Charakterisierung der Semantik terminologischer Zyklen mit Hilfe endlicher Automaten

Diplomarbeit
im Fach Informatik
am
Lehr– und Forschungsgebiet Theoretische Informatik
der
Rheinisch–Westfälischen Technischen Hochschule Aachen
Prof. Dr.–Ing. F. Baader

vorgelegt von

Ralf Küsters
Matrikelnummer 191278

betreut durch

Universitätsprofessor Dr.–Ing. Franz Baader

Aachen, im Mai 1997

Erklärung

Hiermit versichere ich, daß ich die vorliegende Arbeit selbständig verfaßt und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Aachen, im Mai 1997 ....

Danksagung

Es ist mir ein Anliegen, all denjenigen dankzusagen, die mich bei dieser Arbeit und auf meinem bisherigen Weg begleitet und geleitet haben. Besonders möchte ich Herrn Prof. Dr. Franz Baader danken, der stets für mich Zeit gefunden hat — auch wenn er keine hatte —, und der mich auch über diese Arbeit hinaus mit viel Engagement unterstützt und fördert. Diesbezüglich gilt auch Herrn Prof. Dr. Klaus Indermark mein aufrichtiger Dank. Mit Rat und Tat stand mir darüber hinaus Dipl.-Inform. Can Adam Albayrak zur Seite, der mich zusammen mit Dipl.-Inform. Jörn Richts zudem in TeXnischen Fragen beraten hat. Tiefe Dankbarkeit gilt meinen Eltern, die mich immer geformt, nie aber verbogen haben. Kathrin Lückgen war vor meinen Prüfungen mindestens so nervös wie ich selbst: Danke dafür! Es darf in diesem Zusammenhang, der verständlicherweise nicht allen einleuchten wird, folgendes Zitat nicht fehlen:

> „Handle so, daß du die Menschheit, sowohl in deiner Person, als in der Person eines jeden andern, jederzeit zugleich als Zweck, niemals bloß als Mittel brauchest.“ (Immanuel Kant; kategorischer Imperativ, formuliert in der Selbstzweckformel)

Schließlich danke ich Ralf Molitor, Jörg (Proto-)Köller, Marcus Raddatz, (Super-)Ramin Sadre, Frank Steiner und Martin Hönigs sowie Martin Küsters, Magnus Dollen, Silke Heydhausen und Anita Gubbels, denen ich eine sehr schöne und aufbauende Studienzeit verdanke, und ohne die mir mein Studium nur höchstens halb soviel Spaß gemacht hätte.

Ralf Küsters

Inhaltsverzeichnis

Kapitel 1

Einleitung

Terminologische Wissensrepräsentationssysteme erlauben die Beschreibung von Konzepten und Instanzen einer Anwendung. Konzepte werden dabei durch Konzeptterme definiert, die aus atomaren Konzepten und Rollen sowie verschiedenen Konzeptkonstruktoren aufgebaut sind. Die verschiedenen Wissensrepräsentationssprachen unterscheiden sich dabei durch Anzahl und Art der Konstruktoren.

Das terminologische Wissen wird in einer *Terminologie* (T-Box) durch eine endliche Menge von Axiomen der Form $C = D$ beschrieben. Dabei bezeichnet C ein Konzept und D einen Konzeptterm. Falls zu C ein Konzept in der T-Box existiert, heißt C *definiertes Konzept*; ansonsten nennen wir C *primitives Konzept*. Zu einem definierten Konzept existiert in einer T-Box immer genau ein Axiom.

Die Menge $\{\mathsf{Frau} = \mathsf{Mensch} \sqcap \neg\, \mathsf{männlich}, \mathsf{Mutter} = \mathsf{Frau} \sqcap \exists^{\geq 1}\mathsf{kind}\}$ ist eine aus zwei Axiomen bestehende Terminologie. In dieser T-Box wird das definierte Konzept Frau als ein Mensch definiert, der nicht männlich ist. Eine Mutter ist eine Frau, die mindestens ein Kind hat.

