d_i, d_{i+1});\ 1 \leq i \leq n,\ S = R_i\} \mathbin{\dot\cup} \{(d_{n+1}, f_i);\ 1 \leq i \leq k,\ S = R\}$ for all roles S in T.
The extensions of the primitive concepts are only defined for J'. By Lemma 25, 1.) and the construction of J_0 there are only finite many direct successors for every individual in $dom(J_0)$. Thus, $|V'^{J_0}(d_0)| < \infty$ for all $V' \in \Sigma^*$ (basis step).

$\boldsymbol{J}_{i+1}$: For the word U_{i+1} of the above enumeration the induction hypothesis implies: $|U_{i+1}^{J_i}(d_0)| < \infty$. With that also the following set is finite: $M_{i+1} \subseteq dom(J_i) \times \Sigma \times (\mathbb{N} \setminus \{0\})$ where $(g, S, m) \in M_{i+1}$ iff $d_0 U_{i+1}^{J_i} g$, $U_{i+1} \in L(A, (\geq m\ S))$, $m \geq 1$, m maximal with that property (i.e., there is no $m' > m$ such that $U_{i+1} \in L(A, (\geq m'\ S)))$, and $|S^{J_i}(g)| < m$. Let $(g_1, S_1, m_1), \ldots, (g_{k_{i+1}}, S_{k_{i+1}}, m_{k_{i+1}})$ denote the triples in M_{i+1} and $k_{i+1} \in \mathbb{N}$ the size of M_{i+1}. Thus, M_{i+1} contains a triple (g, S, m) if in the tree corresponding to J_i for g additionally $(m - |S^{J_i}(g)|)$ S-successors must be added in order to satisfy condition (P2) in Theorem 22 w.r.t. d_0 and A.

For this purpose, let $l_j := |S_j^{J_i}(g_j)|$ (note $l_j < m_j$) for all $1 \leq j \leq k_{i+1}$ and $D_{i+1} := \{g_1^1, \ldots, g_{m_1-l_1}^1, \ldots, g_1^{k_{i+1}}, \ldots, g_{m_{k_{i+1}}-l_{k_{i+1}}}^{k_{i+1}}\}$ a set of new, pairwise distinct individuals. We define:
$dom(J_{i+1}) := dom(J_i) \dot{\cup} D_{i+1}$; $S^{J_{i+1}} := S^{J_i} \dot{\cup} \bigcup_{1 \leq j \leq k_{i+1}, S=S_j} \{(g_j, g_1^j), \ldots, (g_j, g_{m_j-l_j}^j)\}$ for all roles S in T.
Since there are only a finite number of direct successors generated for $g_1, \ldots, g_{k_{i+1}}$, it holds for J_{i+1}: $|V'^{J_{i+1}}(d_0)| < \infty$ for all $V' \in \Sigma^*$ (induction step).
With that the extended primitive canonical interpretation $\boldsymbol{J'}$ is defined as follows: $dom(J') := \bigcup_{i \in \mathbb{N}} dom(J_i)$; $S^{J'} := \bigcup_{i \in \mathbb{N}} S^{J_i}$ for all roles S in T; for all primitive concepts P and individuals d let $d \in P^{J'}$ iff there is a $V' \in L(A, P)$ such that $d_0 V'^{J'} d$.
The *extended canonical gfp-models* $I' = I(A, d_0, W)$ and $I' = I(A, d_0, W, R, r)$, respectively, are the gfp-models of T with the corresponding primitive interpretations $J' = J(A, d_0, W)$ and $J' = J(A, d_0, W, R, r)$, respectively. ◇

The extended primitive canonical interpretation J' is sketched in figure 4.1 (also see Lemma 35, 4.) and 5.)). This interpretation can be seen as a tree with root d_0 which is iterative constructed out of J_0. The tree for J_0 is extended iteratively at final nodes of paths labeled with U_{i+1}, $i \geq 0$, which are not necessarily leaves of the tree J_i).

In order to prove $d_0 \in A^{I'}$ for the extended canonical model $I' = I(A, d_0, W)$ $(r = 0)$ and $I' = I(A, d_0, W, R, r)$ $(r > 0)$, the following condition is sufficient (and as it is easy to show using Theorem 29 and Lemma 33 even necessary):

The concept A is consistent and A is not excluded by W; additionally, for $r > 0$ the concept A is not excluded by WR. For all $l < r$ it holds $W \notin L(A, (\leq l\ R))$. (4.3)

We summarize the properties of J' and I' in

Lemma 35.
Using the denotations in Definition 34 it holds:

1.) J' is a tree with root d_0, i.e., for each $e \in dom(J')$ there is a unique (label) $V' \in \Sigma^*$ such that $d_0 V'^{J'} e$ and for each (node) $e \in dom(J')$, $e \neq d_0$, there is a unique predecessor, i.e., a unique role $S \in \Sigma$ and a unique individual $d \in dom(J')$ such that $d S^{J'} e$; the node d_0 has no predecessor.

2.) For each $d \in dom(J')$ successors are generated in J_{i+1} for at most *one* $i \geq 0$. More precisely, only finitely many direct successors are generated. Thus, the individual d has only finitely many direct successors, i.e., J' is a finite branched tree.

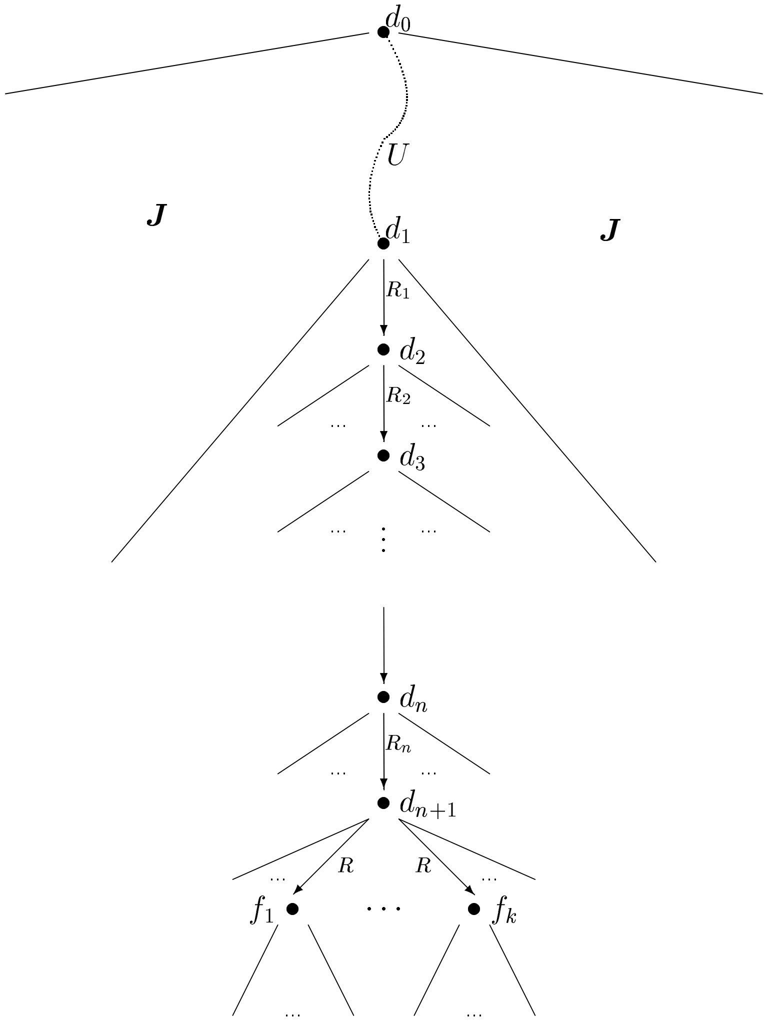


Figure 4.1: extended primitive canonical interpretation J'

3.) If $r = 0$ and W is required by A, then $S^{J'} = S^J$ for all roles S, $P^{J'} = P^J$ for all primitive concepts P, and $dom(J') = dom(J)$, i.e., J' and J coincide.

4.) All elements $d \in dom(J') \setminus dom(J)$ are (direct or indirect) successors of $d_1 \in dom(J)$. Apart from the direct successors in $dom(J)$ the individual d_1 merely has direct successors in $dom(J_0) \setminus dom(J)$, i.e., d_2 if W is not required by A and $f_1, \ldots, f_k$ if W is required by A.

5.) For all individuals $d, e \in dom(J)$ and $V' \in \Sigma^*$ it holds: $dV'^{J'}e$ iff $dV'^{J}e$; and if $d_0V'^{J'}d$, then V' is required by A.

6.) If $V' \in \Sigma^*$ is required by A, there is (at least) one $d \in dom(J)$ such that $d_0V'^{J}d$. Furthermore, 5.) implies $d_0V'^{J'}d$.

7.) For $V' \in \Sigma^*$ and $d \in dom(J')$ where $d_0V'^{J'}d$ it holds: $d \in P^{J'}$ iff $V' \in L(A, P)$.

8.) Let $V' \in L(A, (\geq m\ S))$, m maximal with this property, and $d \in dom(J') \setminus \{d_1, \ldots, d_{n+1}\}$ where $d_0 V'^{J'} d$. Then $|S^{J'}(d)| = m$. For $r = 0$ this holds even for $d = d_{n+1}$.

9.) There is an individual $d \in dom(J')$ such that $d_0 W^{J'} d$ and $|R^{J'}(d)| \geq r$.
If conditions (4.3) is satisfied, the properties (P1), (P2), and (P3) in Theorem 22 are valid w.r.t. A, d_0, T, and J'. Since I' is the gfp-model w.r.t. J' and T, Theorem 22 implies $d_0 \in A^{I'}$.

Proof. Statement 1.) can be shown easily by induction over the inductive definition of J': Considering Lemma 25, 1.) and the definition of J_0 it is not hard to see that J_0 is a tree with root d_0. According to the construction of J_{i+1} the tree J_i is extended at the final points (g_j) of paths labeled with U_{i+1} by a finite number of (S_j-)successors $(g_1^j, \ldots, g_{m_j-l_j}^j)$. It should be obvious that this preserves the tree property.

Statement 2.) is a consequence of 1.) since there is a unique $i \geq 0$ for every $d \in dom(J')$ such that $d_0 U_{i+1}^{J'} d$. Following the construction of J', only in J_{i+1} (a finite number of) successors for d are generated. Thus, in J' there are only a finite number of direct successors of d.

3.): Provided that $r = 0$ and W is required by A, no individuals are generated in J_0. Since for J condition (P2) of Theorem 22 is satisfied, for all $i \geq 0$ we have: $M_{i+1} = \emptyset$. Hence the construction leaves J unchanged. Note that the extensions of primitive concepts of J' and J are defined in the same way.

4.): (P2) of Theorem 22 is satisfied by A, d_0, and J. Consequently, for elements in $dom(J) \setminus \{d_1\}$ no successors are generated in J'. Therefore, all elements in $dom(J') \setminus dom(J)$ are (direct or indirect) successors of d_1. Additionally, for d_1 no direct successors are generated in J_{i+1}, $i \geq 0$: otherwise a role S and a non-negative integer m would exist such that $U_{i+1} \in L(A, (\geq m\ S))$, m maximal with this property, as well as $d_0 U_{i+1}^{J_i} d_1$ and $|S^{J_i}(d_1)| < m$. Furthermore, 1.) and $d_0 U^{J_i} d_1$ imply $U_{i+1} = U$. According to the definition of U it holds $d_0 U^J d_1$. Finally, $|S^{J_i}(d_1)| < m$ implies $|S^J(d_1)| < m$. Because of $U \in L(A, (\geq m\ S))$ this is a contradiction to the validity of (P2) w.r.t. A, d_0, and J. Thus, all direct successors of d_1 are contained in $dom(J_0)$.

Statement 5.) is a consequence of the fact that (the tree) J' is an extension of J and a consequence of Lemma 25, 1.). Statement 6.) follows from 5.) and Lemma 25, 3.). The validity of 7.) can easily be shown using the definition of the extensions of the primitive concepts and 1.).

Now assume that the conditions of statement 8.) hold. For $d \in dom(J)$ 5.) implies $d_0 V'^J d$. Lemma 25, 2.) yields $|S^J(d)| = m$. According to 4.) no successors are generated in J_{i+1}, $i \geq 0$, for elements d in $dom(J)$; there may be direct R-successors (resp., R_1-successors) in J_0 for $d = d_1$. This verifies statement 8.) for individuals $d \in dom(J) \setminus \{d_1\}$. Now let $d \in dom(J') \setminus (dom(J) \cup \{d_2, \ldots, d_{n+1}\})$. Consequently, d is an element of $\{f_1, \ldots, f_k\}$ or an element generated in J_{j+1} for one $j \geq 0$. Thus, there are no successors of d in J_0 and J_{j+1}, respectively. There is an $i \geq 0$ such that $V' = U_{i+1}$. For S-successors of d to be generated in J_{i+1} the condition $d \in J_i$ is necessary. This condition is true for $d \in J_0$. If d has been generated in

J_{j+1}, then there is an e in J_j such that for $V' = V''S'$, $S' \in \Sigma$, the statement $d_0 V''^{J_j} e$ is valid and d is a S'-successor of e. It holds $V'' = U_{j+1}$, hence $j+1 < i+1$, since the enumeration $U_1, U_2, \ldots$ of Σ^* is ascending ordered by the length of the words. Because of $j+1 \leq i$ it follows $d \in J_i$. Following the construction, in J_{i+1} exactly m S-successors of d are generated, thus $|S^{J_{i+1}}(d)| = m$. No other successors of d are generated (see 2.)) which implies $|S^{J'}(d)| = m$. Provided that $r = 0$, even for d_{n+1} in J_0 there are no successors. Thus, the above argument can be applied for $d = d_{n+1}$ as well. This proves 8.).

9.): According to the definition of J_0 there is an individual $d \in dom(J')$ such that $d_0 W^{J'} d$ and $|R^{J'}(d)| \geq r$.

We assume that condition (4.3) is satisfied. The properties (P1), (P2), and (P3) of Theorem 22 w.r.t. A, d_0, T, and J' have to be shown.

(P1): Let P be a primitive concept in T, V' a word with $V' \in L(A, P)$, and $g \in dom(J')$ such that $(d_0, g) \in V'^{J'}$. Statement 7.) implies $g \in P^{J'}$. Thus, (P1) holds.

(P2): Now let $(\geq l\ S)$ be a maximum-restriction in T, V' a word where $V' \in L(A, (\geq l\ S))$, and $g \in dom(J')$ with $(d_0, g) \in V'^{J'}$. For $l = 0$ it follows immediately $g \in (\geq l\ S)^{J'}$ because of $(\geq l\ S)^{J'} = dom(J')$. Let $l \geq 1$. There is an $i \geq 0$ such that $V' = U_{i+1}$. Let m ($m \geq l$) be maximal with $V' \in L(A, (\geq m\ S))$. Analogous to the proof of 8.) it can be shown $g \in dom(J_i)$. If $|S^{J_i}(g)| < m$, then it follows $(g, S, m) \in M_{i+1}$, and according to the definition of J_{i+1} we have $|S^{J_{i+1}}(g)| \geq m$. On the other hand, if $|S^{J_i}(g)| \geq m$, then we have $|S^{J_{i+1}}(g)| \geq m$ since J_{i+1} is an extension of J_i. Now in both cases the definition of J' yields $|S^{J'}(g)| \geq m \geq l$, hence $g \in (\geq l\ S)^{J'}$. This proves (P2).

(P3): Let $(\leq l\ S)$ be a minimum-restriction in T— $l > 0$ because of T in $\mathcal{FLN}^r$ —, V' a word where $V' \in L(A, (\leq l\ S))$ as well as $g \in dom(J')$ such that $(d_0, g) \in V'^{J'}$.

For $g \in dom(J) \setminus \{d_1\}$ statement 5.) implies $(d_0, g) \in V'^{J}$. Since A is consistent, Lemma 25, 5.) implies $g \in (\leq l\ S)^J$. Because of 4.), there is no other individual than d_1 in $dom(J)$ with successors in $dom(J') \setminus dom(J)$. Thus it holds $g \in (\leq l\ S)^{J'}$, and in particular $g \in (\leq l\ S)^{J_0}$.

Now let $g \in (dom(J_0) \setminus dom(J)) \cup \{d_1\}$. First, let W be required by A, i.e., $g = d_1$ or $g \in \{f_1, \ldots, f_k\}$. For $g = d_1$ statement 1.) implies $V' = W$. For $S = R$ we have $W \notin L(A, (\leq l'\ R))$ for $l' < r$ according to the assumption, and thus $l \geq r$. Consequently, for $|R^{J_0}(d_1)| = r$ it follows $d_1 \in (\leq l\ S)^{J_0}$. For $|R^{J_0}(d_1)| > r$ the construction of J_0 implies $R^J(d_1) = R^{J_0}(d_1)$. Because of $V' = W$ it follows $d_0 V'^J d_1$. Furthermore, Lemma 25, 5.) implies the validity of (P3) w.r.t. A, d_0, and J. Thus, we have $d_1 \in (\leq l\ R)^J$, and because of $R^J(d_1) = R^{J_0}(d_1)$ and $S = R$ also $g \in (\leq l\ S)^{J_0}$. For $S \neq R$ it follows $S^J(d_1) = S^{J_0}(d_1)$ which also yields $d_1 \in (\leq l\ S)^{J_0}$ because of Lemma 25, 5.). Since $f_1, \ldots, f_k$ have no successors in J_0, we have $g \in (\leq l\ S)^{J_0}$ for $g = f_i$, $1 \leq i \leq k$.

Now let W not be required by A, i.e., it holds $g \in \{d_1, \ldots, d_{n+1}\}$ or $g \in \{f_1, \ldots, f_k\}$. Since d_j, $2 \leq j \leq n$, has exactly one successor in J_0, namely the

R_{j+1}-successor d_{j+1}, for $g = d_j$, $2 \leq j \leq n$, it follows $g \in (\leq l\ S)^{J_0}$ because $l > 0$. For $g = d_1$ and $S \neq R_1$ we have $S^J(d_1) = S^{J_0}(d_1)$, which by Lemma 25, 5.) also implies $g \in (\leq l\ S)^{J_0}$. For $g = d_1$ and $S = R_1$ the word UR_1 is not required by A, i.e., according to Lemma 25, 1.) g has no R_1-successor in J. Consequently, in J_0 there is exactly one R_1-successor of g, namely d_2 which again implies $g \in (\leq l\ S)^{J_0}$. For $g = d_{n+1}$ we have $V' = W$. By the construction of J_0 and because W is not required by A the individual g has exactly r R-successors in J_0 and no other successors: For $S = R$ we have $l \geq r$ since $W \notin L(A, (\leq l'\ R))$ for all $l' < r$, and thus $g \in (\leq l\ S))^{J_0}$. For $S \neq R$ it holds $|S^{J_0}(g)| = 0$, hence $g \in (\leq l\ S)^{J_0}$. There are no successors for $f_1, \ldots, f_k$ in J_0 which, again, for $g = f_i$, $1 \leq i \leq k$, implies $g \in (\leq l\ S)^{J_0}$.

This means that $g \in (\leq l\ S)^{J_0}$ holds for all $g \in dom(J_0)$ where $d_0 V'^{J'} g$ and $V' \in L(A, (\leq l\ S))$ (basis step).