Formal wird die Semantik — wie in der Prädikatenlogik — in einem modell-theoretischen Ansatz definiert. Dazu definiert man eine *Interpretation*, die aus einer Grundmenge (Menge von Individuen/Objekten) besteht und die jedem Konzept eine Teilmenge der Grundmenge sowie jeder Rolle eine binäre Relation über der Grundmenge zuordnet. Konzepte und Rollen entsprechen in der Prädikatenlogik somit einstelligen bzw. zweistelligen Prädikaten. Die Interpretation wird auf Konzeptterme erweitert, so daß diese, wie Konzepte, als Teilmengen der Grundmenge interpretiert werden. Z.B. wird der Konzeptterm $\exists^{\geq 1}\mathsf{kind}$ als Menge der Individuen interpretiert, die bzgl. der zu kind gehörenden binären Relation mindestens einen direkten Rollennachfolger, d.h. mindestens einen kind-Rollennachfolger, besitzen. Eine Interpretation ist ein *Modell* einer T-Box, wenn alle Axiome der T-Box bzgl. dieser Interpretation erfüllt sind. Ein Axiom wird von einer Interpretation erfüllt, falls linke und rechte Seite des Axioms bzgl. der Interpretation die selben Mengen beschreiben. Die sogenannte *deskriptive Semantik* einer Terminologie ist bestimmt durch die Menge aller Modelle dieser Terminologie.

Man erwartet, daß durch die Interpretation der primitiven Konzepte und Rollen, d.h. durch die sogenannte *primitive Interpretation*, die Interpretation der definierten Konzepte festgelegt ist; die primitive Interpretation soll also eindeutig zu einem Modell der Terminologie fortgesetzt werden können. Dies ist für azyklische Terminologien, d.h. solche, bei denen kein definiertes Konzept (direkt oder indirekt) durch sich selbst definiert wird, der Fall. Durch „Auffalten" der Terminologie erhält man nämlich eine bzgl. der deskriptiven

Semantik äquivalente Terminologie, in der jedes Axiom auf der rechten Seite neben Rollen nur noch primitive Konzepte enthält [Neb90a]. In azyklischen Terminologien sind deshalb definierte Konzepte nur Abkürzungen für Konzeptterme, die aus *primitiven* Konzepten und Rollen aufgebaut sind. Jedes definierte Konzept ist insbesondere eindeutig durch primitive Konzepte und Rollen definiert. Für zyklische Terminologien gilt diese Eindeutigkeit nicht. Um jedoch auch für diese die eindeutige Fortsetzung der primitiven Interpretation zu erhalten, werden statt der deskriptiven Semantik sogenannte Fixpunktsemantiken betrachtet.

Die Betrachtung zyklischer Terminologien ist sinnvoll, da die Darstellung terminologischen Wissens in natürlicher Weise terminologische Zyklen aufweisen kann. Ein Mensch kann z.B. als Lebewesen beschrieben werden, welches genau zwei Elternteile hat, die ihrerseits wiederum Menschen sind. Ein Binärbaum besteht aus Elementen (Knoten), welche zu einem Baum gehören und höchstens jeweils zwei Nachfolgerknoten besitzen, die ihrerseits selbst zum Binärbaum gehören müssen[1]. Die Axiome zu den Beispielen können wie folgt formuliert werden:

$$\mathsf{Mensch} = \mathsf{Lebewesen} \sqcap \exists^{\geq 2}\mathsf{eltern} \sqcap \exists^{\leq 2}\mathsf{eltern} \sqcap \forall \mathsf{eltern.Mensch} \tag{1.1}$$

$$\mathsf{Bin\ddot{a}rbaum} = \mathsf{Baum} \sqcap \exists^{\leq 2}\mathsf{direkter\text{-}Nachfolger} \sqcap \forall \mathsf{direkter\text{-}Nachfolger.Bin\ddot{a}rbaum} \tag{1.2}$$

In beiden Fällen werden Aussagen über die transitive Hülle einer Relation formuliert. In (1.1) wird gefordert, daß alle Vorfahren eines Menschen (transitive Hülle von eltern) ebenfalls Menschen sind. Entsprechend beschreibt (1.2) Bedingungen über alle (direkten und indirekten) Nachfolger eines Knotens.

Obwohl also zyklische Beschreibungen natürliche Darstellungen terminologischen Wissens sein können und diese weiterhin — im Gegensatz zur Prädikatenlogik der ersten Stufe (vgl. [AU79]) — Aussagen über transitive Hüllen von Relationen erlauben, können sie in vielen Systemen, z.B. KRYPTON [BPGL85], LOOM [MB87], CLASSIC [PSMB$^+$91] und KRIS [BH91], nicht behandelt werden. In diesen Systemen werden die Terminologien auf Zyklenfreiheit getestet oder Zyklen führen zur Nicht-Terminierung von Entscheidungsalgorithmen für Inferenzprobleme (z.B. Subsumtion).

Erste eingehendere Untersuchungen terminologischer Zyklen finden sich in [Neb87, Neb90a, Neb91]. Dort werden alternativ zur deskriptiven Semantik auch die gfp-Semantik (größter Fixpunkt) und die lfp-Semantik (kleinster Fixpunkt) für die Sprache $\mathcal{NTF}$ (vgl. [Neb90a]) betrachtet. In [Baa96] werden anhand der Sprache $\mathcal{FL}_0$, die nur Konjunktion und Werterestriktion enthält[2], die drei Semantiken mit Hilfe von endlichen Automaten charakterisiert. Dies ermöglicht einen tieferen Einblick und ein eingehenderes Verständnis der verschiedenen Semantiken und liefert Entscheidungsalgorithmen für die Subsumtion von Konzepten. Obwohl auch hier nicht abschließend geklärt wurde, welche Semantik bei zyklischen Terminologien zu bevorzugen ist, zeigt sich für die gfp-Semantik jedoch, daß sie im Vergleich zu den anderen Semantiken „natürlichere“ Charakterisierungen durch endliche Automaten zuläßt. Andererseits gibt es Beispiele, bei denen jeweils mit einer der drei Semantiken die intuitive Bedeutung einer Terminologie besser beschrieben wird als mit den jeweils anderen.

[1] Die Beispiele stammen aus [Neb91] und werden später nochmals aufgegriffen.

[2] Für die deskriptive und die gfp-Semantik wird auch $\mathcal{FL}^-$, d.h. $\mathcal{FL}_0$ mit existentieller Restriktion (vgl. [LB87]), betrachtet.

Da die Sprache $\mathcal{FL}_0$ für viele Repräsentationsaufgaben unzureichend ist, werden in dieser Arbeit Erweiterungen dieser Sprache betrachtet. Dazu wird $\mathcal{FL}_0$ um primitive Negation und um Zahlenrestriktionen zur Sprache $\mathcal{ALN}$ erweitert. Neben der Charakterisierung der drei genannten Semantiken und der Subsumtion ist bei den Erweiterungen auch die Betrachtung der Inkonsistenz von Konzepten interessant. Auch für die erweiterten Sprachen stellt sich heraus, daß die gfp-Semantik eine einfachere Beschreibung durch endliche Automaten zuläßt als die beiden anderen Semantiken. Aus den Charakterisierungen leiten wir Entscheidungsalgorithmen und Komplexitätsaussagen für Inkonsistenz und Subsumtion ab. Die gewonnenen Ergebnisse werden schließlich dazu genutzt, Aussagen über Schemata abzuleiten. Schemata enthalten Konzeptinklusionen — statt Axiome — und zusätzlich Rolleninklusionen. Wir werden Entscheidungsalgorithmen und Komplexitätsaussagen für Konsistenz, (lokale) Gültigkeit sowie Subsumtion bzgl. $\mathcal{ALN}$-Schemata durch Reduktion dieser Probleme auf Konsistenz und Subsumtion bzgl. $\mathcal{ALN}$-Terminologien erhalten. Umgekehrt gewinnen wir aus den in [BDNS97] beobachteten Komplexitätsaussagen zu ($\mathcal{SL}_{dis}$-)Schemata mit Hilfe der erwähnten Reduktion Härte-Resultate für die Konsistenz in $\mathcal{ALN}$-Terminologien.

Im nächsten Kapitel werden zunächst grundlegende Begriffe und Aussagen zu Ordinalzahlen, Fixpunkten sowie endlichen Automaten eingeführt. Ordinalzahlen und Fixpunkte werden später dazu benutzt, die erwähnten Fixpunktsemantiken zu definieren. Im Abschnitt zu endlichen Automaten wird ein (N)PSPACE-Algorithmus zur Entscheidung der Inklusion regulärer Sprachen angegeben, da dieser in ähnlicher Form auch zur Entscheidung der Subsumtion in den folgenden Kapiteln auftritt.

Im dritten Kapitel werden wir Syntax und Semantik der terminologischen Repräsentationssprache $\mathcal{ALN}$ einführen. Hier werden die oben angesprochenen Begriffe „Interpretation" und „Modell" formal definiert.

Das vierte Kapitel führt schließlich den Begriff der terminologischen Zyklen formal ein. Es wird das Beispiel des Binärbaumes aufgegriffen, welches die Einführung der Fixpunktsemantiken motiviert. Schon an dieser Stelle wird sich die gfp-Semantik als die natürlichere herausstellen.