Induction hypothesis: for all $g \in dom(J_i)$, $d_0 V'^{J'} g$, and $V' \in L(A, (\leq l\ S))$ it holds $g \in (\leq l\ S)^{J_i}$. We will prove in the induction step:

For all $g \in dom(J_{i+1})$, $d_0 V'^{J'} g$ and $V' \in L(A, (\leq l\ S))$ it holds $g \in (\leq l\ S)^{J_{i+1}}$. (4.4)

Before proving (4.4), we show that this implies (P3): Let $g \in dom(J')$, $d_0 V'^{J'} g$, and $V' \in L(A, (\leq l\ S))$. Then there is a $j \geq 0$ such that $g \in dom(J_j)$. For at most one $i \geq 0$ (see 2.)) successors are generated for g in J_{i+1}, $j < i + 1$ (proof of 8.)). By (4.4) it holds $g \in (\leq l\ S)^{J_{i+1}}$. Thus, 2.) implies $g \in (\leq l\ S)^{J'}$.

Proof of (4.4): Let $g \in dom(J_{i+1})$, $d_0 V'^{J'} g$, and $V' \in L(A, (\leq l\ S))$. For $g \in dom(J_{i+1}) \setminus dom(J_i)$ the individual g was newly generated in J_{i+1} and according to the construction of J_{i+1} has no successors. This yields $g \in (\leq l\ S)^{J_{i+1}}$.

If $g \in dom(J_i)$ and there is not m such that $(g, S, m) \in M_{i+1}$, then no S-successor of g are generated in J_{i+1}. The induction hypothesis yields $g \in (\leq l\ S)^{J_i}$, and thus $g \in (\leq l\ S)^{J_{i+1}}$.

If on the other hand $(g, S, m) \in M_{i+1}$, i.e., it holds $d_0 U_{i+1}^{J_i} g$ as well as $U_{i+1} \in L(A, (\geq m\ S))$, $m \geq 1$ maximal with this property, and $|S^{J_i}(g)| < m$, then exactly $m - |S^{J_i}(g)|$ new successors are generated for g. Thus, $|S^{J_{i+1}}(g)| = m$. If $g \notin (\leq l\ S)^{J_{i+1}}$, then it holds $|S^{J_{i+1}}(g)| > l$, hence $m > l$. Because of $d_0 V'^{J'} g$ and $d_0 U_{i+1}^{J'} g$ by 1.) it follows: $V' = U_{i+1}$. Thus, $U_{i+1} \in L(A, (\geq m\ S)) \cap L(A, (\leq l\ S))$ and $m > l$. If U_{i+1} is required by A, by 6.) the individual g is an element of $dom(J)$. Following 5.) it holds $d_0 U_{i+1}^{J} g$. Since (P2) and (P3) of Theorem 22 hold for A and d_0 by Lemma 25, 5.), we have $g \in (\geq m\ S)^J \cap (\leq l\ S)^J = \emptyset$. But this is a contradiction. Hence, U_{i+1} is not required by A, and thus $g \in dom(J') \setminus dom(J)$. All elements in $dom(J') \setminus dom(J)$ are (direct or indirect) successors of d_1 (4.)). Consequently, U_{i+1} is of the form UXY for $X \in \Sigma^+$ (4.)), $Y \in \Sigma^*$ such that UX is maximum prefix of WR ($r > 0$) or W (r=0), respectively. According to the construction of J' the word UXY is required by A starting from UX, because for a final point of a path labeled with U_j, $j \geq 0$, S'-successors are generated if and only if $U_j \in L(A, (\geq m'\ S'))$ for $m' \geq 1$. Now $UXY \in L(A, (\geq m\ S)) \cap L(A, (\leq l\ S))$ and $m > l$ implies that WR ($r > 0$) and W ($r = 0$), respectively, exclude A which is a contradiction to the assumption. Thus, $g \in (\leq l\ S)^{J_{i+1}}$. This completes the proof of (4.4). □

Note that in Lemma 35, 9.) (proof of (P3)) we have used the fact that T is an $\mathcal{FLN}^r$-terminology. Therefore, in the following characterization of subsumption we restrict our attention to $\mathcal{FLN}^r$-terminologies (w.o.l.g.).

Theorem 36 (Characterizing subsumption w.r.t. gfp-semantics).
Let T be an $\mathcal{FLN}^r$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton and A, B atomic concepts in T. Then it holds: $A \sqsubseteq_{gfp,T} B$ iff

1.) $L(B,P) \subseteq L(A,P) \cup E_A$ for all primitive concepts P in T;

2.) $L(B,(\geq l\ R)) \subseteq \bigcup_{r\geq l} L(A,(\geq r\ R)) \cup E_A$ for all maximum-restrictions of the form $(\geq l\ R)$ in T where $l > 0$; and

3.) $L(B,(\leq l\ R)) \cdot R \subseteq (\bigcup_{r\leq l} L(A,(\leq r\ R))) \cdot R \cup E_A$ for all minimum-restrictions of the form $(\leq l\ R)$ in T.

Proof. "$\Rightarrow$": We assume that one of the inclusions is invalid and show $A \not\sqsubseteq_{gfp,T} B$.

(1) Assumption: $L(B,P) \not\subseteq L(A,P) \cup E_A$ for a primitive concept P in T. Consequently, there is a word $W \in L(B,P) \setminus (L(A,P) \cup E_A)$. If A is inconsistent, then $E_A = \Sigma^*$ which is a contradiction to $W \notin E_A$. Thus, A is consistent. Because of $W \notin E_A$ the concept A is not excluded by W. Hence, according to Lemma 35, 9.) the extended canonical model $I' = I(A, d_0, W)$ for A, the individual d_0, and W exists such that $d_0 \in A^{I'}$. Furthermore, Lemma 35, 9.) implies the existence of $d \in dom(I')$ where $d_0 W^{I'} d$. Lemma 35, 7.) yields $d \notin P^{I'}$ because of $W \notin L(A,P)$. Using $W \in L(B,P)$, $d_0 W^{I'} d$, and $d \notin P^{I'}$ Theorem 22 implies $d_0 \notin B^{I'}$. Hence, $A \not\sqsubseteq_{gfp,T} B$.

(2) Assumption: $L(B,(\geq l\ R)) \not\subseteq \bigcup_{r\geq l} L(A,(\geq r\ R)) \cup E_A$ for a maximum-restriction $(\geq l\ R)$ in T where $l > 0$. Thus, there is a $W \in \Sigma^*$ where $W \in L(B,(\geq l\ R)) \setminus (\bigcup_{r\geq l} L(A,(\geq r\ R)) \cup E_A)$. Analogous to (1) A is consistent and is not excluded by W. With that, the extended canonical model $I' = I(A, d_0, W)$ for A, the individual d_0, and W exists an it holds $d_0 \in A^{I'}$ (Lemma 35, 9.)). Furthermore, there is a $d \in dom(I')$ such that $d_0 W^{J'} d$. Because of $W \notin \bigcup_{r\geq l} L(A,(\geq r\ R))$ Lemma 35, 8.) implies that the individual d has less than l R-successors ($l > 0$), and thus, $d \notin (\geq l\ R)^{I'}$. Now by Theorem 22 w.r.t. B and d_0 we know $d_0 \notin B^{I'}$. Again, this shows $A \not\sqsubseteq_{gfp,T} B$.

(3) Assumption: $L(B,(\leq l\ R)) \cdot R \not\subseteq ((\bigcup_{r\leq l} L(A,(\leq r\ R))) \cdot R \cup E_A)$ for a minimum-restriction $(\leq l\ R)$ in T.
Thus, there is a $W \in \Sigma^*$ where $WR \in L(B,(\leq l\ R)) \cdot R \setminus ((\bigcup_{r\leq l} L(A,(\leq r\ R))) \cdot R \cup E_A)$. Analogous to (1) the concept A is consistent and not excluded by WR. Using $W \notin \bigcup_{r\leq l} L(A,(\leq r\ R))$ it follows that the extended canonical model $I' = I(A, d_0, W, R, l+1)$ exists, where $d_0 \in A^{I'}$, $|R^{I'}(d)| \geq l+1$, and $d_0 W^{I'} d$ for a $d \in dom(I')$ (Lemma 35, 9.)). Because of $W \in L(B,(\leq l\ R))$ and (P3) of Theorem 22 w.r.t. B and d_0 it follows $d_0 \notin B^{I'}$, and thus $A \not\sqsubseteq_{gfp,T} B$.

"$\Leftarrow$": Let the right-hand side of the equivalence be valid. Assume $A \not\sqsubseteq_{gfp,T} B$. Thus, there is a gfp-model I for T and an individual $d_0 \in dom(I)$ such that $d_0 \in$

$A^I \setminus B^I$. Because of $d_0 \notin B^I$ at least one of the conditions (P1), (P2), and (P3) of Theorem 22 is violated w.r.t. B and d_0.

(4) If (P1) is invalid, then there is a primitive concept P, a word $W \in L(B, P)$, and an individual $e \in dom(I)$ such that $d_0 W^I e$ and $e \notin P^I$. Because of $L(B, P) \subseteq L(A, P) \cup E_A$ is holds $W \in L(A, P)$ or $W \in E_A$. In case $W \in L(A, P)$, (P1) for A and d_0 implies immediately $d_0 \notin P^I$ in contradiction to the assumption. In case $W \in E_A$, Lemma 33 and $d_0 W^I e$ imply $d_0 \notin A^I$, again, in contradiction to the assumption.

(5) If (P2) is invalid, then there is a maximum-restriction $(\geq l\ R)$ in T, a word $W \in L(B, (\geq l\ R))$, and a $e \in dom(I)$ such that $d_0 W^I e$ and $e \notin (\geq l\ R)^I$ which, in particular, implies $l > 0$. Because of $L(B, (\geq l\ R)) \subseteq (\bigcup_{r \geq l} L(A, (\geq r\ R)) \cup E_A)$ it follows $W \in \bigcup_{r \geq l} L(A, (\geq r\ R))$ or $W \in E_A$. Let $r \in \mathbb{N}$ where $r \geq l$ and $W \in L(A, (\geq r\ R))$. Because of $|R^I(e)| < l \leq r$ it holds $e \notin (\geq r\ R)^I$ which using $d_0 W^I e$ and (P2) w.r.t. A and d_0 implies $d_0 \notin A^I$ in contradiction to the assumption. In case of $W \in E_A$, analogously to (4), it follows $d_0 \notin A^I$, again.

(6) If (P3) is invalid, then there is a minimum-restriction $(\leq l\ R)$ in T, a $W \in L(B, (\leq l\ R))$, and an individual $e \in dom(I)$ such that $d_0 W^I e$ and $e \notin (\leq l\ R)^I$. Because of $L(B, (\leq l\ R)) \cdot R \subseteq ((\bigcup_{r \leq l} L(A, (\leq r\ R))) \cdot R \cup E_A)$ we have $W \in L(A, (\leq r\ R))$ for a $r \leq l$ or $WR \in E_A$. First, let $W \in L(A, (\leq r\ R))$. Since $|R^I(e)| > l \geq r$, we know $e \notin (\leq r\ R)^I$ which using $d_0 W^I e$ and (P3) w.r.t. A and d_0 implies $d_0 \notin A^I$ in contradiction to the assumption. Analogous to (4) $WR \in E_A$, $d_0 W^I e$, and $|R^I(e)| > l$ yield $d_0 \notin A^I$ which is again a contradiction to the assumption. □

Note that the unions in 2.) and 3.) of the above theorem are finite since a terminology contains only a finite number of number-restrictions.

Remark 37.
For an $\mathcal{ALN}^\tau$-terminology the following condition must be added in Theorem 36:

$$L(B, \neg P) \subseteq L(A, \neg P) \cup E_A \text{ for all terms of the form } \neg P \text{ in } T.$$

Additionally, the definition of E_A must be modified, i.e., beside conflicting number-restrictions also pairs P, $\neg P$ must be taken into account. ◇

In Example 19 we have $E_A = \mathsf{RS}\Sigma^*$, $L(\mathsf{B}, \mathsf{P}) = \{\mathsf{RS}\}$, and $L(\mathsf{B}, (\geq 1\ \mathsf{R})) = L(\mathsf{B}, (\geq 2\ \mathsf{R})) = L(\mathsf{B}, (\geq 3\ \mathsf{S})) = L(\mathsf{B}, (\leq 2\ \mathsf{S})) = \emptyset$. With that, it is easy to see that the conditions of Theorem 36 are satisfied w.r.t. A and B. Thus, $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$.

By Theorem 36 it follows the intuitively expected subsumption relation $\mathsf{Binary\text{-}tree} \sqsubseteq_{gfp,T} \mathsf{Ternary\text{-}tree}$ in Example 10. Since the terminology contains no maximum-restrictions, it holds $E_{\mathsf{Binary\text{-}tree}} = E_{\mathsf{Ternary\text{-}tree}} = \emptyset$. Furthermore, the languages $L(\mathsf{Binary\text{-}tree}, \mathsf{Tree})$, $L(\mathsf{Ternary\text{-}tree}, \mathsf{Tree})$ as well as $L(\mathsf{Binary\text{-}tree}, (\leq 2\ \mathsf{direct\text{-}successor}))$ and $L(\mathsf{Ternary\text{-}tree}, (\leq 3\ \mathsf{direct\text{-}successor}))$ are equal to the language $\mathsf{direct\text{-}successor}^*$. Thus, the inclusions required in Theorem 36 for the subsumption relation $\mathsf{Binary\text{-}tree} \sqsubseteq_{gfp,T} \mathsf{Ternary\text{-}tree}$ are satisfied. The subsumption relation $\mathsf{Ternary\text{-}tree} \sqsubseteq_{gfp,T} \mathsf{Binary\text{-}tree}$ does not hold as intuitively expected. If in

Example 9 a concept definition for Donkey is added, which is defined analogously to the definition of Human, then the languages concerned in Theorem 36 for Human and Donkey coincide. Thus, the concepts Human and Donkey are equivalent which intuitively is not expected.

Using Theorem 36 one can decide subsumption by verifying the conditions 1.), 2.), and 3.). For this purpose, we first characterize the set E_A with the help of exclusion sets. Therefore, we need

Definition 38 (reaching an exclusion set).
Let T be a terminology and $\mathcal{A}_T$ the corresponding semi-automaton without word-transitions (see Remark 18). An exclusion set is *reachable* by a word $W \in \Sigma^* \cup \Sigma^\omega$ starting from the atomic concept A if there is a finite prefix V of W such that $next_\varepsilon(A, V)$ is an exclusion set.[6] ◇

Lemma 39.
Let T be an $\mathcal{FLN}^r$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton without word-transitions, and A an atomic concept in T. Then it holds:

$E_A = \{W \in \Sigma^*;$ an exclusion set is reachable by W starting from $A\}$.

Proof. It is not hard to prove this lemma by using the definitions of the notions exclusion words, exclusions sets, and applying Lemma 5. □

As already mentioned on page 25, definition 32 has to be modified for $\mathcal{FLN}$-terminologies. A word W excludes an atomic concept A even if W contains a prefix VR, $V \in \Sigma^*$, $R \in \Sigma$ where $V \in L(A, (\leq 0\ R))$. This case cannot be handle by the reachability of exclusion sets. Let $(\leq 0\ R)$ be an element of the set $next_\varepsilon(A, V)$. Now it depends on the next letter S $(W = VSV')$ in W if W excludes A; A is excluded by W if $S = R$. For $S \neq R$ the fact $(\leq 0\ R) \in next_\varepsilon(A, V)$ does not imply exclusion. Such a condition can easily be verified. But for the sake of simplicity we only consider $\mathcal{FLN}^r$-terminologies.

We now formulate a non-deterministic algorithm (NPSPACE-algorithm) in order to decide $L(B, P) \subseteq L(A, P) \cup E_A$ for a primitive concept P. The idea of this algorithm is as follows: The algorithm guesses a word which refutes the inclusion. More precisely, the algorithm simulates the product automaton constructed of the power-set automaton of $\mathcal{A}_T$ to decide the emptiness problem of the language $L(B, P) \cap (\Sigma^* \setminus (L(A, P) \cup E_A))$. It holds:

$W \in L(B, P) \cap (\Sigma^* \setminus (L(A, P) \cup E_A))$ iff (*) $P \in next_\varepsilon(B, W)$, $P \notin next_\varepsilon(A, W)$, and by W no exclusion set is reachable starting from A (consequence of Lemma 5 and Lemma 39). Furthermore, using an "pumping-lemma" argument it is easy to see that if there is a word W with properties (*), then there is also such a word W' which additionally satisfies $|W'| \leq 2^{2\cdot|Q|} - 1$. (4.5)

[6] The case $W \in \Sigma^\omega$ will be relevant in the next chapter.

With that, correctness and completeness of the following algorithm is not hard to prove.

Algorithm 40.

Input: semi-automaton $\mathcal{A}_T = (\Sigma, Q, E)$ without word-transitions for the $\mathcal{FLN}^r$-terminology T; atomic concepts A, B, and primitive concept P in T. $(n = |Q|)$

Output: There is a computation with output "yes" iff $L(B,P) \not\subseteq L(A,P) \cup E_A$.

$T_1 := \varepsilon\text{-}closure(\{B\})$;
$T_2 := \varepsilon\text{-}closure(\{A\})$;
$z := 0$;
while $z < 2^{2 \cdot n} - 1$ and $T_2 \notin \{F \subseteq Q;\ F \text{ exclusion set}\}$ and $(P \notin T_1$ or $P \in T_2)$ do
  $z := z + 1$;
  Guess (non-deterministic) an $R \in \Sigma$;
  $T_1 := next_\varepsilon(T_1, R)$;
  $T_2 := next_\varepsilon(T_2, R)$
end;
If $T_2 \notin \{F \subseteq Q;\ F \text{ exclusion set}\}$ and $P \in T_1$ and $P \notin T_2$
  then output "yes"
  else output "no" △

Since $T_2 \notin \{F \subseteq Q;\ F \text{ exclusion set}\}$ is decidable using polynomial space (see algorithm 28), the above algorithm is an NPSPACE-algorithm. Algorithm 40 can also be used to decide $L(B, (\geq l\ R)) \subseteq \bigcup_{r \geq l} L(A, (\geq r\ R)) \cup E_A$ for a maximum-restriction $(\geq l\ R)$ in T, $l \geq 1$. However, because of the union of regular languages $L(A, (\geq l\ R))$, $r \geq l$, some modifications are necessary: Let $Z := \{(\geq r\ R);\ r \geq l$ and $(\geq r\ R)$ a maximum-restriction in $T\}$. The expression "$(P \notin T_1$ or $P \in T_2)$" in algorithm 40 (while-loop) is replaced by "$((\geq l\ R) \notin T_1$ or $T_2 \cap Z \neq \emptyset)$". In the last if-condition "$(P \in T_1$ and $P \notin T_2)$" is substituted by "$((\geq l\ R) \in T_1$ and $T_2 \cap Z = \emptyset)$". Analogous to algorithm 40, it is not hard to see that for the so modified algorithm it holds: There is a computation with output "yes" iff $L(B, (\geq l\ R)) \not\subseteq \bigcup_{r \geq l} L(A, (\geq r\ R)) \cup E_A$. Obviously, this algorithm requires only polynomial space.