Die Kapitel fünf und sechs bilden den Kern der Arbeit. Sie enthalten die Charakterisierung der erweiterten Sprachen mit Hilfe endlicher Automaten. Im fünften Kapitel werden wir für die Sprache $\mathcal{AL}_0$, d.h. $\mathcal{FL}_0$ mit primitiver Negation, die Semantiken, die Inkonsistenz sowie die Subsumtion mit endlichen Automaten charakterisieren. Aufgrund der primitiven Negation können auch bzgl. der gfp- und der deskriptiven Semantik Konzepte inkonsistent sein — ohne primitive Negation treten nur für die lfp-Semantik inkonsistente Konzepte auf. Die primitive Negation ist auch dafür verantwortlich, daß Konzepte von Wörtern „ausgeschlossen" werden können. Diese Wörter verhindern, daß bestimmte Individuen zur Extension eines Konzeptes gehören können; sie spielen bei der Charakterisierung der Subsumtion eine wichtige Rolle. Für die algorithmische Behandlung der Inkonsistenz und der Subsumtion wird der Begriff des „Ausschlußzustandes" wesentlich sein. Wir werden die Mengen der Wörter, die ein Konzept ausschließen, mit Hilfe von Ausschlußzuständen charakterisieren. Dies erlaubt es (NPSPACE-)Entscheidungsalgorithmen für die Inkonsistenz und die Subsumtion von Konzepten anzugeben. Für die Subsumtion werden — in Anlehnung an die Verfahren in [Baa96] — auch Entscheidungsalgorithmen angeben, die sich ausschließlich auf die Entscheidung der Inklusion von regulären und ω-regulären Sprachen stützen. Das Problem bei diesen Algorithmen ist jedoch, daß die Behandlung von Zahlenrestriktionen hier nicht so einfach möglich ist. Dagegen lassen sich die auf Ausschlußzustände basierenden Entscheidungsalgorithmen leichter auf Zahlenrestriktionen erweitern. Es sind dazu im wesentlichen nur die Definitionen der Ausschlußzustände zu modifizieren.

Kapitel sechs behandelt die Erweiterung von $\mathcal{AL}_0$ durch Zahlenrestriktionen, d.h. die Sprache $\mathcal{ALN}$. Wir werden sehen, daß bei Anwesenheit von Zahlenrestriktionen auf die primitive Negation — sogar auf primitive Konzepte überhaupt — verzichtet werden kann. Weiter führen Zahlenrestriktionen (wie z.B. $\exists^{\geq 2}\mathsf{eltern}$) dazu, daß für Individuen Nachfolger „gefordert" werden können. Dies macht die Handhabung von Modellen dieser Terminologien aufwendiger und erfordert, wie bereits erwähnt, die Anpassung der im fünften Kapitel eingeführten zentralen Begriffe „Ausschluß" und „Ausschlußzustand".

Im siebten Kapitel wird der angesprochene Zusammenhang zwischen Terminologien und Schemata behandelt. Neben zyklischen Terminologien und Schemata werden hier auch (schwach-)azyklische Terminologien und Schemata untersucht.

Wir werden schließlich die Ergebnisse zusammenfassen und kurz auf verwandte Arbeiten eingehen.

Kapitel 2

Vorbereitung

In diesem Kapitel werden grundlegende Begriffe zu Ordinalzahlen, Fixpunkten und endlichen Automaten eingeführt. Außerdem werden dazu — vor allem zu endlichen Automaten — einige Aussagen formuliert, auf die in den folgenden Kapiteln zurückgegriffen wird.

2.1 Ordinalzahlen und Fixpunkte

Wir gehen im folgenden auf Begriffe und Aussagen zu Ordinalzahlen und Fixpunkten ein. Diese können auch in [Ros82] bzw. [Llo87] nachgelesen werden.

Eine *lineare Ordnung* $<$ über der Grundmenge D ist eine zweistellige, transitive, totale[1] und irreflexive Relation über D. Sie heißt *wohl-geordnet*, wenn zusätzlich jede nicht-leere Teilordnung[2] von $<$ ein kleinstes Element enthält. Ein *Ordnungstyp* ist eine Äquivalenzklasse von isomorphen, linearen Ordnungen. *Ordinalzahlen* bezeichnen Ordnungstypen von isomophen, wohl-geordneten Ordnungen. Man kann auf der Menge der Ordinalzahlen eine wohl-geordnete Ordnung definieren. Dazu sei die Ordnung $<$ auf dieser Menge wie folgt definiert: $\alpha < \beta$ gdw. die Ordinalzahl α isomorph zu einem echten initialen Segment der Ordinalzahl β ist. Da dies eine wohl-geordnete Ordnung darstellt, besitzt damit insbesondere jede Menge von Ordinalzahlen ein kleinstes Element sowie eine kleinste obere Schranke (d.h. kleinstes Element der oberen Schranken) und enthält keine unendlich absteigende Kette.

2, 17, 42 sind Beispiele für endliche Ordinalzahlen. Dabei ist 17 der Ordnungstyp der Menge $\{0, 1, 2, \ldots, 16\}$ mit der üblichen Kleiner-Ordnung auf natürlichen Zahlen. Der kleinste Ordnungstyp mit unendlicher Anzahl von Elementen ist ω, d.h. die Menge $\{0, 1, 2, \ldots\}$ mit der üblichen Kleiner-Ordnung auf natürlichen Zahlen. Es ist $n < \omega$ für alle endlichen Ordinalzahlen n.

Ist α eine Ordinalzahl, dann heißt $\alpha + 1$ *Nachfolger* von α. Eine Ordinalzahl, die Nachfolger einer (anderen) Ordinalzahl ist, heißt *Nachfolgerordinal*; alle anderen heißen *Grenzordinale*. Wegen $17 = 16 + 1$ ist z.B. 17 ein Nachfolgerordinal. Dagegen ist ω kein Nachfolgerordinal, jedoch $\omega + 1$ mit Ordnungstyp $\{0, 1, 2, \ldots\} \cup \{\infty\}$, wobei alle Elemente aus $\{0, 1, 2, \ldots\}$ kleiner als ∞ sind und $\{0, 1, 2, \ldots\}$ wie oben geordnet ist. Ein Grenzordinal α kann als kleinste obere Schranke (*lub*) aller kleineren Ordinalzahlen erhalten werden,

[1] Für alle $a, b \in D, a \neq b$, gilt $a < b$ oder $b < a$.

[2] Eine Relation ist Teilordnung von $<$, falls sie Durchschnitt von $<$ und $S \times S$ für eine Teilmenge S von D ist.

d.h. $\alpha = lub(\{\beta; \beta < \alpha\})$.

Nun wenden wir uns den Grundlagen von Fixpunkten zu.

Eine partielle Ordnung[3] D ist ein *vollständiger Verband*[4], falls jede Teilordnung C von D eine *kleinste obere Schranke* $lub(C)$ in D besitzt. Wegen $glb(C) = lub(\{d \in D; d$ ist untere Schranke von $C\})$ existiert zu C dann auch die *größte untere Schranke* (glb). Damit existieren auch das *kleinste Element*, $bottom = lub(\emptyset)$, und das *größte Element*, $top = glb(\emptyset)$, von D. Auf folgendes Beispiel werden wir später bei der Definition der Fixpunktsemantiken zurückgreifen:

Beispiel 2.1.
Es sei $D = 2^S \times \cdots \times 2^S$ das n-fache kartesische Produkt über der Potenzmenge von S. Mit der komponentenweisen Inklusion $((A_1, \ldots, A_n) \subseteq (B_1, \ldots, B_n)$ gdw. $A_1 \subseteq B_1, \ldots, A_n \subseteq B_n)$ ist D ein vollständiger Verband; $top = (S, \ldots, S)$ und $bottom = (\emptyset, \ldots, \emptyset)$. ◇

Ist D eine partielle Ordnung und $T : D \longrightarrow D$ eine Abbildung, so ist T *monoton* gdw. $T(a) \leq T(b)$ für alle $a, b \in D$ mit $a \leq b$ gilt. Für $f \in D$ und $T(f) = f$ ist f *Fixpunkt* von T. Ist D ein vollständiger Verband und $T : D \longrightarrow D$ monoton, so besitzt T einen *kleinsten Fixpunkt* $lfp(T)$ und einen *größten Fixpunkt* $gfp(T)$, die identisch sein können. Alle anderen Fixpunkte liegen dazwischen (vgl. [Ros82], Proposition 5.1). Der kleinste und der größte Fixpunkt lassen sich durch Ordinalpotenzen von T charakterisieren. Die Ordinalpotenzen $T{\uparrow}^{\alpha}$ und $T{\downarrow}^{\alpha}$ sind induktiv definiert durch:

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

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

iii.) ist α ein Grenzordinal, so ist $T{\uparrow}^{\alpha} := lub(\{T{\uparrow}^{\beta}; \beta < \alpha\})$ und $T{\downarrow}^{\alpha} := glb(\{T{\downarrow}^{\beta}; \beta < \alpha\})$.