Now we specify a (non-deterministic) algorithm which decides $L(B, (\leq l\ R)) \cdot R \subseteq (\bigcup_{r \leq l} L(A, (\leq r\ R))) \cdot R \cup E_A$ for a minimum-restriction $(\leq l\ R)$ in T. The algorithm guesses a word which refutes the inclusion if such a word exists. The inclusion says that for every $W \in L(B, (\leq l\ R))$ it holds: $W \in \bigcup_{r \leq l} L(A, (\leq r\ R))$ or $WR \in E_A$. Thus, the algorithm must guess a word W such that $W \in L(B, (\leq l\ R))$ as well as $W \notin \bigcup_{r \leq l} L(A, (\leq r\ R))$ and $WR \notin E_A$. If we had had the condition $W \notin E_A$ in place of $WR \notin E_A$, we could have applied the algorithm proposed for the maximum-restrictions. We would have only had to substitute $(\geq l\ R)$ by $(\leq l\ R)$ and define $Z := \{(\leq r\ R);\ r \leq l$ and $(\leq r\ R)$ a minimum-restrictions in $T\}$ (*). It holds $WR \notin E_A$ iff both $W \notin E_A$ and for $T_2' := next_\varepsilon(A, W)$ the set $next_\varepsilon(T_2', R)$ is not an exclusion set. Thus, the algorithm for maximum-restriction has to be modified as follows in order to apply it to minimum-restrictions: The algorithm has minimum-restrictions $(\leq l\ R)$ instead of $(\geq l\ R)$ as input; Z is defined according to (*); the last if-condition is replaced by "$T_2 \notin \{F \subseteq Q;\ F$ exclusion set$\}$ and $next_\varepsilon(T_2, R) \notin \{F \subseteq Q;\ F$ exclusion set$\}$ and $(\leq l\ R) \in T_1$ and $T_2 \cap Z = \emptyset$". Condition $T_2 \notin \{F \subseteq Q;\ F$ exclusion set$\}$ ensures $W \notin E_A$, and $next_\varepsilon(T_2, R) \notin \{F \subseteq Q;\ F$ exclusion set$\}$ guarantees $WR \notin E_A$.

As shown in [4] subsumption for $\mathcal{FL}_0$ w.r.t. gfp-semantics is PSPACE-complete. Thus, we have

Corollary 41.
Subsumption with respect to gfp-semantics in general $\mathcal{ALN}$- and $\mathcal{FLN}$-terminologies is PSPACE-complete. □

Chapter 5

Semi-automata and descriptive semantics

For the descriptive semantics structurally identical definitions (like those for Human and Donkey, see page 33) need not lead to semantically equivalent concepts, as we verify later in this chapter. The characterization of the descriptive semantics already reveals that for defined concepts the relation to (possibly) other defined concepts is crucial.

5.1 Characterizing the descriptive semantics

As for the gfp-semantics the characterization of the descriptive semantics in $\mathcal{FLN}$ is a straightforward extension of the characterization in $\mathcal{FL}_0$, which has been proved in [4].

Theorem 42 (Characterizing the descriptive semantics w.r.t. $\mathcal{FLN}$).
Let T be an $\mathcal{FLN}$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton, J a primitive interpretation, and $\underline{A}$ a tuple where $T_J(\underline{A}) \subseteq \underline{A}$. Let I denote an model of T given by J and $\underline{A}$-$gfp(T_J)$ (see Proposition 4). For all atomic concepts A and all individuals $d \in dom(I)$ it holds: $d \in A^I$ iff

(P1) for all primitive concepts P in T and all words $W \in L(A, P)$ it holds $d \in (\forall W.P)^I$;

(P2) for all maximum-restrictions $(\geq n\ R)$ in T and all words $W \in L(A, (\geq n\ R))$ it holds $d \in (\forall W.(\geq n\ R))^I$;

(P3) for all minimum-restrictions $(\leq n\ R)$ in T and all words $W \in L(A, (\leq n\ R))$ it holds $d \in (\forall W.(\leq n\ R))^I$; and

(P4) for all defined concepts B, all $W \in L(A, B)$, and all individuals $e \in dom(I)$ where $(d, e) \in W^I$ it holds $e \in (\underline{A})_j$ $(j = index(B))$.

Proof. Analogous to the proof of [4], Proposition 28. □

Remark 43.
Analogous to Remark 23 one can generalize the above theorem to the language $\mathcal{ALN}$ by taking primitive negation into account. $\diamond$

5.2 Inconsistency w.r.t. descriptive semantics

Since every atomic concept A is T-consistent w.r.t. descriptive semantics iff A is T-consistent w.r.t. gfp-semantics, Theorem 29 also holds for the descriptive semantics. In particular, the notions “canonical model” (definition 24) and “exclusion set” (definition 27) are defined for the descriptive semantics as for the gfp-semantics. Consequently, we have

Theorem 44 (Inconsistency w.r.t. descriptive semantics).
Inconsistency with respect to descriptive semantics for $\mathcal{ALN}$-($\mathcal{FLN}$-)terminologies is PSPACE-complete and NP-complete for (weak-)acyclic $\mathcal{ALN}$-($\mathcal{FLN}$-)terminologies. $\square$

5.3 Subsumption w.r.t. descriptive semantics

For atomic concepts A, B and an $\mathcal{FLN}$-terminology T the subsumption relation $A \sqsubseteq_T B$ implies $A \sqsubseteq_{gfp,T} B$. For this reason, we adopt the conditions formulated in Theorem 36 as well as the notions “exclusion” and E_A (definition 32) for the descriptive semantics.

Remark 45.
Lemma 33 also holds for the descriptive semantics which can easily be verified by Theorem 42. $\diamond$

In [4], Theorem 29 subsumption w.r.t. descriptive semantics for atomic concepts A, B, and an $\mathcal{FL}_0$-terminology T has been characterized as follows:

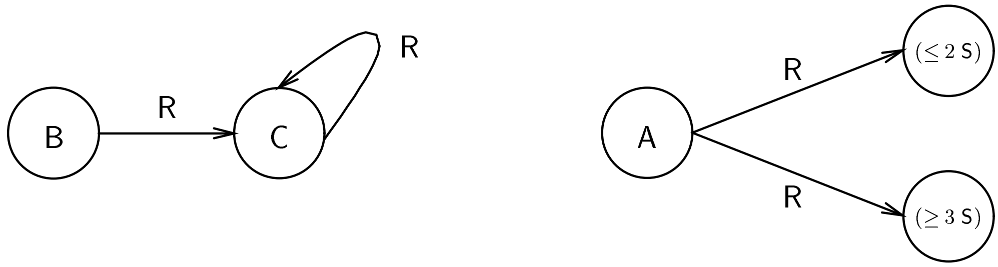
$A \sqsubseteq_T B$ iff

1.) $L(B, P) \subseteq L(A, P)$ for all primitive concepts P in T; and

2.) for all defined concepts C and infinite paths of the form $B, U_0, C, U_1, C, U_2, \ldots$ there is a $k \geq 0$ such that $U_0 \cdots U_k \in L(A, C)$. (5.1)

As already mentioned, in 1.) we have to take excluding words into account. Additionally, as for the gfp-semantics, conditions for number-restrictions are needed. Furthermore, in 2.) excluding words must be taken into account. We show this fact in the following example where the words R^n, $n \geq 1$, exclude A.

Example 46.
Let T be a terminology given by the corresponding semi-automaton $\mathcal{A}_T$:



The automaton $\mathcal{A}_T$ contains an infinite path $\mathsf{B}, \mathsf{R}, \mathsf{C}, \mathsf{R}, \mathsf{C}, \mathsf{R}, \mathsf{C}, \ldots,$, and it holds $\mathsf{R}^n \notin L(\mathsf{A}, \mathsf{C}) = \emptyset$ for all $n \geq 1$. As a consequence of 2.) the concept A should not be subsumed by B. Nevertheless, $\mathsf{A} \sqsubseteq_T \mathsf{B}$ holds since if I is a model of T where $d \in \mathsf{A}^I$, the individual d cannot have an R-successor. Such an R-successor would be an element of $(\leq 2\ S)^I$ and $(\geq 3\ S)^I$ according to Theorem 42 which is a contradiction. Thus, Theorem 42 implies $d \in \mathsf{B}^I$. ◇

In the proof of the characterization of $A \sqsubseteq_T B$ we use a model which refutes this subsumption relation. Similar to the gfp-semantics (definition 34) we define an extended primitive canonical interpretation. In definition 34 we have considered $J' = J(A, d_0, W)$ for an atomic concept A, an individual d_0, and a finite word W. Because of condition (P4) of Theorem 42 also ω-words W have to be taken into account for the descriptive semantics.

Definition 47 (extended primitive canonical interpretation).
Let T be an $\mathcal{FLN}^\tau$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton as well as A an atomic concept in T, and W a word in $\Sigma^* \cup \Sigma^\omega$. For $W \in \Sigma^*$ the *extended primitive canonical interpretation* $J' = J(A, d_0, W)$ for A, an individual d_0, and W is declared as in definition 34. For $W = R_1' R_2' R_3' \cdots \in \Sigma^\omega$ we define J' as follows:
If W is required by A, then let J' be the primitive canonical interpretation $J(A, d_0)$ (see definition 24). If W is not required by A, then there is a finite prefix U of W with maximum length ($W = UR_1R_2R_3\cdots$) which is required by A. According to Lemma 25, 3.) there is an individual d_1 such that $d_0 U^J d_1$. Let $d_2, d_3, \ldots$ be new individuals. We define J_0 by:
$dom(J_0) := dom(J) \dot\cup \{d_2, d_3, d_4, \ldots\}$; $S^{J_0} := S^J \dot\cup \{(d_i, d_{i+1});\ i \geq 1,\ S = R_i\}$ for all roles S in T. As in definition 34 it holds $|V'^{J_0}(d_0)| < \infty$ for all $V' \in \Sigma^*$.

Thus, J' is definable inductively by $J_1, J_2, \ldots$ as in definition 34.[1] ◇

In order to verify the properties (P1), (P2), (P3), and (P4) of Theorem 42 w.r.t. A, d_0, T, and J' the following condition it sufficient:[2]

> The atomic concept A is consistent and is not excluded by the (finite or infinite) word W. (5.2)

[1]The primitive interpretation $J' = J(A, d_0, W, R, r)$ for $R \in \Sigma$ and $r > 0$ is not needed (see proof of Theorem 48 ("⇒")).

[2]and necessary (see Theorem 29 and Remark 45)

For the extended primitive canonical interpretation J' w.r.t. a finite word W the properties stated in Lemma 35 also hold, because J' is defined as in definition 34. Note, that we have not extended J' to a model so far. We define a model in the proof of the characterization of subsumption. Therefore, we can not prove $d_0 \in A^{I'}$ yet. For a word $W \in \Sigma^\omega$ the statements of Lemma 35 can be verified as for the case $W \in \Sigma^*$ where the properties 8.) and 9.) are formulated as follows:

8.) Let $V' \in L(A, (\geq m\ S))$, m maximal with this property, and $d \in dom(J') \setminus \{d_1, d_2, d_3, \ldots\}$ where $d_0 V'^{J'} d$. Then it holds $|S^{J'}(d)| = m$.

9.) There are individuals $d_1, d_2, d_3, \ldots \in dom(J')$ such that $d_0 U^{J'} d_1 R_1^{J'} d_2 R_2^{J'} d_3 \cdots$ If condition (5.2) is satisfied, then the properties (P1), (P2), and (P3) of Theorem 42 (Theorem 22) w.r.t. A, d_0, T, and J' hold.

Now we can prove

Theorem 48 (Characterizing subsumption w.r.t. descriptive semantics). Let T be an $\mathcal{FLN}^r$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton, and A, B atomic concepts in T. It holds $A \sqsubseteq_T B$ iff

1.) $L(B, P) \subseteq L(A, P) \cup E_A$ for all primitive concepts P in T;

2.) $L(B, (\geq l\ R)) \subseteq \bigcup_{r \geq l} L(A, (\geq r\ R)) \cup E_A$ for all maximum-restrictions of the form $(\geq l\ R)$ in T where $l > 0$;

3.) $L(B, (\leq l\ R)) \cdot R \subseteq (\bigcup_{r \leq l} L(A, (\leq r\ R))) \cdot R \cup E_A$ for all minimum-restrictions of the form $(\leq l\ R)$ in T; and

4.) for all defined concepts C and all infinite paths of the form $B, U_0, C, U_1, C, U_2, C, \ldots$ there is a $k \geq 0$ such that $U_0 \cdots U_k \in L(A, C) \cup E_A$.

Proof. "$\Leftarrow$": Assume that the right-hand side of the equivalence is valid. Furthermore, let I be a model of T defined by the primitive interpretation J and the 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 $d \in dom(I)$ where $d \notin B^I$. It is to show $d \notin A^I$.

Because of $d \notin B^I$ Theorem 42 implies that at least one of the conditions (P1), (P2), (P3), and (P4) do not hold. The case in which (P1), (P2), or (P3) do not hold can be shown analogously to the proof of Theorem 36 ("$\Leftarrow$") using Theorem 42 (in place of Theorem 22).

Now consider the case that (P4) is invalid. Thus, there is a defined concept C_1, a word $W_1 \in L(B, C_1)$ as well as an individual $e_1 \in dom(I)$ where $(d, e_1) \in W_1^I$ and $e_1 \notin (\underline{A})_{i_1}$ ($i_1 = index(C_1)$). According to the assumption it holds $(\underline{A})_{i_1} = C_1^I$, and we can proceed with C_1 in place of B. We assume that we already have a sequence $C_1, W_1, e_1, \ldots, C_k, W_k, e_k$ for $e_0 := d$ and $C_0 := B$ 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 all $1 \leq i \leq k$.

Because of $e_k \notin C_k^I$ one of the conditions (P1), (P2), (P3), or (P3) of Theorem 42 is invalid. In case of (P1) there is a primitive concept P, a word $W \in$

$L(C_k, P)$, and an individual $e \in dom(I)$ where $(e_k, e) \in W^I$ and $e \notin P^I$. Thus, $W_1 \cdots W_k W \in L(B, P) \subseteq L(A, P) \cup E_A$, $e \notin P^I$, and $d(W_1 \cdots W_k W)^I e$. In case of $W_1 \cdots W_k W \in L(A, P)$ condition (P1) of Theorem 42 is violated, hence, $d \notin A^I$. In case of $W_1 \cdots W_k W \in E_A$ Remark 45 implies $d \notin A^I$. In case of (P2) and (P3) it follows $d \notin A^I$, analogously.

In case that (P4) is invalid we can assume for all $k \in \mathbb{N}$ that (P4) does not hold, otherwise we can apply the already considered cases (P1), (P2), and (P3). Thus, we have an infinite path $B, W_1, C_1, W_2, C_2, W_3, C_3, \ldots$ and individuals $e_1, e_2, e_3, \ldots$ with the properties stated above. Consequently, there is an atomic concept C such that $C = C_i$ for an infinite number of indices i. Hence, we have an infinite path of the form $B, U_0, C, U_1, C, U_2, \ldots$ By 4.) there is a $k \geq 0$ where $U_0 \cdots U_k \in L(A, C) \cup E_A$. Furthermore, there is an index i such that $d(U_0 \cdots U_k)^I e_i$ and $e_i \notin C^I = (\underline{A})_j$ ($j = index(C)$). In case of $U_0 \cdots U_k \in L(A, C)$ condition (P4) of Theorem 42 implies $d \notin A^I$; for $U_0 \cdots U_k \in E_A$ Lemma 45 yields $d \notin A^I$ as well.

"$\Rightarrow$": Assume that $A \sqsubseteq_T B$ holds. This implies $A \sqsubseteq_{gfp,T} B$. Since E_A is the same for the gfp-semantics and the descriptive semantics, the statements 1.), 2.), and 3.) of Theorem 36 and Theorem 48 coincide. Thus, it remains to show condition 4.). We assume that 4.) does not hold and construct a model which refutes $A \sqsubseteq_T B$. If 4.) does not hold, there is an infinite path of the form $B, U_0, C, U_1, C, U_2, \ldots$ such that $U_0 \cdots U_k \notin L(A, C) \cup E_A$ for all $k \geq 0$. Because of $U_0 \cdots U_k \notin E_A$ the concept A is consistent, otherwise $E_A = \Sigma^*$. Furthermore, A is not excluded by the (finite or infinite) word $W = U_0 U_1 U_2 \cdots$. This yields the existence of the extended primitive canonical interpretation $J = J_{min}(A, d_0, W)$ for A, the individual d_0, and the word W with the properties stated in Lemma 35 (and on page 40). Let $j_1 \leq j_2 \leq \cdots$ be indices such that $d_0 U_0^J d_{j_1} U_1^J d_{j_2} U_2^J \cdots$ holds. The tuple $\underline{A}$ is defined as follows: for a defined concept D in T with index m let $(\underline{A})_m := dom(J) \setminus \{e;$ there are words X, Y, and a $k \geq 0$ where $XY = U_0 \cdots U_k$, $X \in L(B, D)$, $Y \in L(D, C)$, $d_0 X^J e$, and $e Y^J d_{j_{k+1}}\}$.

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

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

According to the definition of $\underline{A}$ from $e \notin (\underline{A})_m$ it follows that there are finite words X, Y, and an index $k \geq 0$ such that $XY = U_0 \cdots U_k$, $X \in L(B, D)$, $Y \in L(D, C)$ as well as $d_0 X^J e$ and $e Y^J d_{j_{k+1}}$. Without loss of generality we can assume that the path from D to C is not empty (otherwise consider $k+1$ instead of k). Therefore, we can choose $Y = Y_1 Y_2$ such that there is an individual e' where $e Y_1^J e'$ and $e' Y_2^J d_{j_{k+1}}$, and such that the concept definition of D is of the form $D = \cdots \sqcap \forall Y_1.D' \sqcap \cdots$. Let m' be the index of D'. The definition of $\underline{A}$ yields $e' \notin (\underline{A})_{m'}$, and thus, $e \notin (T_J(\underline{A}))_m$. This completes the proof of the claim.

Now let I be the (well-defined, see claim) model of T defined by J and $\underline{A}$-$gfp(T_J)$. Let j be the index of B, i.e., $B^I = (\underline{A}\text{-}gfp(T_J))_j$. Because of $d_0 \varepsilon^I d_0$, $d_0 U_0^J d_{j_1}$, and $U_0 \in L(B, C)$ it follows $d_0 \notin (\underline{A})_j$, thus $d_0 \notin (\underline{A}\text{-}gfp(T_J))_j = B^I$.

Assumption: $d_0 \notin A^I$. Since A is consistent and not excluded by W, Lemma 35, 9.) implies that (P1), (P2), and (P3) for A, d_0, T, and J are satisfied. By Theorem 42, (P4) is not satisfied. But then, there is a defined concept D, a word $U \in L(A, D)$, and an individual $e \in dom(I)$ where $d_0 U^I e$ and $e \notin (\underline{A})_l$ ($l = index(D)$). Thus, according to the definition of $\underline{A}$, there are words X, Y, and an index $k \geq 0$ such that $XY = U_0 \cdots U_k$, $X \in L(B, D)$, $Y \in L(D, C)$ as well as $d_0 X^J e$ and $e Y^J d_{j_{k+1}}$. Because of $d_0 U^J e$ and $d_0 X^J e$ Lemma 35, 1.) yields that the word U and X are identical. Thus, $UY = XY = U_0 \cdots U_k \in L(A, C)$, which is a contradiction to the assumption that 4.) does not hold. □

Analogously to Remark 37, Theorem 48 can be generalized to $\mathcal{ALN}$-terminologies.

On page 33 we have pointed out that the concepts Human and Donkey are equivalent w.r.t. the gfp-semantics. On the other hand, Theorem 48 yields that these concepts are incomparable w.r.t. the descriptive semantics. Proof: The semi-automaton contains the infinite path Human, parents, Human, parents, Human, ... and Donkey, parents, Donkey, parents, Donkey, ..., respectively. Since $E_{\mathsf{Human}} = E_{\mathsf{Donkey}} = \emptyset$ and $L(\mathsf{Human}, \mathsf{Donkey}) = L(\mathsf{Donkey}, \mathsf{Human}) = \emptyset$, there is no $k \geq 0$ such that $\mathsf{parents}^k \in L(\mathsf{Human}, \mathsf{Donkey}) \cup E_{\mathsf{Human}}$ and $\mathsf{parents}^k \in L(\mathsf{Donkey}, \mathsf{Human}) \cup E_{\mathsf{Donkey}}$, respectively. Although, the descriptive semantics yields the intuitive subsumption relation in this example, this is not the case for example 10—unlike the gfp-semantics (see page 33): It holds $E_{\mathsf{Binary\text{-}tree}} = E_{\mathsf{Ternary\text{-}tree}} = \emptyset$ and $L(\mathsf{Binary\text{-}tree}, \mathsf{Ternary\text{-}tree}) = L(\mathsf{Ternary\text{-}tree}, \mathsf{Binary\text{-}tree}) = \emptyset$. For the infinite paths Binary-tree, direct-successor, Binary-tree, direct-successor,... and Ternary-tree, direct-successor, Ternary-tree, direct-successor,... there is no $k \geq 0$ such that $\mathsf{direct\text{-}successor}^k$ is an element of the language $L(\mathsf{Binary\text{-}tree}, \mathsf{Ternary\text{-}tree}) \cup E_{\mathsf{Binary\text{-}tree}}$ and $L(\mathsf{Ternary\text{-}tree}, \mathsf{Binary\text{-}tree}) \cup E_{\mathsf{Ternary\text{-}tree}}$, respectively. Thus, Theorem 48 implies that Binary-tree and Ternary-tree are incomparable which contradicts the intuition Binary-tree $\sqsubseteq_T$ Ternary-tree.

Since the conditions 1.), 2.), and 3.) of Theorem 48 are the same as in Theorem 36, this conditions can be decided using polynomial space. Now we formulate an (NPSPACE-)decision algorithm for condition 4.). To prove that there is a PSPACE-algorithm for this problem it is sufficient to show that there is a NPSPACE-algorithm for the following problem: Given an semi-automaton $\mathcal{A}_T$ without word-transitions as well as atomic concepts A, B, and C the question is,

$$\text{if there exists an infinite path of the form } B, U_0, C, U_1, C, U_2, C, \ldots \text{ such that for all } k \geq 0 \text{ it holds } U_0 \cdots U_k \notin L(A, C) \cup E_A. \tag{5.3}$$

The following NPSPACE-algorithm guesses such an infinite path, if there is such a path. To be more precise, if n is the size of the set of states of $\mathcal{A}_T$, then the algorithm guesses the words $U_0 \in \Sigma^*$, $|U_0| < 2^{2 \cdot n}$, and $U_1 \in \Sigma^+$, $|U_1| \leq 2^{2 \cdot n}$, such that $B, U_0, C, U_1, C, U_1, C, U_1, C \ldots$ is an infinite path in $\mathcal{A}_T$ where $U_0 U_1^k \notin L(A, C) \cup E_A$ for all $k \geq 0$. If C lies on an an ε-cycle, then in order to satisfy condition (5.3), it is sufficient to find a word $U_0 \in \Sigma^*$ such that $U_0 \in L(B, C)$ and $U_0 \notin L(A, C) \cup E_A$.

Algorithm 49.

Input: semi-automaton $\mathcal{A}_T = (\Sigma, Q, E)$ without word-transitions for a $\mathcal{FLN}^r$-terminology T; atomic concepts A, B, and C in T

Output: There is a computation with output "yes" iff (5.3) is satisfied.

Let n denote the size of the set of states, M the set of atomic concepts in T which lay on an ε-cycle as well as L the set of exclusion sets. For $S \subseteq Q$ the set $\overline{S}$ denotes the complement of S, i.e., $\overline{S} := Q \setminus S$.

```
    T₁:=ε-closure({B});
    T₂:=ε-closure({A});
    z:=0;
    stop:=false;
    (* Guess U₀ *)
(1) If C ∈ T₁ ∩ ¬T₂ then begin
        let stop be false or true (* (non-det.) U₀ = ε *)
    end;
    while z < 2^(2·n) − 1 and (not stop) and T₂ ∉ L do begin
        z:=z+1;
        Guess (non-det.) one R ∈ Σ;
        T₁ := next_ε(T₁, R);
        T₂ := next_ε(T₂, R);
(2)     If C ∈ T₁ ∩ ¬T₂ then begin
            let stop be true or false
        end
    end;
(3) If T₂ ∉ L and C ∈ T₁ ∩ ¬T₂ and C ∈ M then output "yes";
(4) If T₂ ∉ L and C ∈ T₁ ∩ ¬T₂ then begin
        (* Guess U₁ *)
        z := 0;
        T₁ := ε-closure({C});
        T₂' := T₂;
        stop := false;
        while z < 2^(2·n) and (not stop) and T₂ ∉ L do begin
            z := z + 1;
            Guess (non-det.) one R ∈ Σ;
            T₁ := next_ε(T₁, R);
            T₂ := next_ε(T₂, R);
(5)         If C ∈ T₁ and T₂ = T₂' (* in particular C ∉ T₂ *) then begin
                let stop be false or true
            end
        end; (*while*)
(6)     If T₂ ∉ L and C ∈ T₁ and T₂ = T₂' then output "yes"
    end; (*if*)
    output "no"                                                  △
```

Soundness: If the algorithm has "yes" as output, then by the definition of the algorithm there is a $U_0 \in \Sigma^*$, $U_0 = R_1 \cdots R_{m_0}$, $m_0 < 2^{2 \cdot n}$, such that for $T_{1,i} :=$

$next_\varepsilon(B, R_1 \cdots R_i)$ and $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$ holds: $T_{2,i} \notin L$ for all $0 \leq i \leq m_0$ and $C \in T_{1,m_0} \cap \overline{T_{2,m_0}}$. Furthermore, we distinguish the following cases:

i) $C \in M$ (if-condition in (3) is satisfied.)
Because of $C \in T_{1,m_0}$ Lemma 5 implies that the word U_0 is an element of $L(B, C)$. Thus, using $C \in M$ for $U_i = \varepsilon$, $i \geq 1$, the path $B, U_0, C, U_1, C, U_2, \ldots$ is an infinite path in $\mathcal{A}_T$. Furthermore, we know $U_0 \notin L(A, C)$ because of $C \notin T_{2,m_0}$ (Lemma 5). By Lemma 39 we have $U_0 \notin E_A$ since $T_{2,i} \notin L$ for all $0 \leq i \leq m_0$. Thus, for the infinite path $B, U_0, C, U_1, C, U_2, \ldots$ there is no $k \geq 0$ such that $U_0 \cdots U_k \in L(A, C) \cup E_A$.

ii) (if-condition in (6) is satisfied)
There is a non-empty word $U_1 \in \Sigma^+$, $U_1 = R'_1 \cdots R'_{m_1}$, $1 \leq m_1 \leq 2^{2 \cdot n}$ such that for $T'_{1,i} := next_\varepsilon(C, R'_1 \cdots R'_i)$ and $T'_{2,i} := next_\varepsilon(T_{2,m_0}, R'_1 \cdots R'_i)$ for all $0 \leq i \leq m_1$ it holds: $C \in T'_{1,m_1}$ and $T'_{2,m_1} = T_{2,m_0}$ (resp., $C \notin T'_{2,m_1}$) and $T'_{2,i} \notin L$ for all $0 \leq i \leq m_1$. Consequently, there is an infinite path $B, U_0, C, U_1, C, U_2, \ldots$ in $\mathcal{A}_T$ where $U_i = U_1$ for all $i \geq 1$, since $U_0 \in L(B, C)$ (because of $C \in T_{1,m_0}$) and $U_1 \in L(C, C)$ (because of $C \in T'_{1,m_1}$ and $T'_{1,0} = \varepsilon\text{-}closure(\{C\})$). Due to $T_{2,i} \notin L$ for all $0 \leq i \leq m_0$ no exclusion set is reachable by U_0 starting from A. Because of $T'_{2,i} \notin L$ for $0 \leq i \leq m_1$ no exclusion set is reachable by U_1 starting form $T'_{2,0} = T_{2,m_0}$. Thus, $U_0U_1 \notin E_A$. Since T'_{2,m_1} and T_{2,m_0} coincide, no exclusion set is reachable by $U_0U_1 \cdots U_k$ starting from A for any $k \geq 0$. But then $U_0U_1 \cdots U_k \notin E_A$ for all $k \geq 0$. Due to $C \notin T_{2,m_0}$ it follows $U_0 \notin L(A, C)$. Because of $C \notin T'_{2,m_1}$ and $T'_{2,0} = T_{2,m_0}$ we have $U_0U_1 \notin L(A, C)$. Hence, using $T_{2,m_0} = T'_{2,m_1}$ this yields $U_0 \cdots U_k \notin L(A, C)$ for all $k \geq 0$. Thus, for the infinite path $B, U_0, C, U_1, C, U_2, \ldots$ there is no $k \geq 0$ such that $U_0 \cdots U_k \in L(A, C) \cup E_A$. This shows the correctness of the algorithm.

Completeness: Let $B, U_0, C, U_1, C, U_2, \ldots$ be an infinite path such that there is no $k \geq 0$ with $U_0 \cdots U_k \in L(A, C) \cup E_A$. We prove the existence of a computation with output "yes". For this purpose, we distinguish the cases iii) and iv).

iii) If $U_0U_1U_2 \cdots$ is a finite word, it follows $C \in M$. Let $W = U_0U_1U_2 \cdots$ be the finite word $R_1 \cdots R_m$. We define $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ and $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$ for all $0 \leq i \leq m$. The path $B, W, C, \varepsilon, C, \varepsilon, \ldots$ is infinite where $W \notin L(A, C)$ and $W \notin E_A$ (choose k such that $W = U_0 \cdots U_k$).

Without loss of generality we can assume $m < 2^{2 \cdot n}$. Otherwise there exist l and r where $0 \leq l < r \leq 2^{2 \cdot n}$ as well as $T_{1,l} = T_{1,r}$ and $T_{2,l} = T_{2,r}$. For $W' = R_1 \cdots R_l R_{r+1} \cdots R_m$ starting from $T_{1,0}$ and $T_{2,0}$, respectively, $\mathcal{A}_T$ is caused to pass through the sets $T_{1,0}, T_{1,1}, \ldots, T_{1,l}, T_{1,r+1}, \ldots, T_{1,m}$ and $T_{2,0}, T_{2,1}, \ldots, T_{2,l}, T_{2,r+1}, \ldots, T_{2,m}$. Thus, even for the shorter word W' it holds $W' \in L(B, C)$ and $W' \notin L(A, C) \cup E_A$.

Now let $W = R_1 \cdots R_m$ be a word with $m < 2^{2 \cdot n}$ such that $B, W, C, \varepsilon, C, \varepsilon, \ldots$ is an infinite path in $\mathcal{A}_T$, and additionally, $W \notin L(A, C) \cup E_A$. With that, we show the existence of a computation with output "yes". Therefore, we distinguish the cases $W = \varepsilon$ and $W \neq \varepsilon$.

For $W = \varepsilon$ we can assign *true* to the variable *stop* in (1).Thus, the first while-loop is skipped. Because of $\varepsilon \notin L(A, C)$ and $\varepsilon \in L(B, C)$ it holds $C \in T_1 \cap \overline{T_2}$. The atomic concept C is an element of M according to the assumption. Because of

$\varepsilon \notin E_A$ we have $T_2 = \varepsilon\text{-}closure(\{A\}) \notin L$. Thus, the condition in (3) is satisfied, and the algorithm terminates with output "yes".

For $W \neq \varepsilon$ we assign $false$ to *stop* in (1) if the if-condition is satisfied. Due to $W \notin E_A$ it holds $T_2 \notin L$ before and after every iteration of the loop. In particular, after (1) the algorithm proceeds in the while-loop. In the i-th iteration of the loop one chooses for R the letter R_i. If the condition in (2) is satisfied, then, if $i < m$, *stop* is assigned to $false$. Because of $m < 2^{2 \cdot n}$ and $W \notin E_A$ the while-condition is satisfied even at the end of the $(m-1)$-th iteration. In the m-th iteration of the loop we assign R_m to R. Then, since $W \in L(B,C)$ and $W \notin L(A,C)$, the condition $C \in T_1 \cap \overline{T_2}$ in (2) is satisfied. Now we assign $true$ to *stop*. Thus, the while-loop terminates. Since the condition in (3) is satisfied ($W \notin E_A$), the algorithm terminates with output "yes".

iv) Now let $W = U_0U_1U_2\cdots$ be the ω-word $W = R_1R_2R_3\cdots \in \Sigma^\omega$. Without loss of generality we can assume $U_i \neq \varepsilon$ for all $i \geq 1$. Let $T_{1,i} := next_\varepsilon(B, R_1\cdots R_i)$ and $T_{2,i} := next_\varepsilon(A, R_1\cdots R_i)$ for all $i \geq 0$. Let $i_0, i_1, i_2, \ldots$ be indices such that $R_1\cdots R_{i_k} = U_0\cdots U_k$ for all $k \geq 0$. Because of $|2^Q| = 2^n$ there are numbers l and r such that $0 \leq l < r$ and $T_{2,i_l} = T_{2,i_r}$. Thus, for $U_0\cdots U_l$ and $U_{l+1}\cdots U_r$ it holds: $U_0\cdots U_l \in L(B,C)$, $U_0\cdots U_l \notin L(A,C) \cup E_A$, $U_{l+1}\cdots U_r \in L(C,C)$, $T_{2,i_l} = T_{2,i_r} = next_\varepsilon(T_{2,i_l}, U_{l+1}\cdots U_r)$ and $U_0\cdots U_l\cdots U_r \notin E_A$.

We now define words U'_0 and U'_1 using the words $U_0\cdots U_l$ and $U_{l+1}\cdots U_r$ such that $B, U'_0, C, U'_1, C, U'_1, \ldots$ is an infinite path in $\mathcal{A}_T$ where $U'_0{U'_1}^k \notin L(A,C) \cup E_A$ for all $k \geq 0$, $|U'_0| < 2^{2\cdot n}$, and $0 < |U'_1| \leq 2^{2\cdot n}$.

If $i_l \geq 2^{2\cdot n}$, then there are numbers s and t such that $0 \leq s < t \leq 2^{2\cdot n}$, $T_{1,s} = T_{1,t}$, and $T_{2,s} = T_{2,t}$. We define $U'_0 := R_1\cdots R_sR_{t+1}\cdots R_{i_l}$. Hence, for U'_0 starting from $T_{1,0}$ and $T_{2,0}$ the sets $T_{1,0}, T_{1,1}, \ldots, T_{1,s}, T_{1,t+1}, \ldots, T_{1,i_l}$ and $T_{2,0}, T_{2,1}, \ldots, T_{2,s}, T_{2,t+1}, \ldots, T_{2,i_l} \notin L$ are passed through. Consequently, as for $U_0\cdots U_l$, it holds $U'_0 \notin L(A,C) \cup E_A$ and $U'_0 \in L(B,C)$. Thus, if U'_0 is of minimal length we have $|U'_0| < 2^{2\cdot n}$, since for a word in $L(B,C) \setminus (L(A,C) \cup E_A)$, which is greater or equal $2^{2\cdot n}$, we can construct a shorter word with corresponding properties.

Because of $U_i \neq \varepsilon$ for all $i \geq 1$ and $l < r$ we know $U_{l+1}\cdots U_r \neq \varepsilon$. If $|U_{l+1}\cdots U_r| > 2^{2\cdot n}$, then for $T'_{1,j} := next_\varepsilon(C, R_{i_l+1}\cdots R_j)$ ($i_l \leq j \leq i_r$) and the word $U_{l+1}\cdots U_r$ the sequence $(T'_{1,i_l}, T_{2,i_l}), (T'_{1,i_l+1}, T_{2,i_l+1}), \ldots, (T'_{1,i_r}, T_{2,i_r})$ of pairs in $2^Q \times 2^Q$ is passed through. Since $|2^Q \times 2^Q| = 2^{2\cdot n}$, there are numbers s and t such that $i_l \leq s < t \leq i_l + 2^{2\cdot n}$, $T'_{1,s} = T'_{1,t}$, and $T_{2,s} = T_{2,t}$. Thus, for $U'_1 := R_{i_l+1}\cdots R_sR_{t+1}\cdots R_{i_r}$ starting from T'_{1,i_l} and T_{2,i_l}, respectively, the sets $T'_{1,i_l}, \ldots, T'_{1,s}, T'_{1,t+1}, \ldots, T'_{1,i_r}$ and $T_{2,i_l}, \ldots, T_{2,s}, T_{1,t+1}, \ldots, T_{2,i_r} \notin L$ are passed through. Thus, as for $U_{l+1}\cdots U_r$, it holds for U'_1: $U'_1 \in L(C,C)$, $T_{2,i_l} = T_{2,i_r} = next_\varepsilon(T_{2,i_l}, U'_1)$, $U'_0U'_1 \notin E_A$, and $|U'_1| > 0$. Following the above argumentation it holds for a non-empty word U'_1 of minimal length which satisfies the above properties $0 < |U'_1| \leq 2^{2\cdot n}$.

Let $U'_i := U'_1$ for all $i \geq 2$. Then because of $U'_0 \in L(B,C)$, $U'_1 \in L(C,C)$, and $|U'_1| > 0$ the path $B, U'_0, C, U'_1, C, U'_2, C, \ldots$ is an infinite path in $\mathcal{A}_T$. Due to $U'_0 \notin L(A,C) \cup E_A$, $U'_0U'_1 \notin L(A,C) \cup E_A$, and $C \notin T_{2,i_l} = T_{2,i_r} = next_\varepsilon(T_{2,i_l}, U'_1)$ we know $U'_0\cdots U'_k \notin L(A,C) \cup E_A$ for all $k \geq 0$.

With that, we can specify a computation of algorithm 49 with output "yes" as follows: If $U'_0 = \varepsilon$, then $\varepsilon \in L(B,C)$ and $\varepsilon \notin L(A,C)$, hence $C \in T_1 \cap \overline{T_2}$. In (1) we assign *true* to *stop*. Thus, the first while-loop is skipped. On the other hand, if $U'_0 \neq \varepsilon$, then, if the if-condition in (1) is satisfied, *stop* is assigned to $false$. Since $U'_0 \notin E_A$, the condition of the while-loop is satisfied. In this loop R is choosen according to U'_0. Besides the last iteration of the loop, *stop* is assigned to $false$ if the if-condition in (2) is satisfied. Because of $|U'_0| < 2^{2 \cdot n}$, $stop = false$ (apart from the last iteration), and $U'_0 \notin E_A$, i.e., $T_2 \notin L$, the loop is iterated $|U'_0|$ times. In the last iteration condition (2) is satisfied because of $U'_0 \in L(B,C)$ and $U'_0 \notin L(A,C)$ such that we can assign *true* to *stop*.

If after termination of the while-loop the condition in (3) is satisfied, then there is nothing more to do. Otherwise, it follows that the condition in (4) is satisfied. In the second while-loop R is choosen according to U'_1. If (5) is satisfied, then *stop* is assigned to $false$, apart from the last iteration. Because of $|U'_1| \leq 2^{2 \cdot n}$, $stop = false$, and $U'_0U'_1 \notin E_A$ the condition of the while-loop is satisfied until the last iteration. Furthermore, since $U'_0U'_1 \in L(B,C)$ and $T_{2,i_l} = T_{2,i_r}$, the condition in (5) is satisfied in the last iteration such that *stop* can be assigned to $true$, which leads to the termination of the while-loop. Finally, because of $U'_0U'_1 \notin E_A$ and since the condition in (5) was satisfied, the condition in (6) is satisfied. Thus, the computation terminates with output "yes".

Hence, in any case we were able to construct a computation with output "yes".