Es gilt

Satz 2.2.
Sei D ein vollständiger Verband und $T : D \longrightarrow D$ eine monotone Abbildung. Für jede Ordinalzahl α gilt $T{\uparrow}^{\alpha} \leq lfp(T)$ und $T{\downarrow}^{\alpha} \geq gfp(T)$. Außerdem existieren Ordinalzahlen β und γ mit $T{\uparrow}^{\beta} = lfp(T)$ und $T{\downarrow}^{\gamma} = gfp(T)$.

Beweis: vgl. [Llo87], Proposition 5.3. □

Ist T *aufwärts ω-stetig* (bzw. *abwärts ω-stetig*), d.h. für jede aufsteigende Kette $d_0 \leq d_1 \leq d_2 \leq \cdots$ (bzw. absteigende Kette $d_0 \geq d_1 \geq d_2 \geq \cdots$) gilt $T(lub(\{d_i; i \geq 0\}) = lub(\{T(d_i); i \geq 0\})$ (bzw. $T(glb(\{d_i; i \geq 0\}) = glb(\{T(d_i); i \geq 0\}))$, so ist T monoton und β bzw. γ können im obigen Satz gleich ω gewählt werden. Dies besagt

Satz 2.3.
Ist D ein vollständiger Verband und $T : D \longrightarrow D$ eine aufwärts ω-stetige (bzw. eine abwärts ω-stetige) Abbildung, so ist $lfp(T) = T{\uparrow}^{\omega} = lub(\{T^n(bottom); n \geq 0\})$ (bzw. $gfp(T) = T{\downarrow}^{\omega} = glb(\{T^n(top); n \geq 0\}))$.

Beweis: Folgerung von Satz 2.4. □

[3] reflexive, transitive und antisymetrische binäre Relation

[4] D bezeichne sowohl die Trägermenge der Ordnung als auch die Ordnung selbst.

Der nächste Satz ist eine Verallgemeinerung von Satz 2.3 für abwärts ω-stetige Abbildungen. Er gilt analog für aufwärts ω-stetige Abbildungen, wird aber in dieser Form hier nicht benötigt.

Satz 2.4.
Es sei D ein vollständiger Verband und $T : D \longrightarrow D$ eine abwärts ω-stetige Abbildung. Weiter sei d ein Element von D mit $d \geq T(d)$. Dann ist $d\text{-}gfp(T) := glb(\{T^n(d); n \geq 0\})$ der größte Fixpunkt von T, der kleiner oder gleich d ist.

Beweis: vgl. [Baa96]. □

2.2 Automaten und Sprachen

Im folgenden Abschnitt werden Begriffe und grundlegende Aussagen zu Automaten und Sprachen eingeführt, auf die in den folgenden Kapiteln Bezug genommen wird. Ausführlichere Informationen dazu finden sich z.B. in [HU79] und [Eil74].

Ist Σ ein *endliches Alphabet*, so bezeichnet Σ^* die Menge aller *endlichen Wörter* über Σ. Ist $W \in \Sigma^*$ mit $W = a_0 \cdots a_{n-1}, a_i \in \Sigma$, $0 \leq i \leq n-1$, $n \in \mathbb{N}$, so ist $|W| := n$ die Länge des Wortes W. Durch ε werde das leere Wort bezeichnet, d.h. das Wort der Länge 0. Weiter bezeichne Σ_ε die Vereinigung von Σ und ε, also $\Sigma \cup \{\varepsilon\}$. Das endliche Wort W kann auch als Abbildung von der Ordinalzahl $n = \{0, \ldots, n-1\}$ in Σ betrachtet werden: $W(i) := a_i$ für alle $0 \leq i < n$. Ein *unendliches Wort* (*ω-Wort*) ist eine Abbildung von ω in Σ. Es ist Σ^ω die Menge aller unendlichen Wörter über Σ. Ein ω-Wort W wird oft als unendliche Sequenz $W(0)W(1)W(2)\cdots$ geschrieben. Die Länge von W ist ∞ und wird, wie im endlichen Fall, mit $|W|$ bezeichnet. Für eine Menge L von endlichen Wörtern über Σ, d.h. $L \subseteq \Sigma^*$, und für einen Buchstaben a aus Σ sei $L \cdot a := \{W \cdot a;\, W \in L\}$ die Menge der mit a konkatenierten Wörter aus L. Die Menge L^ω enthält alle ω-Wörter über Σ der Form $W_1W_2W_3\cdots$ mit $W_i \in L$ für alle $i \geq 1$.