Obviously, algorithm 49 is a NPSPACE-algorithm. Since inconsistency can be reduced to subsumption in $\mathcal{FLN}$, Theorem 44 implies

Corollary 50.
Subsumption w.r.t. descriptive semantics in general $\mathcal{ALN}$- and $\mathcal{FLN}$-terminologies PSPACE-complete. □

Chapter 6

Semi-automaton and lfp-semantics

Different from $\mathcal{FL}_0$ in $\mathcal{FLN}$ infinite chains of roles can be required. This will be crucial for the characterization of inconsistency and subsumption in $\mathcal{FLN}$.

6.1 Characterizing the lfp-semantics

Again, the extension of the characterization of the lfp-semantics from $\mathcal{FL}_0$ to $\mathcal{FLN}$ is easy.

Theorem 51 (Characterizing the lfp-semantics w.r.t. $\mathcal{FLN}$).
Let T be an $\mathcal{FLN}$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton, I a lfp-model of T, and A an atomic concept in T. Then it holds for every $d \in dom(I)$: $d \in A^I$ iff

(P1) for all primitive concepts P in T and all words $W \in L(A, P)$ it holds $d \in (\forall W.P)^I$;

(P2) for all maximum-restrictions $(\geq n\ R)$ in T and all words $W \in L(A, (\geq n\ R))$ it holds $d \in (\forall W.(\geq n\ R))^I$;

(P3) for all minimum-restrictions $(\leq n\ R)$ in T and all words $W \in L(A, (\leq n\ R))$ it holds $d \in (\forall W.(\leq n\ R))^I$;

(P4) for all infinite paths of the form $A, W_1, C_1, W_2, C_2, \ldots$ and all individuals $d_1, d_2, d_3, \ldots \in dom(I)$ there is a $n \geq 1$ such that $(d_{n-1}, d_n) \notin W_n^I$. ("Infinite chains are prohibited.")

Proof. Analogous to the proof of [4], Proposition 19 □

Remark 52.
Similar to Remark 23 one can generalize Theorem 51 to $\mathcal{ALN}$-terminologies. ◇

In example 9 it holds $\mathsf{parents}^n \in L(\mathsf{Human}, (\geq 2\ \mathsf{parents}))$ for all $n \geq 0$, i.e., the ω-word parents parents parents... is required by the concept Human. Let I be a lfp-model of T and $d_0 \in \mathsf{Human}^I$ an individual. Following Lemma 21 there are individuals $d_1, d_2, d_3, \ldots$ such that $d_0 \mathsf{parents}^I d_1 \mathsf{parents}^I d_2 \ldots$ Furthermore, $\mathsf{parents}^\omega \in U(\mathsf{Human})$. Thus, for d_0 there is an infinite chain which is prohibited according to (P4). Consequently, Human is inconsistent w.r.t. the lfp-semantics (see also page 50). Thus, the lfp-semantics seems not to be a appropriate semantics in this example.

6.2 Inconsistency w.r.t. lfp-semantics

As mentioned, infinite chains can lead to inconsistent concepts. Again, in order to prove the characterization of inconsistency we need

Definition 53 (canonical lfp-model w.r.t. $\mathcal{FLN}$).
Let T be an $\mathcal{FLN}$-terminology, $\mathcal{A}_T = (\Sigma, Q, E)$ the corresponding semi-automaton, and A an atomic concept in T. The *primitive canonical interpretation* $J = J(A, d_0)$ for A and individual d_0 is defined as for the gfp-semantics (definition 24). The *canonical lfp-model* $I = I(A, d_0)$ is the lfp-model defined by J and T. $\diamond$

In order to derive $d_0 \in A^I$ for the canonical lfp-model I, condition (4.1) has to be extended.

> There is no word $W \in \Sigma^*$ and there are no conflicting number-restrictions $(\geq l\ R)$ and $(\leq r\ R)$, $l > r$, such that W is required by A and $W \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$; furthermore, there is no word $W \in \Sigma^* \cup \Sigma^\omega$ required by A such that $W \in U(A)$. (6.1)

Since J is defined as in definition 24, the properties 1.) – 4.) of J formulated in Lemma 25 hold. Additionally, we show

Lemma 54.
If condition (6.1) holds, then (P1), (P2), (P3), and (P4) of Theorem 51 w.r.t. A, d_0, and J are satisfied. In particular, for the canonical lfp-model I it follows $d_0 \in A^I$.

Proof. According to Lemma 25, 5.) the conditions (P1), (P2), and (P3) of Theorem 22 (and thus, the conditions of Theorem 51) hold, since condition (6.1) implies (4.1).

In order to show $d_0 \in A^I$, by Theorem 51 it is sufficient to show (P4) w.r.t. A, d_0, and J. If (P4) does not hold, then there is an infinite path $A, W_1, C_1, W_2, C_2, \ldots$ in $\mathcal{A}_T$ and there are individuals $d_1, d_2, d_3, \ldots$ such that $d_{i-1} W_i^I d_i$ for all $i \geq 1$. Thus, for $W = W_1 W_2 W_3 \cdots$ we have $W \in U(A)$ and because of $d_0 (W_1 W_2 \cdots W_i)^J d_i$ for all $i \geq 0$, Lemma 25, 1.) implies that every finite prefix of W is required by A. Thus, W is required by A which which contradicts (6.1) since $W \in U(A)$. Hence, (P4) is satisfied and it follows $d_0 \in A^I$. $\square$

Again, the notion "exclusion set" comes into the picture in order to specify decision algorithms for inconsistency and subsumption.

Definition 55 (exclusion set w.r.t. the lfp-semantics in $\mathcal{FLN}$).
Let T denote an $\mathcal{FLN}$-terminology and $\mathcal{A}_T = (\Sigma, Q, E)$ the corresponding semi-automaton without word-transitions. A set of states $F_0 \subseteq Q$ is called *exclusion set* w.r.t. $\mathcal{A}_T$ (and w.r.t. to the lfp-semantics in $\mathcal{FLN}$), if

1.) there exists an ω-word $R_1R_2R_3 \cdots \in \Sigma^\omega$ and there are sets $F_1, F_2, F_3, \ldots \subseteq Q$ as well as number-restrictions $(\geq m_i\ R_i)$, $m_i \geq 1$, for all $i \geq 1$ such that $F_i = next_\varepsilon(F_{i-1}, R_i)$, $i \geq 1$, and $(\geq m_{i+1}\ R_{i+1}) \in F_i$, $i \geq 0$; or

2.) there exists a number $n \geq 0$ as well as a word $R_1 \cdots R_n \in \Sigma^*$ and there are sets $F_1, \ldots, F_n \subseteq Q$ and number-restrictions $(\geq m_i\ R_i)$, $m_i \geq 1$, for all $1 \leq i \leq n$ such that $F_i = next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq n$, and $(\geq m_{i+1}\ R_{i+1}) \in F_i$ for all $0 \leq i < n$. Additionally, there are conflicting number-restrictions $(\geq l\ R)$ and $(\leq r\ R)$, $l > r$, such that $(\geq l\ R), (\leq r\ R) \in F_n$, or there is a defined concept C laying on an ε-cycle such that $C \in F_n$.

◇

In Definition 55, 2.) also P and $\neg P$ for the primitive concept P must be considered if T is an $\mathcal{ALN}$-terminology. In order to formulate decision algorithms for inconsistency and subsumption we first specify a (NPSPACE-)decision algorithm for exclusion sets.

Algorithm 56.

Input: semi-automaton $\mathcal{A}_T = (\Sigma, Q, E)$ without word-transitions for the terminology T; $F_0 \subseteq Q$.

Output: There is a computation with output "yes" iff F_0 is an exclusion set.

Let M denote the set of defined concepts which lay on an ε-cycle.

```
    F := F_0;
    z := 0;
    while z < 2^|Q| do
(1)     if F ∩ M ≠ ∅ or "there are conflicting number-restrictions
            (≥ l R), (≤ r R) ∈ F, l > r" then output "yes";
        if "there are m ≥ 1 and a maximum-restriction (≥ m R) ∈ F"
            then "Guess (non-det.) a (≥ m R) ∈ F where m ≥ 1"
            else output "no";
        F := next_ε(F, R);
        z := z + 1
    end;
(2) output "yes".
```

△

It is not hard to see that this is an NPSPACE-algorithm. Correctness and completeness can easily be shown by using an "pumping-lemma" argument.

Theorem 57 (characterizing inconsistency w.r.t. lfp-semantics).
Let T be an $\mathcal{FLN}$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton without word-transitions,[1] and A an atomic concept in T. The following statements are equivalent:

1.) A is T-inconsistent w.r.t. the lfp-semantics in $\mathcal{FLN}$.

2.) i.) There is a (finite or infinite) word W required by A where $W \in U(A)$; or

ii.) there are conflicting number-restrictions $(\geq l\ R)$ and $(\leq r\ R)$, $l > r$, and there is a word $W \in \Sigma^*$ required by A such that $W \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$.

3.) ε-$closure(\{A\})$ is an exclusion set.

Proof. Equivalence of 1.) and 2.):

"1. $\Rightarrow$ 2.": If the right-hand side of the equivalence is not valid, then according to Lemma 54 there exists the canonical lfp-model $I = I(A, d_0)$ for A and individual d_0 such that $d_0 \in A^I$. Thus, A is consistent.

"1. $\Leftarrow$ 2.": Assume, there is a lfp-model I of T as well as an individual $d_0 \in dom(I)$ such that $d_0 \in A^I$. Furthermore, one of the following cases should hold:
i.) There is a word $W \in U(A)$ required by A.

If W is finite, then following Lemma 21 there is an individual e such that $d_0 W^I e$. Because of $W \in U(A)$ there exists a defined concept C such that $A, W, C, \varepsilon, C, \varepsilon, C, \ldots$ is an infinite path in $\mathcal{A}_T$; furthermore, $d_0 W^I e \varepsilon^I e \varepsilon^I e \cdots$. Hence, Theorem 51, (P4) implies $d_0 \notin A^I$, which is a contradiction to the assumption. If W is the infinite word $R_1 R_2 R_3 \cdots$, then according to Lemma 21 there are individuals $d_1, d_2, d_3, \ldots$ such that $d_0 R_1^I d_2 R_2^I d_2 \cdots$. Again, using $W \in U(A)$ and Theorem 51, (P4) it follows $d_0 \notin A^I$, which also contradicts the assumption.
ii.) There is a word $W \in \Sigma^*$ required by A and there are conflicting number-restrictions $(\geq l\ R)$, $(\leq r\ R)$, $l > r$, such that $W \in L(A, (\geq l\ R)) \cap L(A, (\leq r\ R))$. Thus, Lemma 21 implies that there is an individual e such that $d_0 W^I e$. Because of $d_0 \in A^I$ Theorem 51, (P2) and (P3) imply: $e \in (\geq l\ R)^I$ and $e \in (\geq l\ R)^I$ which is a contradiction since $l > r$.

The equivalence of 2.) and 3.) is not hard to prove using Lemma 5 and Lemma 7. □

Theorem 57, i.) describes the case that for elements in the extension of A infinite chains are required. Thus, Human in Example 9 is inconsistent by Theorem 57, i.), since $\mathsf{parents}^\omega \in U(\mathsf{Human})$ and $\mathsf{parents}^\omega$ is required by Human.

If in Theorem 57, ii.) also P and $\neg P$ are considered, then this Theorem is also valid for $\mathcal{ALN}$-terminologies.

[1]The equivalence of 1.) and 2.) is also valid for arbitrary semi-automaton $\mathcal{A}_T$.

Theorem 58 (Inconsistency w.r.t. lfp-semantics).
Inconsistency w.r.t. lfp-semantics for $\mathcal{ALN}$-($\mathcal{FLN}$-)terminologies is PSPACE-complete and NP-complete for (weak-)acyclic $\mathcal{ALN}$-($\mathcal{FLN}$-)terminologies (see Definition 78)

Proof. As already shown, inconsistency for $\mathcal{ALN}$-($\mathcal{FLN}$-)terminologies is decidable using polynomial space. It remains to show that inconsistency is PSPACE-hard. As already mentioned, in [8] consistency w.r.t. $\mathcal{AL}$-schemas (see Remark 71) and descriptive semantics was shown to be PSPACE-complete. The proof is by reducing the validity of an (arbitrary) quantified Boolean formulae q to consistency of an atomic concept C_s w.r.t. a cyclic $\mathcal{AL}$-schema S_q (for details see [9]). The problem of this reduction is that $\mathcal{AL}$ does not allow for disjunction such that it is straightforward to construct a tree encoding the allocations of the variables. For this purpose, a binary counter is simulated to test the possible allocations of the variables of q. The schema S_q may contain infinite chains starting from C_s although q is valid. For example, if $q = \exists v_1 \forall v_2\, v_1$, then q is obviously valid. But C_s is inconsistent w.r.t. S_q since there is an infinite chain starting from C_s (see Theorem 57). In S_q the word $A_0(A^- A_c A_1)^\omega$ is required by C_S.:

$$C_s \xrightarrow{A_0} \{B_1^1, B_2^{-1}\} \xrightarrow{A^-} \{C_{10}^-, C_{20}^-\} \xrightarrow{A_c} \{C_{11}^-, C_{21}^-\} \xrightarrow{A_1} \{B_1^1, B_2^{-1}\} \xrightarrow{A^-} \cdots$$

Fortunately, one can avoid such infinite chains. It is only necessary to be able to increase the counter for Boolean variables which are existential quantified. For all-quantified variables the following is sufficient: Let

$$q = \cdots \forall v_h \cdots \exists v_i \cdots \forall v_j \cdots$$

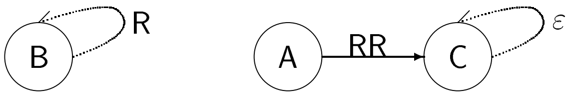
If the counter for v_i changes, then the value of v_h must be propagated. In the construction [9] this is done by propagating B_h^{-1} and B_h^{+1}, respectively, depending on the value of v_h. On the other hand, both values of v_j has to be tested. This is done by propagating B_j^1. For existential quantified variables v_k the B's are propagated according to the counter, i.e., if $k < i$ then B_k^{-1} is propagated and if $k > i$, then the propagation coincides with the current value of v_k. Thus, for the atomic concepts C_{in}^- and C_{in}^+, v_i all quantified, the concept inclusions $C_{in}^- \sqsubseteq \exists A_i$ and $C_{in}^+ \sqsubseteq \exists A_i$ (and hence, $C_{in}^- \sqsubseteq \forall A_i.B_i^1$ and $C_{in}^+ \sqsubseteq \forall A_i.B_i^1$) in S_q can be dispensed with. This yields a schema without infinite chains. Consequently, C_s (in the modified schema) is consistent w.r.t. the gfp-semantics iff C_s is consistent w.r.t. the lfp-semantics. Thus, with this modified schema the reduction of validity of quantified Boolean formulae to consistency w.r.t. $\mathcal{AL}$-schemas also holds for the lfp-semantics. Hence, Theorem 75 implies the PSPACE-hardness of consistency for general $\mathcal{ALN}$-($\mathcal{FLN}$-)terminologies.

NP-completeness of inconsistency for (weak-)acyclic terminologies is shown in Theorem 79. □

6.3 Subsumption w.r.t. lfp-semantics

Theorem 51 implies that for $A \sqsubseteq_{lfp,T} B$ the conditions $L(B,P) \subseteq L(A,P)$ for all primitive concepts P, $L(B,(\geq n\ R)) \subseteq L(A,(\geq n\ R))$ for all maximum-restrictions $(\geq n\ R)$, $L(B,(\leq n\ R)) \subseteq L(A,(\leq n\ R))$ for all minimum-restrictions $(\leq n\ R)$, and $U(B) \subseteq U(A)$ are sufficient. Example 19 shows that the conditions for primitive concepts and number-restrictions are not necessary. The condition $U(B) \subseteq U(A)$ is not necessary as well, as the following example shows.

Example 59.
Let T be the terminology which is defined by the corresponding automaton $\mathcal{A}_T$ as follows:



Although $\mathsf{RRR}\cdots \in U(\mathsf{B})$ and $\mathsf{RRR}\cdots \notin U(\mathsf{A})$, thus $U(\mathsf{B}) \not\subseteq U(\mathsf{A})$, it holds $\mathsf{A} \sqsubseteq_{lfp,T} \mathsf{B}$. Proof: Let I be a lfp-model of T and d an individual such that $d \in \mathsf{A}^I$. Then d cannot have RR-successors. If the individual e is an RR-successor of d, then because of the infinite path $\mathsf{A}, \mathsf{RR}, \mathsf{C}, \varepsilon, \mathsf{C}, \varepsilon, \mathsf{C}, \varepsilon, \mathsf{C}, \ldots$, $d(\mathsf{RR})^I e$, and $e\varepsilon^I e$ condition (P4) implies that d is an element of A^I. For B and d the conditions (P1), (P2), and (P3) are obviously satisfied. The only infinite path starting from B is $\mathsf{B}, \mathsf{R}, \mathsf{B}, \mathsf{R}, \mathsf{B}, \mathsf{R}, \ldots$ Since d has no RR-successor, (P4) w.r.t. B and d is satisfied. Hence, $d \in \mathsf{B}^I$. ◇

In this example the subsumption relation $A \sqsubseteq_{lfp,T} B$ holds because the words in RRR^* exclude A in the following sense.

Definition 60 (exclusion w.r.t. the lfp-semantics in $\mathcal{FLN}^r$).
Let T be an $\mathcal{FLN}^r$-terminology,[2] $\mathcal{A}_T = (\Sigma, Q, E)$ the corresponding semi-automaton, and A an atomic concept in T. The finite word $W \in \Sigma^*$ *excludes* A if one of the following conditions hold:

1.) There is a (finite or infinite) word $\alpha \in U(A)$ as well as a word $V \in \Sigma^*$ which is prefix of W and α such that α is required by A starting from V.

2.) There is a prefix $V \in \Sigma^*$ of W as well as a word $V' \in \Sigma^*$, and there are conflicting number-restriction $(\geq l\ R)$ and $(\leq r\ R)$, $l > r$, such that $VV' \in L(A,(\geq l\ R)) \cap L(A,(\leq r\ R))$ and VV' is required by A starting from V.

The ω-word $W \in \Sigma^\omega$ *excludes* A if there is a finite prefix of W which excludes A or if $W \in U(A)$.

Furthermore, let $E_A := \{W \in \Sigma^*;\ W \text{ excludes } A\}$, $E_{A,\omega} := \{W \in \Sigma^* \cup \Sigma^\omega;\ W \text{ excludes } A\}$, and $E^f_{A,\omega} := \{W \in \Sigma^* \cup \Sigma^\omega;\ \text{a finite prefix of } W \text{ excludes } A\}$. ◇

[2] By considering P and $\neg P$ beside conflicting number-restrictions the definition can be generalized to $\mathcal{ALN}^r$-terminologies.

Note that as for the gfp-semantics the notion "exclusion" is only defined for $\mathcal{FLN}^r$-terminologies rather than for $\mathcal{FLN}$-terminologies.

If A is inconsistent, then it holds $E_A = \Sigma^*$ and $E^f_{A,\omega} = E_{A,\omega} = \Sigma^* \cup \Sigma^\omega$, since by Theorem 57 the empty word ε already excludes A. Conversely, if $E_A = \Sigma^*$ ($E^f_{A,\omega} = \Sigma^* \cup \Sigma^\omega$ or $E_{A,\omega} = \Sigma^* \cup \Sigma^\omega$), then A is excluded by ε. Thus, Theorem 57 and Definition 60 imply that A is inconsistent. An important statement concerning Definition 60 is captured in

Lemma 61.
Using the denotations and assumptions of Definition 60 it holds for the lfp-model I of T:

1.) If $W \in \Sigma^* \cup \Sigma^\omega$ as well as $V \in \Sigma^*$ a prefix of W with the properties 1.) or 2.) in Definition 60, and $d, e \in dom(I)$ individuals where $dV^I e$, then $d \notin A^I$.

2.) If $W \in U(A)$ is an ω-word $W = R_1R_2R_3\cdots$ and $d_0, d_1, d_2, \ldots$ individuals where $d_0R_1^I d_1 R_2^I d_2 \cdots$, then $d_0 \notin A^I$.

Proof. 1.) The proof is very similar to the proof of Lemma 33.
2.) Using Theorem 51, (P4) it follows immediately $d_0 \notin A^I$. □

As for the other two semantics the following definition is useful to prove the characterization of subsumption.

Definition 62 (extended canonical lfp-model in $\mathcal{FLN}^r$).
Let T be an $\mathcal{FLN}^r$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton, and A an atomic concept in T. Furthermore, let W denote a word in $\Sigma^* \cup \Sigma^\omega$, $r \in \mathbb{N}$, and $R \in \Sigma$. For $W \in \Sigma^*$ the *extended primitive canonical interpretations* $J' = J(A, d_0, W)$ $(r = 0)$ and $J' = J(A, d_0, W, R, r)$ $(r > 0)$, respectively, are defined as for the gfp-semantics in Definition 34. For $W \in \Sigma^\omega$ the primitive interpretation $J' = J(A, d_0, W)$ is defined as for the descriptive semantics in Definition 47. The *extended canonical lfp-models* $I' = I(A, d_0, W)$ and $I' = I(A, d_0, W, R, r)$ are the lfp-models defined by T, $J' = J(A, d_0, W)$, and $J' = J(A, d_0, W, R, r)$, respectively. ◇

In order to prove $d_0 \in A^{I'}$, i.e., the validity of (P1) – (P4) of Theorem 51 w.r.t. A, d_0, T, and J', we need the following condition:

> The atomic concept A is consistent and is not excluded by W; for $r > 0$ — in this case W should be a finite word — also WR should not exclude A. Furthermore, for all $l < r$ it is assumed $W \notin L(A, (\leq l\ R))$. (6.2)

If one finite word $W \in \Sigma$ excludes the atomic concept A w.r.t. to Definition 32, then also w.r.t. Definition 60. Thus, condition (6.2) implies the conditions (4.3) and (5.2). Furthermore, since the extended primitive canonical interpretation for the lfp-semantics is defined as for the other two semantics, Lemma 35 and the statements

8.) and 9.) also hold w.r.t. Definition 62. Thus, in order to prove $d_0 \in A^{I'}$ (I' defined as in Definition 62) it is sufficient to verify (P4) of Theorem 51 w.r.t. A, d_0, T, and J'. This statement is proven in

Lemma 63.
Using the denotations of Definition 62 condition (6.2) implies $d_0 \in A^{I'}$.

Proof. As mentioned above we only have to show (P4) for A, d_0, T, and J'. Assume that (P4) does not hold. Thus, there is an infinite path of the form $A, V_1, C_1, V_2, C_2, \ldots$ in $\mathcal{A}_T$, and there are individuals $e_1, e_2, e_3, \ldots \in dom(J')$ such that $(e_{i-1}, e_i) \in V_i^{J'}$ for all $i \geq 1$, $e_0 := d_0$. Let $I = I(A, d_0)$ be the canonical model of A and d_0 as well as $J = J(A, d_0)$ the corresponding primitive interpretation. If $e_1, e_2, e_3, \ldots \in dom(I)$, then (P4) of Theorem 51 is violated w.r.t. A, d_0, and J, since by Lemma 35, 5.) it holds $(e_{i-1}, e_i) \in V_i^J$. On the other hand, following Lemma 54 it holds $d_0 \in A^I$ which by Theorem 51 implies the validity of Theorem 51, (P4) w.r.t. A, d_0, and J. Thus, we have a contradiction.

Consequently, there is a $i \geq 1$ where $e_i \in dom(J') \setminus dom(J)$. Following Lemma 35, 4.) all elements in $dom(J') \setminus dom(J)$ are (direct or indirect) successors of d_1 (see Definition 34, $d_0 U^J d_1$). Thus, for $V = V_1 V_2 V_3 \cdots$ and $e_i \in dom(J') \setminus dom(J)$ it holds: $V = W$ (for $r = 0$), $V = WR$ (for $r > 0$), or $V = UXY$ where $X \in \Sigma^+$, $Y \in \Sigma^* \cup \Sigma^\omega$, UX maximal prefix of W (for $r = 0$) and of WR (for $r > 0$)(see proof of Lemma 35, 9.)). In the case of $V = W$ or $V = WR$ the concept A is excluded by W and WR, respectively, because of $V \in U(A)$, which is a contradiction to the assumption. Thus, it holds $V \neq W$ and $V \neq WR$, respectively, and thus $V = UXY$. By the Definition of J' the word V is required by A starting from UX. Because of $V \in U(A)$ and since UX is a prefix of W (resp., WR) A is excluded by W (resp., WR), which is a contradiction to the assumption. Thus, (P4) holds, and hence it follows $d_0 \in A^{I'}$. □

Now we are prepared to show

Theorem 64 (Characterizing subsumption w.r.t. lfp-semantics).
Let T be an $\mathcal{FLN}^r$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton, and A, B atomic concepts in T. Then it holds $A \sqsubseteq_{lfp,T} B$ iff

1.) $L(B, P) \subseteq L(A, P) \cup E_A$ for primitive concepts P in T;

2.) $L(B, (\geq l\ R)) \subseteq \bigcup_{r \geq l} L(A, (\geq r\ R)) \cup E_A$ for all maximum-restrictions of the form $(\geq l\ R)$ in T where $l > 0$;

3.) $L(B, (\leq l\ R)) \cdot R \subseteq (\bigcup_{r \leq l} L(A, (\leq r\ R))) \cdot R \cup E_A$ for all minimum-restrictions of the form $(\leq l\ R)$ in T; and

4.) $U(B) \subseteq U(A) \cup E_{A,\omega}$.

Proof. "⇒": We assume that (at least) one of the conditions 1.) – 4.) are not valid and prove that this implies $A \not\sqsubseteq_{lfp,T} B$.

The cases 1.), 2.), and 3.) can be treated using Lemma 63 and Theorem 51 as in the proof of Theorem 36.

For 4.) we assume $U(B) \not\subseteq U(A) \cup E_{A,\omega}$.
Thus, there is a word $W \in U(B) \setminus (U(A) \cup E_{A,\omega})$. Because of $W \notin E_{A,\omega}$ the concept A is consistent, since otherwise $E_{A,\omega} = \Sigma^* \cup \Sigma^\omega$. Furthermore, A is not excluded by W. Consequently, by Lemma 63 there exists the lfp-model $I' = I'(A, d_0, W)$ such that $d_0 \in A^{I'}$.

If W is finite, then there is $d \in dom(I')$ such that $d_0 W^I d$. Because of $W \in U(B)$ there is a (defined) concept C which lays on an ε-cycle. Thus, $A, W, C, \varepsilon, C, \varepsilon, C, \ldots$ is an infinite path in $\mathcal{A}_T$. Using $d_0 W^{I'} d$ and $d \varepsilon^{I'} d$ Theorem 64, (P4) w.r.t B and d_0 implies $d_0 \notin B^{I'}$.

If W is the ω-word $R_1 R_2 R_3 \cdots$, then there are individuals $d_1, d_2, d_3, \ldots \in dom(I')$ such that $d_0 R_1^{I'} d_1 R_2^{I'} d_2 \cdots$. Because of $W \in U(B)$ and (P4) w.r.t. B and d_0 this implies $d_0 \notin B^{I'}$. Hence, $A \not\sqsubseteq_{lfp,T} B$.

"$\Leftarrow$": We assume that the right-hand side of the equivalence is valid and additionally $A \not\sqsubseteq_{lfp,T} B$. Thus, there is a lfp-model of T and an individual $d_0 \in dom(I)$ such that $d_0 \in A^I \setminus B^I$. Because of $d_0 \notin B^I$ one of the conditions (P1), (P2), (P3), and (P4) of Theorem 51 w.r.t. B, d_0, T, and I do not hold.

The cases (P1), (P2), and (P3) can be treated with the help of Theorem 51 and Lemma 61 as in the proof of Theorem 36 ("$\Leftarrow$").

If (P4) is not valid, then there is an infinite path of the form $B, W_1, C_1, W_2, C_2, \ldots$ 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 $n \geq 1$ (*). Because of $W = W_1 W_2 W_3 \cdots \in U(B)$ and $U(B) \subseteq U(A) \cup E_{A,\omega}$ we know $W \in U(A)$ or $W \in E_{A,\omega}$. Now $W \in U(A)$, (*) and (P4) w.r.t. A and d_0 imply $d_0 \notin A^I$, which contradicts the assumption. Furthermore, using (*) and Lemma 61 $W \in E_{A,\omega}$ imply $d_0 \notin A^I$, which is also a contradiction. Hence, A is subsumed by B. □

In order to generalize Theorem 64 to $\mathcal{ALN}^r$-terminologies see Remark 37.

Again, restricting Definition 60 to $\mathcal{FLN}^r$ we can characterize the sets E_A and $E^f_{A,\omega}$ using exclusion sets.

Lemma 65.
Let T be a $\mathcal{FLN}^r$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton without word-transitions, and A an atomic concept in T. Then is holds:

1.) $E_A = \{W \in \Sigma^*;$ an exclusion set is reachable by W starting from $A\}$;

2.) $E^f_{A,\omega} = \{W \in \Sigma^* \cup \Sigma^\omega;$ an exclusion set is reachable by W starting from $A\}$.

"Reaching an exclusion set" is defined in Definition 38.

Proof. The proof is very similar to the proof of Lemma 39, and therefore is omitted here. □

This characterization of the set E_A allows to decide the conditions 1.), 2.), and 3.) of Theorem 64 as for the gfp-semantics using Algorithm 40 and the modified versions (see page 35). Note that the condition $T_2 \notin \{F \subseteq Q;\ F$ exclusion set$\}$ must be interpreted w.r.t. Definition 55. Algorithm 56 shows that this condition is decidable using polynomial space. Thus, the problem to decide 1.), 2.), and 3.) of Theorem 64 is decidable using polynomial space.

It remains to formulate a (NPSPACE-)decision algorithm for condition 4.) of Theorem 64, $U(B) \subseteq U(A) \cup E_{A,\omega}$. Because of $E_{A,\omega} = E^f_{A,\omega} \cup U(A)$ this condition is equivalent to $U(B) \subseteq U(A) \cup E^f_{A,\omega}$.

The algorithm guesses non-deterministicaly a word $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$ if such a word exists. If there is a finite word $U_0 = W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$, then there is also a word of length $|U_0| < 2^{2 \cdot n}$. If there is a ω-word $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$, the algorithm first guesses a word U_0, $|U_0| < 2^{2 \cdot n}$, such that $next_\varepsilon(A, U_0) = \emptyset$ and $next_\varepsilon(B, U_0) \neq \emptyset$. Then it guesses a word U_1 of length $2^n - 1$ where $next_\varepsilon(next_\varepsilon(B, U_0), U_1) \neq \emptyset$. Now, using U_0 and U_1 it is possible to construct an ω-word W such that $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$.

Algorithm 66.

Input: semi-automaton $\mathcal{A}_T = (\Sigma, Q, E)$ without word-transitions for the terminology T; atomic concepts A, B in T

Output: There is a computation with output "yes" iff $U(B) \not\subseteq U(A) \cup E_{A,\omega} (= U(A) \cup E^f_{A,\omega})$

Let n denote the size of Q, M the set of atomic concepts in T which lay on an ε-cycle, and L the set of exclusion sets.

```
T_1:=ε-closure({B});
T_2:=ε-closure({A});
z:=0;
(* Guess U_0 *)
while z < 2^(2·n) − 1 and T_2 ∉ L and (T_1 ∩ M = ∅ or T_2 ∩ M ≠ ∅) do begin
    z:=z+1;
    Guess (non-det.) R ∈ Σ;
    T_1 := next_ε(T_1, R);
    T_2 := next_ε(T_2, R)
end;
(1) If T_2 ∉ L then begin
(2)     If T_1 ∩ M ≠ ∅ and T_2 ∩ M = ∅ then output "yes"
        else
(3)         if (* z = 2^(2·n) − 1 and *) T_2 = ∅ then begin
                (* Guess U_1 *)
                z := 0;
                while z < 2^n − 1 do begin
                    z := z + 1;
                    Guess (non-det.) R ∈ Σ;
                    T_1 := next_ε(T_1, R);
```

```
                    (*T_2 := next_ε(T_2, R) = ∅;*)
                 end; (*while*)
(4)              If T_1 ≠ ∅ then output "yes"
                 else output "no"
             end
             else output "no"
         end(*if*)
     end
     else output "no".
```
△

Soundness: If the algorithm terminates with "yes", then two cases have to be distinguished:

i) (Ouput "yes" in (2)) There is a finite word $W = R_1 \cdots R_m \in \Sigma^*$, $m < 2^{2 \cdot n}$, such that for $T_1 := next_\varepsilon(B, W)$ and $T_2 := next_\varepsilon(A, W)$ it holds: $T_1 \cap M \neq \emptyset$ and $T_2 \cap M = \emptyset$. Thus, there is an atomic concept $C \in T_1 \cap M$. Consequently, there is an infinite path of the form $B, W, C, \varepsilon, C, \varepsilon, C, \ldots$, and hence, $W \in U(B)$. If $W \in U(A)$, then there would be a (defined) concept C' which lays on an ε-cycle such that $A, W, C', \varepsilon, C', \varepsilon, C', \ldots$ is an infinite path in $\mathcal{A}_T$. Thus, it would hold $C' \in T_2 \cap M$ in contradiction to $T_2 \cap M = \emptyset$. Hence, $W \notin U(A)$. Furthermore, the algorithm ensures for $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$, $0 \leq i \leq m$, that $T_{2,i} \notin L$ for all $0 \leq i \leq m$ (because of the condition in the first while-loop and because of (1)). By Lemma 65 this implies $W \notin E^f_{A,\omega}$. Thus, we have $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$, and consequently $U(B) \not\subseteq U(A) \cup E^f_{A,\omega}$.

ii) (Output "yes" in (4)) There is a word $W = R_1 \cdots R_m \in \Sigma^*$, $m = 2^{2 \cdot n} - 1 + 2^n - 1$, such that for $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ and $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$, $0 \leq i \leq m$, it holds: $T_{1,i} \neq \emptyset$ for all $0 \leq i \leq m$, $T_{2,j} = \emptyset$ for all $2^{2 \cdot n} - 1 \leq j \leq m$ as well as $T_{2,i} \notin L$ for all $0 \leq i \leq m$ (note: $\emptyset \notin L$). Because of $|2^Q \setminus \{\emptyset\}| = 2^n - 1$ there are non-negative integers l and r such that $2^{2 \cdot n} - 1 \leq l < r \leq m$ and $T_{1,l} = T_{1,r}$. Since for the ω-word $\alpha := R_1 \cdots R_{2^{2 \cdot n} - 1} \cdots R_l (R_{l+1} \cdots R_r)^\omega$ starting from $T_{1,0}$ the sets $T_{1,0}, T_{1,1}, \ldots, T_{1,l}, T_{1,l+1}, \ldots, T_{1,r-1}, T_{1,r}, T_{1,l+1}, \ldots, T_{1,r}, \ldots$ are passed through, which, in addition, are all non-empty, it follows $\alpha \in U(B)$ (Lemma 7). Furthermore, for α starting form $T_{2,0}$ the sets $T_{2,0}, T_{2,1}, \ldots, T_{2,2^{2 \cdot n} - 2}, \emptyset, \emptyset, \emptyset, \ldots$ are passed through. Now Lemma 7 implies $\alpha \notin U(A)$. Since this sequence of sets contain no exclusion sets, it follows from Lemma 65: $\alpha \notin E^f_{A,\omega}$. Thus, we have $\alpha \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$, and hence $U(B) \not\subseteq U(A) \cup E^f_{A,\omega}$.

Completeness: If $U(B) \not\subseteq U(A) \cup E^f_{A,\omega}$, then we distinguish the cases iii) and iv).

iii) (Verification of a computation with output "yes" in (2)) There is a finite word $W = R_1 \cdots R_m \in \Sigma^*$ such that $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$. Let $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ and $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$ for all $0 \leq i \leq m$. If $m \geq 2^{2 \cdot n}$, then because of $|2^Q \times 2^Q| = 2^{2 \cdot n}$ there are numbers l and r such that $0 \leq l < r \leq 2^{2 \cdot n}$ as well as $T_{1,l} = T_{1,r}$ and $T_{2,l} = T_{2,r}$. For $W' = R_1 \cdots R_l R_{r+1} \cdots R_m$ the tuples $(T_{1,0}, T_{2,0}), (T_{1,1}, T_{2,1}), \ldots, (T_{1,l}, T_{2,l}), (T_{1,r+1}, T_{2,r+1}), \ldots, (T_{1,m}, T_{2,m})$ are passed through. Because of $T_{1,m} \cap M \neq \emptyset$ (since $W \in U(B)$) it follows as in i): $W' \in U(B)$.

Analogously to i), $T_{2,m} \cap M = \emptyset$ (because of $W \notin U(A)$) implies $W' \notin U(A)$. Since $T_{2,i}$ is not contained in L for any i, $0 \leq i \leq m$ (because of $W \notin E^f_{A,\omega}$ and Lemma 65) we also know $W' \notin E^f_{A,\omega}$. Thus, for W' we have $W' \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$. Without loss of generality we can assume $m = |W| < 2^{2 \cdot n}$. With that, we can define a computation of algorithm 66 with output "yes" in (2). In the first while-loop of the algorithm we choose $R = R_i$ in the ith iteration. As a consequence of the definition of W, in every iteration step it holds $T_2 \notin L$. If the while-loop is iterated m times, then at the end of the mth iteration it holds $T_1 \cap M \neq \emptyset$, $T_2 \cap M = \emptyset$, and $T_2 \notin L$. Thus, in (2) the algorithm terminates with output "yes". If the while-loop is iterated less than m-times, then before the mth iteration it must hold $T_1 \cap M \neq \emptyset$ and $T_2 \cap M = \emptyset$. Again, we get the output "yes" in (2).
Or:

iv) (Verification of a computation with output "yes" in (4)) There is an ω-word $W = R_1 R_2 R_3 \cdots \in \Sigma^\omega$ such that $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$. For $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ and $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$, $i \geq 0$, it holds: (*) $T_{1,i} \neq \emptyset$ for all $i \geq 0$ (because of $W \in U(B)$ and Lemma 7); furthermore, there is a number $k \geq 0$ such that $T_{2,k} = \emptyset$ (because of $W \notin U(A)$ and Lemma 7); finally, we have $T_{2,i} \notin L$ for all $i \geq 0$ because of $W \notin E^f_{A,\omega}$ and Lemma 65.

Due to $|2^Q \times 2^Q| = 2^{2 \cdot n}$ there are numbers l and r where $0 \leq l < r \leq 2^{2 \cdot n}$ as well as $T_{1,l} = T_{1,r}$ and $T_{2,l} = T_{2,r}$. Thus, for $W' := R_1 \cdots R_l R_{r+1} R_{r+2} R_{r+3} \cdots \in \Sigma^\omega$ the tuples $(T_{1,0}, T_{2,0}), (T_{1,1}, T_{2,1}), \ldots, (T_{1,l}, T_{2,l}), (T_{1,r+1}, T_{2,r+1}), (T_{1,r+2}, T_{2,r+2}), \ldots$ are passed through. Statement (*) implies $W' \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$. Thus, we can assume $k < 2^{2 \cdot n}$ without loss of generality.