Ein *Semi-Automat (mit Worttransitionen)* ist ein Tripel $\mathcal{A} = (\Sigma, Q, E)$, das aus einem endlichen Alphabet Σ, einer endlichen Zustandsmenge Q sowie einer endlichen Menge von Transitionen $E \subseteq Q \times \Sigma^* \times Q$ besteht. Dabei kann $\mathcal{A}$ als endlicher Graph mit Knotenmenge Q und Kantenmenge E dargestellt werden, wobei die Label der Kanten endliche Wörter aus Σ^* sind. Ist $E \subseteq Q \times \Sigma_\varepsilon \times Q$, so heißt $\mathcal{A}$ *Semi-Automat ohne Worttransitionen.* Ist $E \subseteq Q \times \Sigma \times Q$, so heißt $\mathcal{A}$ *Semi-Automat mit Buchstaben-Transitionen.*

Es sei nun $\mathcal{A}$ ein Semi-Automat, p und q seien Zustände von $\mathcal{A}$. Es existiert ein *endlicher Pfad* der Länge $n \geq 0$ von p nach q in $\mathcal{A}$ mit Label U, falls es für alle $1 \leq i \leq n$ Transitionen (p_{i-1}, U_i, p_i) in $\mathcal{A}$ gibt mit $p_0 = p$, $p_n = q$ und U=$U_1 \cdots U_n$. Für $n = 0$ ist dies ein *leerer Pfad* mit Label $U = \varepsilon$ und $q = p$. Die Sequenz $q_0, V_1, q_1, V_2, q_2, \ldots, V_n, q_n$ mit $q_i \in Q$ für alle $0 \leq i \leq n$ und $V_i \in \Sigma^*$ für alle $1 \leq i \leq n$ bezeichnet einen endlichen Pfad von q_0 nach q_n, falls für alle $1 \leq i \leq n$ ein endlicher Pfad von q_{i-1} nach q_i existiert mit Label V_i. Es ist zu beachten, daß (q_{i-1}, V_i, q_i) keine Transition in $\mathcal{A}$ sein muß; dies wird ansonsten explizit erwähnt. Entsprechend stellt die Sequenz $q_0, V_1, q_1, V_2, q_2, V_3, \ldots$ einen *unendlichen Pfad* von q_0 mit Label $W = V_1V_2V_3\cdots$ dar, falls für alle $i \geq 1$ ein endlicher Pfad von q_{i-1} nach q_i mit Label V_i existiert. Dabei muß für unendlich viele Indizes $i \geq 1$ der Pfad von q_{i-1} nach q_i ungleich dem leeren Pfad sein. Das Wort W kann endlich oder unendlich sein. Ist W ein endliches Wort, so existiert ein $i \geq 0$, so daß — aufgrund der Endlichkeit der Zustandsmenge — q_i auf einem *ε-Zyklus* liegt. Die Sequenz $q_0, W, q_i, \varepsilon, q_i, \varepsilon, q_i, \ldots$ bezeichnet also einen unendlichen Pfad in $\mathcal{A}$ mit Label W vom

Zustand q_0 aus. Dabei liegt ein Zustand q von $\mathcal{A}$ auf einem ε-Zyklus, falls ein nicht-leerer Pfad in $\mathcal{A}$ von q nach q mit Label ε existiert .

Sind I und J Teilmengen von Q, so bezeichnet $L_{\mathcal{A}}(I,J)$ (oder $L(I,J)$, falls der Bezug zu $\mathcal{A}$ klar ist) die Menge der endlichen Wörter über Σ, die Label eines endlichen Pfades von einem Zustand aus I zu einem Zustand aus J sind. Ist $I = \{p\}$ und $J = \{q\}$, so schreiben wir auch $L_{\mathcal{A}}(p,q)$ (bzw. $L(p,q)$). Zusammen mit I und J ist $\mathcal{A}$ ein *endlicher Automat* $(\mathcal{A}, I, J)$ mit Anfangszustandsmenge I und Endzustandsmenge J. Bekanntlich ist damit $L_{\mathcal{A}}(I,J)$ eine reguläre Sprache. Man kann o.E. davon ausgehen, daß I und J einelementig sind. Ist dies nicht der Fall, dann ergänzt man (mit linearem Zeitaufwand) $\mathcal{A}$ um zwei neue Zustände p, q und fügt zu jedem $p' \in I$ eine Transition (p, ε, p') sowie zu jedem $q' \in J$ eine Transition (q', ε, q) in $\mathcal{A}$ ein. Bezeichnet $\mathcal{A}'$ den so erhaltenen Automaten, dann gilt offensichtlich $L_{\mathcal{A}}(I,J) = L_{\mathcal{A}'}(p,q)$.