We now specify for W a computation of algorithm 66 with output "yes". In the first while-loop we choose $R = R_i$ in the ith iteration. Because of $T_{2,i} \notin L$ for all $i \geq 0$ it holds $T_2 \notin L$ in every iteration. If $T_1 \cap M \neq \emptyset$ and $T_2 \cap M = \emptyset$, then because of $T_2 \notin L$ in (2) "yes" is displayed, and nothing more is to show. Otherwise, the algorithm continues in (3). By the assumption it holds $k < 2^{2 \cdot n}$. Since the first while-loop was iterated $(2^{2 \cdot n} - 1)$ times, it follows $T_2 = \emptyset$. Consequently, the second loop is iterated $(2^n - 1)$-times. In the ith iteration we choose $R = R_{2^{2 \cdot n} - 1 + i}$. After termination of the loop it holds $T_1 \neq \emptyset$ because of $T_{1,i} \neq \emptyset$ for all $i \geq 0$. Thus, the algorithm terminates in (4) with output "yes".

Algorithm 66 shows that the problem $U(B) \subseteq U(A) \cup E_{A,\omega}$ is decidable using polynomial space. Thus, since subsumption w.r.t. the lfp-semantics in $\mathcal{FL}_0$ is PSPACE-complete [4] (and since inconsistency w.r.t. lfp-semantics in $\mathcal{FLN}$ is PSPACE-complete, Theorem 58), it follows

Corollary 67.
Subsumption w.r.t. the lfp-semantics in general $\mathcal{ALN}$- and $\mathcal{FLN}$-terminologies is PSPACE-complete. □

Chapter 7

$\mathcal{SLN}$-schemas and $\mathcal{ALN}$-terminologies

In [7], terminologies are divided into a schema and a view part—following (object-oriented) databases. Schemas allow to specify necessary conditions for atomic concepts (concept inclusions instead of concept definitions) as well as simple necessary conditions for roles (role inclusions, which are not allowed in terminologies considered here). Such necessary conditions are for example subsumption relations for atomic concepts and range restrictions for roles. The schema merely restricts the number of admissible models of the terminology which implies that the meaning of schemas is captured by descriptive semantics. In the view part of the terminology concepts are *defined* with the help of schema concepts. For this reason, fixed-point semantics is used for this part.

Knowledge and database engineers are interested in validity of schemas as well as subsumption w.r.t. schemas. In [7], $\mathcal{SL}_{dis}$-schemas have been introduced, which can express constraints frequently occurring in the static part of object-oriented database schemas. Furthermore, a special PSPACE-decision algorithm has been developed for deciding (local) validity of these schemas. In this paper, it is formally shown that inconsistency, validity, and subsumption for $\mathcal{SLN}$-schemas (and hence, for $\mathcal{SL}_{dis}$-schemas) can be reduced to corresponding problems for $\mathcal{ALN}$-terminologies. In addition, it is possible—as already mentioned—to prove some hardness results for $\mathcal{ALN}$-terminologies using this reduction.

7.1 $\mathcal{SLN}$-schemas

An $\mathcal{SLN}$-schema defines necessary conditions for atomic concepts and roles.

Definition 68 ($\mathcal{SLN}$-schemas).
Syntax:
A $\mathcal{SLN}$*-schema* S consists of a finite set of concept inclusions and role inclusions.

Let A, B be atomic concepts in S (atomic S-concepts), R a role name in S (S-role), and n a non-negative integer. *Concept inclusions* are of the form $A \sqsubseteq C$

where C denotes a concept of the form

$$B \mid \neg B \mid \forall R.B \mid (\geq n\ R) \mid (\leq n\ R)$$

Role inclusions are of the form $R \sqsubseteq A \times B$ for a role name R and atomic concepts A, B. An atomic concept A with at least one concept inclusion of the form $A \sqsubseteq C$ in S is called *defined*, and otherwise *primitive*. Analogous, we refer to *defined* and *primitive* roles. We only allow for primitive negation, i.e., for concept inclusions $A \sqsubseteq \neg B$ where B is primitive.
Semantics:
An interpretation I of S is defined as in Definition 11. An interpretation I is a *(S-)model* of S if all concept inclusions $A \sqsubseteq C$ and all role inclusions $R \sqsubseteq A \times B$ in S are satisfied, i.e., $A^I \subseteq C^I$ and $R^I \subseteq A^I \times B^I$. The semantics of S is defined by the descriptive semantics, i.e., the set of all S-models.

A concept A is *consistent* w.r.t. S (*S-consistent*) if there is a model I of S such that $A^I \neq \emptyset$. A schema S is *locally valid* if every atomic concept in S is consistent. We call a schema S *valid* if there is a S-model I with $A^I \neq \emptyset$ for *every* atomic concept A in S.

The concept A is *subsumed* by B w.r.t. S (*S-subsumed*) if for all S-models I it holds $A^I \subseteq B^I$. ◇

The following is an example of an $\mathcal{SLN}$-schema describing a extract of a company environment [7].

Example 69.
S:

$$
\begin{array}{rcl}
\mathsf{Employee} & \sqsubseteq & (\geq 1\ \mathsf{salary}) \\
\mathsf{Employee} & \sqsubseteq & (\leq 1\ \mathsf{salary}) \\
\mathsf{Employee} & \sqsubseteq & (\geq 1\ \mathsf{boss}) \\
\mathsf{Manager} & \sqsubseteq & \mathsf{Employee} \\
\mathsf{Manager} & \sqsubseteq & \forall \mathsf{salary.HighSalary} \\
\mathsf{High} & \sqsubseteq & \mathsf{Salary} \\
\mathsf{salary} & \sqsubseteq & \mathsf{Employee} \times \mathsf{Salary} \\
\mathsf{boss} & \sqsubseteq & \mathsf{Employee} \times \mathsf{Manager}
\end{array}
$$

◇

The following fact can be used to reduce validity of schemas to consistency in terminologies:

Proposition 70.
An $\mathcal{SLN}$-schema is valid iff it is locally valid.

Proof. The proof is analogous to the proof for $\mathcal{SL}_{dis}$-schemas (see below) in [7]. Let I_1, I_2 be models of the $\mathcal{SLN}$-schema S with disjoint domains. By induction over the structure of the concept terms it is easy to verify that $D^I = D^{I_1} \dot\cup D^{I_2}$ for every $\mathcal{ALN}$-concept D where I denotes the union of I_1 and I_2, i.e., $dom(I) := dom(I_1) \cup dom(I_2)$, $C^I := C^{I_1} \cup C^{I_2}$ for all atomic S-concepts C, and $R^I := R^{I_1} \cup R^{I_2}$ for all S-roles R. Furthermore, it is not hard to prove that I is a model of S.

Now if S is locally valid, one only has to construct the union of all S-models I_A for the atomic S-concepts A, which satisfy $A^{I_A} \neq \emptyset$. The other direction of the proof is trivial. □

A schema S' has the *same properties* w.r.t. consistency, (local) validity, and subsumption as the schema S if the following conditions hold:

1.) For all atomic S-concepts A it holds: A is S-consistent iff A is S'-consistent;

2.) S is locally valid iff for every atomic S-concept A there is a S'-model I_A such that $A^{I_A} \neq \emptyset$;

3.) S is valid iff there is a S'-model I such that $A^I \neq \emptyset$ for all atomic S-concepts; and

4.) for all atomic S-concepts A, B it holds: $A \sqsubseteq_S B$ iff $A \sqsubseteq_{S'} B$.

Obviously 1.) implies 2.) and using Proposition 70 by 2.) it follows 3.).

Remark 71 ($\mathcal{ALN}$-, $\mathcal{SL}_{dis}$-, and $\mathcal{AL}$-schemas).
$\mathcal{SLN}$-schemas as defined above only allow for "flat" concepts on the right-hand side of a concept inclusion, whereas$\mathcal{ALN}$*-schemas* allow for arbitrary $\mathcal{ALN}$-concepts on the right-hand side of concept inclusions. However, it can easily be shown that for every $\mathcal{ALN}$-schemas S one can construct in time linear in the size S an $\mathcal{SLN}$-schema with the same properties w.r.t. consistency, (local) validity, and subsumption. For this purpose, first every $\mathcal{ALN}$-concept inclusion of the form $A \sqsubseteq D_1 \sqcap \cdots \sqcap D_n$ is replaced by the inclusions $A \sqsubseteq D_1, \ldots, A \sqsubseteq D_n$. Secondly, every $\mathcal{ALN}$-concept inclusion of the form $A \sqsubseteq \forall RW.D$ is replaced by $A \sqsubseteq \forall R.A_1$ and $A_1 \sqsubseteq \forall W.D$ (where A_1 denotes a new atomic concept) until there are only $\mathcal{SLN}$-concept inclusions left.

As $\mathcal{SLN}$-schemas, $\mathcal{SL}_{dis}$*-schemas*, which have been introduced in [7], allow only for "flat" concept inclusions, however, number-restrictions are restricted to the form $(\geq 1\ R)$ and $(\leq 1\ R)$. On the other hand, $\mathcal{SL}_{dis}$-schemas allow for concept inclusions of the form $A \sqsubseteq \neg B$ where B may be a defined concept in the schema. Such inclusions can be substituted by $A \sqsubseteq \neg P_B$ and $B \sqsubseteq P_B$ for a newly introcuded primitive concept P_B. The resulting schema has the same properties w.r.t. consistency, (local) validity, and subsumption as S. Consequently, $\mathcal{SLN}$-schemas and $\mathcal{SL}_{dis}$-schemas only differ in terms of number-restrictions.

$\mathcal{AL}$*-schemas* introduced in [8] coincide with $\mathcal{SL}_{dis}$-schemas apart from the fact that they do not allow for role inclusions. ◇

7.2 Reducing schemas to terminologies

In this section we construct an $\mathcal{ALN}$-terminology T_S from an $\mathcal{SLN}$-schema S such that the properties of T_S are the same as for S w.r.t. consistency and subsumption. Before defining T_S, we transform S into a schema S' that does not contain role inclusions. Role inclusions only have to be taken into account if role successors are required, otherwise they can be neglected. The formal definition of S' is as follows: The schema S' contains all concept inclusions of S of the form $A \sqsubseteq B$, $A \sqsubseteq \neg B$, $A \sqsubseteq \forall R.B$, and $A \sqsubseteq (\leq n\ R)$. For every $A \sqsubseteq (\geq n\ R) \in S$, $n \geq 1$, we distinguish two cases: (a) R is not defined in S; (b) R is defined in S. In case (a), $A \sqsubseteq (\geq n\ R)$ is contained in S'. In case (b), S' contains the concept inclusion $A \sqsubseteq (\geq n\ R) \sqcap \forall R.C_2 \sqcap C_1$ for every $R \sqsubseteq C_1 \times C_2 \in S$. The schema S' contains no other inclusions than these, in particular no role inclusions.

Definition 72 (the terminology T_S).
Let S be an $\mathcal{SLN}$-schema and S' as defined above. For every defined concept A in S a concept definition for A in T_S is constructed as follows:
Let $A \sqsubseteq C_1, \ldots, A \sqsubseteq C_n$ be all concept inclusions belonging to the defined concept A in S'. Let $\overline{A}$ be a new (primitive) concept. Then the concept definition for A in T_S is of the form $A = \overline{A} \sqcap C_1 \sqcap \cdots \sqcap C_n$. ◇

Obviously, we can construct T_S from S in time linear in the size of S.

Instead of defining concept definitions of the form $A = \overline{A} \sqcap \cdots$ in Definition 72, where $\overline{A}$ denotes a newly introduced primitive concept, one could also consider concept definition $A = A \sqcap \cdots$. However, in this case Theorem 77 is only valid for the descriptive semantics.

The corresponding terminology to the schema S in Example 69 is the following:

Example 73 (continuation of example 69).
T_S:

$$\begin{aligned}
\mathsf{Employee} &= \overline{\mathsf{Employee}} \sqcap (\geq 1\ \mathsf{salary}) \sqcap \forall \mathsf{salary.Salary} \sqcap \mathsf{Employee} \sqcap \\
&\quad (\leq 1\ \mathsf{salary}) \sqcap (\geq 1\ \mathsf{boss}) \sqcap \forall \mathsf{boss.Manager} \sqcap \mathsf{Employee} \\
\mathsf{Manager} &= \overline{\mathsf{Manager}} \sqcap \mathsf{Employee} \sqcap \forall \mathsf{salary.HighSalary} \\
\mathsf{HighSalary} &= \overline{\mathsf{HighSalary}} \sqcap \mathsf{Salary}
\end{aligned}$$

◇

Before considering S and T_S w.r.t. consistency we need the following

Lemma 74.
Let T be an $\mathcal{ALN}$-terminology, $\mathcal{A}_T$ the corresponding semi-automaton, and A, B atomic concepts in T. Furthermore, let W denote a word where $W \in L_{\mathcal{A}_T}(A, B)$, I a model of T, and $d, e \in dom(I)$ where $dW^I e$ and $d \in A^I$. Then it holds $e \in B^I$.

Proof. Let $A = B_1, U_1, B_2, U_2, \ldots, U_n, B_{n+1} = B$ be a path from A to B labeled with $W = U_1 \cdots U_n$ such that (B_i, U_i, B_{i+1}) are transitions in $\mathcal{A}_T$ for all $1 \leq i \leq n$.

We prove the claim by induction over the length n of the path.
Basis step ($n = 0$): We know $A = B$ and $W = \varepsilon$. Thus, $d = e$ because of $dW^I e$. Now $d \in A^I$ implies $e \in B^I$.

Induction step: Because of $dW^I e$ there is an individual $f \in dom(I)$ such that $dU_1^I f(U_2 \cdots U_{n+1})^I e$. The concept definition of A is $A = \cdots \sqcap \forall U_1.B_2 \sqcap \cdots$. Since I is a model of T, the individual f must be contained in B_2^I because of $d \in A^I$ and $dU_1^I f$. Thus, the induction hypothesis yields $e \in B^I$. □

We now show that consistency/inconsistency of atomic concepts in S is preserved by T_S.

Theorem 75 (consistency).
Let S be an $\mathcal{SLN}$-schema and T_S as defined in Definition 72. For all atomic concepts A in S it holds: A is S-consistent iff A is T_S-consistent w.r.t. gfp-semantics.

Proof. "⇒": Let A be a S-consistent atomic S-concept, i.e., there is a S-model I such that $A^I \neq \emptyset$. We define a gfp-model $\overline{I}$ of T_S such that $A^{\overline{I}} \neq \emptyset$. The interpretation $\overline{I}$ coincide with I on all atomic concepts and roles of S. Furthermore, for the corresponding primitive concept $\overline{B}$ of B in T_S let $\overline{B}^{\overline{I}} := B^I$.
Claim: $\overline{I}$ is gfp-model of T_S where $A^{\overline{I}} \neq \emptyset$.
Proof of the Claim: In order to show that $\overline{I}$ is a model of T_S it is sufficient to show that every S'-concept inclusion $B \sqsubseteq C$ (see page 62 for the definition of S') is satisfied w.r.t. $\overline{I}$ (*). We first verify that this is indeed sufficient: For the concept definition $B = \overline{B} \sqcap C_1 \sqcap \cdots \sqcap C_n$ of B in T_S, where $B \sqsubseteq C_i$, $1 \leq i \leq n$, are the S'-concept inclusions of B, it holds $(\overline{B} \sqcap C_1 \sqcap \cdots \sqcap C_n)^{\overline{I}} \subseteq B^{\overline{I}}$ since $\overline{B}^{\overline{I}} = B^{\overline{I}}$. Conversely by (*), we have $B^{\overline{I}} \subseteq (\overline{B} \sqcap C_1 \sqcap \cdots \sqcap C_n)^{\overline{I}}$ because of $\overline{B}^{\overline{I}} = B^{\overline{I}}$ and $B^{\overline{I}} \subseteq C_i^{\overline{I}}$.

We prove (*): Let $B \sqsubseteq C$ be a S'-concept inclusion. If $B \sqsubseteq C$ is a S-concept inclusion, then by the Definition of $\overline{I}$ and since I is a S-model it follows $B^{\overline{I}} \subseteq C^{\overline{I}}$. On the other hand, if $B \sqsubseteq C$ is no concept inclusion of S, then $C = (\geq n\ R) \sqcap \forall R.C_2 \sqcap C_1$, $n \geq 1$. Thus, S contains the inclusion axioms $B \sqsubseteq (\geq n\ R)$ and $R \sqsubseteq C_1 \times C_2$. Because of $R^{\overline{I}} = R^I$, $C_1^{\overline{I}} = C_1^I$, and $C_2^{\overline{I}} = C_2^I$ we know $B^{\overline{I}} \subseteq (\geq n\ R)^{\overline{I}}$ and $R^{\overline{I}} \subseteq C_1{}^{\overline{I}} \times C_2{}^{\overline{I}}$. Thus, $(\geq n\ R)^{\overline{I}} \subseteq C_1^{\overline{I}}$ (note: $n \geq 1$) and $(\forall R.C_2)^{\overline{I}} = dom(\overline{I})$. Hence, $B^{\overline{I}} \subseteq C^{\overline{I}}$. This proves that $\overline{I}$ is a model of T_S.

Furthermore, it is to show that $\overline{I}$ is a gfp-model of T_S. For this purpose, let I' be a model of T_S such that the primitive interpretations of I' and $\overline{I}$ coincide. Because of $\overline{B}^{\overline{I}} = \overline{B}^{I'}$, $B^{\overline{I}} = \overline{B}^{\overline{I}}$ and $B^{I'} \subseteq \overline{B}^{I'}$ (since I' satisfies $B = \overline{B} \sqcap \cdots$) it follows $B^{I'} \subseteq B^{\overline{I}}$ for all defined concepts B. Hence, $\overline{I}$ is a gfp-model of T_S.

Finally, because of $A^I \neq \emptyset$ and $A^{\overline{I}} = A^I$ we have $A^{\overline{I}} \neq \emptyset$. Thus, A is T_S-consistent w.r.t. the gfp-semantics.

"⇐": Let A be an atomic concept in S which is consistent w.r.t. T_S and the gfp-semantics. We show the existence of a model I of S such that $A^I \neq \emptyset$. Since A is consistent w.r.t. T_S, by Remark 26 there exists the canonical gfp-model $I = I(A, d_0)$ for A and the individual d_0 such that $d_0 \in A^I$. We denote the corresponding primitive interpretation by $J = J(A, d_0)$.

Claim: I is a model of S.
Proof of the Claim: Let $B \sqsubseteq C$ ($C \neq (\geq 0\ R)$) be a concept inclusion in S. The concept definition of B in T_S is of the form $B = \overline{B} \sqcap \cdots \sqcap C \sqcap \cdots$. Since I satisfies this concept definition, it follows $B^I \subseteq C^I$. If $C = (\geq 0\ R)$, then because of $C^I = dom(I)$ it also holds $B^I \subseteq C^I$. Thus, I satisfies all S-concept inclusions.

Now let $R \sqsubseteq C_1 \times C_2$ be a role inclusion in S and $(d,e) \in R^I$ for individuals d, e. It is to show $(d,e) \in {C_1}^I \times {C_2}^I$. According to 25, 1.) for d there is an unique word W required by A such that $d_0 W^J d$. By Lemma 25, 1.) we also have that WR is required by A because of $d_0 W^J d R^J e$; in particular $W \in L_{\mathcal{A}_{T_S}}(A, (\geq n\ R))$ for one $n \geq 1$. Following the definition of T_S the maximum-restriction $(\geq n\ R)$ occurs in a concept definition in the form $\cdots \sqcap (\geq n\ R) \sqcap \cdots$. Thus, there is a defined concept C in T_S such that $A, W, C, \varepsilon, (\geq n\ R)$ is a path in $\mathcal{A}_{T_S}$ labeled with W. The concept definition of C is of the form $C = \overline{C} \sqcap \cdots \sqcap (\geq n\ R) \sqcap \cdots$. By the definition of T_S this implies that $C \sqsubseteq (\geq n\ R)$ is a S-concept inclusion. Since $R \sqsubseteq C_1 \times C_2$ is a S-role inclusion, more precisely, we have for the concept definition of C: $C = \overline{C} \sqcap \cdots \sqcap (\geq n\ R) \sqcap \forall R.C_2 \sqcap C_1 \sqcap \cdots$. Lemma 74 implies $d \in C_1^I$ because of $d_0 W^I d$ and $W \in L_{\mathcal{A}_{T_S}}(A, C_1)$. Furthermore, using $d_0 (WR)^I e$ and $WR \in L_{\mathcal{A}_{T_S}}(A, C_2)$ Lemma 74 implies $e \in C_2^I$. Thus, $(d,e) \in {C_1}^I \times {C_2}^I$. With that it has been shown that I satisfies the S-role inclusions. Hence, I is model of S, which proves the claim.

Due to $d_0 \in A^I$ the claim implies the consistency of A w.r.t. S. □

Since the gfp- and descriptive semantics coincide w.r.t. consistency, Theorem 75 also holds for the descriptive semantics.

As an immediate consequence of Proposition 70 and the above Theorem we have

Corollary 76 (validity).
Using the denotation of Theorem 75 it holds: S is valid iff all atomic S-concepts are consistent w.r.t. T_S and the descriptive (gfp-semantics). □

In chapter 4.3 it has been shown that inconsistency, and therefore consistency, is decidable using polynomial space. Thus, Theorem 75 and Corollary 76 imply that both consistency and validity for $\mathcal{SLN}$-($\mathcal{ALN}$, $\mathcal{SL}_{dis}$-)schemas are also decidable using polynomial space.

It remains to show that T_S and S coincide w.r.t. subsumption.

Theorem 77 (subsumption).
Let S be an $\mathcal{SLN}$-schema and T_S the corresponding $\mathcal{ALN}$-terminology. Then for all atomic concepts A, B in S it holds: $A \sqsubseteq_S B$ iff $A \sqsubseteq_{T_S} B$ (iff $A \sqsubseteq_{gfp,T_S} B$).

Proof. "$\Leftarrow$": If $A \not\sqsubseteq_S B$, then there is a model I of S and an individual d such that $d \in A^I \setminus B^I$. Let $\overline{I}$ be defined as in the proof of Theorem 75 ("$\Leftarrow$"). With that $\overline{I}$ is a (gfp-)model of T_S in which the atomic concepts of S are interpreted as in I. Thus, $d \in A^{\overline{I}} \setminus B^{\overline{I}}$, and hence $A \not\sqsubseteq_{T_S} B$ (resp., $A \not\sqsubseteq_{gfp,T_S} B$).

"$\Rightarrow$": If $A \not\sqsubseteq_{T_S} B$ (resp., $A \not\sqsubseteq_{gfp,T_S} B$), then there is a (gfp-)model I of T_S and an individual d such that $d \in A^I \setminus B^I$. We define an S-model I' such that $d \in A^{I'} \setminus B^{I'}$

holds, and thus $A \not\sqsubseteq_S B$:
$dom(I') := dom(I)$; $C^{I'} := C^I$ for all atomic concepts C in S; for given s-roles T and all S-role inclusions $R \sqsubseteq C_1^1 \times C_2^1, \dots, R \sqsubseteq C_1^n \times C_2^n$ of R in S we define $R^{I'} := R^I \cap ((C_1^1)^I \times (C_2^1)^I) \cap \dots \cap ((C_1^n)^I \times (C_2^n)^I)$ (for $n = 0$ let $R^{I'} = R^I$).

By definition of I' the role inclusions of S are satisfied I'.

For an S-concept inclusion $C \sqsubseteq D$ (resp., $C \sqsubseteq \neg D$) where C, D are atomic concepts the concept definition of C in T_S is of the form $C = \overline{C} \sqcap \dots \sqcap D \sqcap \cdots$ (resp., $C = \overline{C} \sqcap \dots \sqcap \neg D \sqcap \cdots$). Since I is a T_S-model, $C^{I'} = C^I$, and $D^{I'} = D^I$, we have $C^{I'} \subseteq D^{I'}$ (and because of $dom(I') = dom(I)$ also $C^{I'} \subseteq (\neg D)^{I'}$). Thus, the concept inclusion $C \sqsubseteq D$ (resp., $C \sqsubseteq \neg D$) is satisfied w.r.t. I'.

If $C \sqsubseteq \forall R.D$ is an S-concept inclusion, then due to $R^{I'} \subseteq R^I$ and $D^{I'} = D^I$ it holds $(\forall R.D)^I \subseteq (\forall R.D)^{I'}$. Furthermore, the concept inclusion of C in T_S is of the form $C = \overline{C} \sqcap \dots \sqcap \forall R.D \sqcap \cdots$. Consequently, it follows $C^I \subseteq (\forall R.D)^I$. Using $C^{I'} = C^I$ this implies $C^{I'} \subseteq (\forall R.D)^{I'}$. Analogously it can be shown that I' satisfies inclusions of the form $C \sqsubseteq (\leq n\ R)$ in S because of $R^{I'} \subseteq R^I$.

A concept inclusion $C \sqsubseteq (\geq 0\ R)$ in S is trivially satisfied by I' since $(\geq 0\ R)^{I'} = dom(I')$.

If $C \sqsubseteq (\geq n\ R)$, $n \geq 1$, is an S-concept inclusion and $R \sqsubseteq C_1^1 \times C_2^1, \dots, R \sqsubseteq C_1^m \times C_2^m$ are the S-role inclusions for R, then the concept inclusion for C in T_S is of the form $C = \overline{C} \sqcap \dots \sqcap (\geq n\ R) \sqcap \forall R.C_2^1 \sqcap \dots \sqcap \forall R.C_2^m \sqcap C_1^1 \sqcap \dots \sqcap C_1^m \sqcap \cdots$. Since I is a (gfp-)model of T_S, it holds for an individual $d \in C^I$ that d has at least n distinct R-successors $e_1, \dots, e_n$ for which holds $e_1, \dots, e_n \in (C_2^1 \sqcap \dots \sqcap C_2^m)^I$. Furthermore, $d \in (C_1^1 \sqcap \dots \sqcap C_1^m)^I$. Thus, by definition of $R^{I'}$ the individuals $e_1, \dots, e_n$ are R-successors of d w.r.t. I' . Hence, $d \in (\geq n\ R)^{I'}$. Consequently, I' satisfies the concept inclusion $C \sqsubseteq (\geq n\ R)$.

Hence we have shown that I' is a model of S. Additionally, $A^{I'} = A^I$ and $B^{I'} = B^I$ imply $d \in A^{I'} \setminus B^{I'}$, and thus $A \not\sqsubseteq_S B$. □

By the Corollaries 41 and 50 this theorem provides us with PSPACE decision algorithms for subsumption w.r.t. $\mathcal{SLN}$- (resp., $\mathcal{ALN}$-)schemas. This Theorem implies using the Corollaries 41 and 50 that subsumption for $\mathcal{SLN}$- (resp.¡ $\mathcal{ALN}$-, $\mathcal{SL}_{dis}$)schemas is decidable using polynomial space.

In [7] it has been shown that consistency and subsumption involving $\mathcal{SL}_{dis}$-schemas is decidable using polynomial space by an algorithm working on special graphs.

The results presented here not only extends the upper bound complexity results for $\mathcal{SL}_{dis}$-schemas to $\mathcal{SLN}$-schemas, but also uses well-known results and techniques in automata theory, rather than defining special graphs as in [7]. Furthermore, it clarifies the relationship between schemas and terminologies for the considered languages: $\mathcal{ALN}$-terminologies are at least as expressive as $\mathcal{SLN}$-schemas w.r.t. the important inference problems.

Finally, it should also be noted that the complexity of subsumption w.r.t. schemas is only due to testing consistency of concepts. By [7], Proposition 4.15 it holds:[1]

> Let S be an $\mathcal{SL}_{dis}$-schema and $A, A_1, \ldots, A_m$ atomic concepts where $A_1 \sqcap \cdots \sqcap A_m$ is S-consistent. Then it holds:
> $A_1 \sqcap \cdots \sqcap A_m \sqsubseteq_S A$ iff $A \in \varepsilon\text{-}closure_{\mathcal{A}_{T_S}}(\{A_1, \ldots, A_m\})$

Hence, disallowing concept forming operators that enable the definition of inconsistent concepts (number restrictions and primitive negation) makes reasoning for schemas tractable, whereas for $\mathcal{FL}_0$-terminologies subsumption is already PSPACE-complete (w.r.t. fixed-point semantics). Furthermore, (weak-)acyclic terminologies and schemas also differ in terms of complexity.

7.3 (Weak-)acyclic terminologies and schemas

Definition 78 (weak-acyclic).
An $\mathcal{ALN}$-terminology T is called *weak-acyclic* iff for all non-empty words $W \in \Sigma^+$ and all (defined) concepts A in T it holds: $W \notin L_{\mathcal{A}_T}(A, A)$.

A schema S is called *weak-acyclic* if the corresponding terminology T_S is weak-acyclic. ◇

The following Theorem shows that consistency for weak-acyclic terminologies is "easier" to decide than for general terminologies.

Theorem 79.
Consistency (w.r.t. gfp-, lfp-, and descriptive semantics) for (weak-)acyclic $\mathcal{ALN}$- ($\mathcal{FLN}$-)terminologies is co-NP-complete.

Proof. We first show the upper bound. Let T be an $\mathcal{ALN}$- ($\mathcal{FLN}$-)terminology, $\mathcal{A}_T$ the corresponding semi-automaton without word-transitions (see Remark 18), and A an atomic concept in T. By Theorem 29 and Theorem 57, respectively, A is consistent iff the set $\varepsilon\text{-}closure_{\mathcal{A}_T}(\{A\})$ is an exclusion set. Thus, to show the claim of the theorem it is sufficient to prove the existence of a NP-algorithm for deciding exclusion sets w.r.t. weak-acyclic terminologies. In fact, in the algorithms 28 and 56 $n+1$, $n = |Q|$, iterations of the while-loop are sufficient. Proof:
Let $F \subseteq Q$. For $F_0 := F$ and $F_i := next_\varepsilon(F_{i-1}, R_i)$, $R_i \in \Sigma$ (arbitrary), $i \geq 1$, it holds: $F_{n+1} = \emptyset$. If $F_{n+1} \neq \emptyset$, then there are defined concepts $A_0, \ldots, A_n$ as well as an atomic concept, a number-restriction, or a primitive negation A_{n+1} such that $A_0, R_1, A_1, R_2, \ldots, R_{n+1}, A_{n+1}$ is a path in $\mathcal{A}_T$. Since $\mathcal{A}_T$ has only n states, there are numbers i, j where $0 \leq i < j \leq n$ and $A_i = A_j$. This implies $\varepsilon \neq R_{i+1} \cdots R_j \in L(A_i, A_j)$ which contradicts the assumption that T is weak-acyclic.

Thus in the algorithms 28 and 56 the while-condition $z < 2^{|Q|}$ can be substituted by $z < |Q| + 1$ since for $z = |Q| + 1$ the variable F is the empty set. Furthermore, in algorithm 56 the output in (2) has to be modified from "yes" to "no" since an

[1] This statement is adapted to the automata theoretic characterization used in the present paper.

empty set of states can not be an exclusion set in terms of Definition 55, 1.). This completes the prove of the upper bound complexity.

Now we prove the lower bound complexity. In [7], validity of $\mathcal{SL}_{dis}$-schemas has been reduced to consistency of $\mathcal{ALE}$-concepts which is co-NP-complete [11]. More precisely, for a $\mathcal{ALE}$-concept C an $\mathcal{SL}_{dis}$-schema S_C has been defined such that

$$C \text{ is consistent iff } S_C \text{ is valid.} \tag{7.1}$$

The schema S_C as defined in [7] contains an atomic concept A_C. The proof of (7.1) reveals that also the following statement holds:

$$C \text{ is consistent iff } A_C \text{ is consistent w.r.t. } S_C. \tag{7.2}$$

Moreover, the schema S_C is acyclic. Thus, also T_{S_C} is acyclic such that the three semantics coincide. Hence, Theorem 75 implies the co-NP-hardness of consistency for (weak-)acyclic $\mathcal{ALN}$-($\mathcal{FLN}$-)terminologies w.r.t. all three semantics. □

The proof of the above theorem also shows that consistency and validity (see Proposition 70) for (weak-)acyclic $\mathcal{SLN}$- ($\mathcal{ALN}$-)schemas is co-NP-complete.

As for general terminologies and schemas, even for acyclic terminologies and schemas there are differences in terms of complexity for subsumption, e.g., subsumption for acyclic $\mathcal{FL}_0$-terminologies is co-NP-complete [19], whereas it is NP-complete for acyclic $\mathcal{SL}_{dis}$-schemas [7].

Chapter 8

Conclusion

In several examples we have seen that cyclic terminologies are a natural way to describe the terminological knowledge of a problem domain. Unlike first-order predicate logic, the transitive closure of relations is expressible using the gfp-semantics. In [2], description logics have been explicitly extended by role constructors union, composition, and transitive closure. Nevertheless, as already pointed out cyclic definitions occur frequently, e.g., in constraints for semantic and object-oriented data models. The previous chapter has shown that the here considered general $\mathcal{ALN}$-terminologies are expressive enough to capture important parts of such constraints.

The examples considered in this work as well as the literature reveal that the descriptive semantics is not suitable in every representation task to capture the intuition of a cyclic terminology. On the other hand, also the introduced fixed-point semantics do not fit in any case. Therefore, it is crucial to gain a more profound understanding of the intuition of these semantics, in order to make decision easier which semantics is to prefer in the representation task at hand. For this reason, automata theoretic characterizations have been provided for the three semantics with respect to $\mathcal{ALN}$. The characterization of the three semantics itself was an easy extension of the characterization for $\mathcal{FL}_0$ [4].

Using this characterizations we have also proven characterizations for the important inference problems inconsistency and subsumption for $\mathcal{FLN}$ ($\mathcal{ALN}$). Due to primitive negation and conflicting number-restrictions inconsistent concepts can occur for all three semantics—unlike $\mathcal{FL}_0$, where inconsistent concepts are only expressible for the lfp-semantics. From the characterizations of inconsistency we have derived decision algorithms and complexity results (see table 1.1) using exclusion sets, which where defined by semi-automata.

Due to inconsistency a straightforward extension of the characterization of subsumption for $\mathcal{FL}_0$ to $\mathcal{FLN}$ is not possible. Rather the notion "exclusion words" was needed, which also has been defined in terms of the corresponding semi-automaton of a terminology. We have described the set of exclusion words with the help of exclusion sets. Again, using this sets we were able to formulate decision algorithms and to show complexity results for subsumption (see table 1.1). It has turned out that subsumption for $\mathcal{ALN}$ is less complex (namely, PSPACE-complete instead of

EXPTIME-complete) than subsumption for much more expressive languages proposed in the literature which allow for fixed-points operators.

Finally, we have shown that the important inference problems for $\mathcal{SLN}$- and $\mathcal{SL}_{dis}$-schemas can be reduced to corresponding problems for terminologies. These schemas are expressive enough to describe constraints frequently occurring in semantic and object-oriented data models. This reduction has provided us with automata theoretic decision algorithms for such problems and clarifies the expressive power of the considered schemas and terminologies; in addition, it extends the results for $\mathcal{SL}_{dis}$-schemas [7] to $\mathcal{SLN}$-schemas, which allow for arbitrary number-restrictions. We have also pointed out the difference between schemas and terminologies. The complexity of subsumption in the here considered schemas is only due to inconsistency, whereas subsumption w.r.t. terminologies is already complex for $\mathcal{FL}_0$-terminologies.

Bibliography

[1] H. Aït-Kaci. *A Lattice-Theoretic Approach to Computations Based on a Calculus of Partially Ordered Type Structures.* PhD thesis, University of Pennsylvania, PA, 1984.

[2] F. Baader. Augmenting concept languages by transitive closure of rules: An alternativ to terminological cycles. In *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, JCAI-91, Sydney, 1991.

[3] F. Baader. A formal definition for the expressive power of terminological knowledge representation languages. *J. of Logic and Computation*, 6(1):33–54, 1996.

[4] F. Baader. Using automata theory for characterizing the semantics of terminological cycles. *Annals of Mathematics and Artificial Intelligence*, 18(2–4):175–219, 1996.

[5] S. Bergamaschi and C. Sartori. On taxonomic reasoning in conceptual design. *ACM Trans. on Database Systems*, 17(3):385–442, 1992.

[6] A. Borgida. Description logics in data management. *IEEE Trans. on Knowledge and Data Engineering*, 7(5):671–682, 1995.

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

[8] D. Calvanese. Reasoning with inclusion axioms in description logics: Algorithms and Complexity. In W. Wahlster, editor, *Proceedings of the 12th European Conference on Artificial Intelligence, ECAI-96.* John Wiley & Sons, 1996.

[9] D. Calvanese. *Unrestricted and Finite Model Reasoning in Class-Based Representation Formalisms.* PhD thesis, Dip. di Inf. e Sist., Univ. di Roma "La Sapienza", 1996.

[10] Diego Calvanese, Maurizio Lenzerini, and Daniele Nardi. A unified framework for class based representation formalisms. In *Proceedings of the Fourth International Conference on the Principles of Knowledge Representation and Reasoning (KR-94)*, pages 109–120. Morgan Kaufmann, Los Altos, 1994.

[11] F.M. Donini, B. Hollunder, M. Lenzerini, A. Marchetti, D. Nardi, and W. Nutt. The complexity of existential quantification in concept languages. *Artificial Intelligence*, 2–3:309–327, 1992.

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

[13] G. De Giacomo and M. Lenzerini. Concept languages with number restrictions and fixpoints, and its relationship with mu-calculus. In *Proceedings of the Eleventh European Conference on Artificial Intelligence, ECAI-94*, pages 411–415. John Wiley and Sons, 1994.

[14] J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory.* Addison-Wesley Publ. Co., 1979.

[15] J.W. Lloyd. *Logic Programming.* Springer Verlag, Berlin, second edition, 1987.

[16] R. MacGregor. What's needed to make a description logic a good KR citizen. In *Working Notes of the AAAI Fall Symposium on Issues on Description Logics: Users meet Developers*, pages 53–55, 1992.

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

[18] B. Nebel. Reasoning and revision in hybrid representation systems. In *Lecture Notes in Computer Science 422*. Springer Verlag, Berlin, 1990.

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

[20] B. Nebel. Terminological cycles: Semantics and computational properties. In J. Sowa, editor, *Formal Aspects of Semantic Networks*, pages 331–361. Morgan Kaufmann, San Mateo, 1991.

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

[22] K. Schild. Terminological cycles and the propositional μ-calculus. In *Proceedings of the Fourth International Conference on Principles of Knowledge Representation and Reasoning, KR'94*, pages 509–520, Bonn, Germany, 1994. Morgan Kaufmann.