Der Automat $(\mathcal{A}, I, J)$ kann auch als Büchi-Automat aufgefaßt werden, der als solcher die Sprache $B_{\mathcal{A}}(I,J)$ bzw. $B(I,J) = \{W \in \Sigma^\omega; W$ ist Label eines unendlichen Pfades, der mit einem Zustand aus I startet und einen Zustand aus J unendlich oft durchläuft$\}$ akzeptiert. Nach [Eil74], Theorem 1.4 ist eine Büchi-erkennbare Sprache $B_{\mathcal{A}}(I,J)$ endliche Vereinigung von Sprachen der Form $H \cdot K^\omega$ mit den regulären Sprachen H und K, weshalb Büchi-erkennbare Sprachen auch *ω-reguläre* Sprachen genannt werden. Büchi-erkennbare Sprachen sind unter den booleschen Operationen Vereinigung, Durchschnitt und Komplement abgeschlossen.

Schließlich wird die durch $\mathcal{A}$ und einen Zustand p von $\mathcal{A}$ definierte Sprache $U_{\mathcal{A}}(p)$ bzw. $U(p) := \{W \in \Sigma^* \cup \Sigma^\omega; W$ ist Label eines unendlichen Pfades, der mit p beginnt$\}$ von Bedeutung sein. Ist $\mathcal{A}$ ein Semi-Automat mit Buchstaben-Transitionen[5], so ist $U_{\mathcal{A}}(p) \subseteq \Sigma^\omega$, und man verifiziert leicht, daß dann $U_{\mathcal{A}}(p) = B_{\mathcal{A}}(\{p\}, Q)$ gilt.

Ist $\mathcal{A} = (\Sigma, Q, E)$ ein Semi-Automat, so ist die ε-Hülle zu $I \subseteq Q$ definiert durch

$$\begin{aligned} \varepsilon\text{-}closure_{\mathcal{A}}(I) \quad := \quad & \{q \in Q; \text{ es existiert ein } q' \in I \text{ und ein endlicher Pfad} \\ & \text{von } q' \text{ nach } q \text{ mit Label } \varepsilon\}; \end{aligned}$$

die Nachfolgermenge von I bzgl. $a \in \Sigma$ ist durch

$$next_{\mathcal{A}}(I,a) \quad := \quad \{q \in Q; \text{ es existiert ein } q' \in I \text{ mit } (q', a, q) \in E\}$$

gegeben; schließlich bezeichnen für $W \in \Sigma^*$ sowie I und a wie oben

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

die Nachfolgermengen einer Zustandsmenge und eines Wortes. Meist wird aus dem Zusammenhang klar sein, auf welchen Automaten Bezug genommen wird; wir schreiben dann kürzer $\varepsilon\text{-}closure(I)$, $next(I,a)$ bzw. $next_\varepsilon(I,W)$. Die Berechnung dieser Mengen ist mit polynomialem Aufwand in der Größe von $\mathcal{A}$ (und W) möglich.

Wir benötigen diese Definitionen zur Simulation des zu $\mathcal{A}$ gehörenden *Potenzmengenautomaten* $\mathcal{B} := (\Sigma, Q', E')$ mit $Q' := 2^Q \cap \{F \subseteq Q; F = \varepsilon\text{-}closure(F)\}$ (unter *ε-closure* abgeschlossene Teilmengen von Q) und $E' := \{(q, a, q'); q, q' \in Q'$ und $q' := next_{\varepsilon\mathcal{A}}(q, a)\}$.

Für $next_\varepsilon(I,W)$ zeigt man leicht durch Induktion über die Länge n von W:

[5] Es genügt auch, daß kein Zustand auf einem ε-Zyklus liegt.

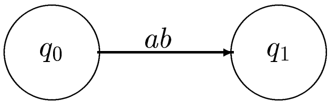
Lemma 2.5.
Es sei $\mathcal{A} = (\Sigma, Q, E)$ ein Semi-Automat *ohne Worttransitionen* und $W \in \Sigma^*$. Weiter seien q und q' Zustände aus Q. Dann gilt:

$$q' \in next_\varepsilon(\{q\}, W) \text{ gdw. } W \in L_{.}(q, q')$$

$\square$

Wie das nächste Beispiel zeigt, ist es in Lemma 2.5 tatsächlich nötig, von einem Semi-Automaten ohne Worttransitionen auszugehen.

Beispiel 2.6.
Es sei ein Semi-Automat $\mathcal{A}$ (mit Worttransitionen) wie folgt gegeben:



Offensichtlich ist $ab \in L(q_0, q_1)$. Es gilt aber auch $q_1 \notin next_\varepsilon(\{q_0\}, ab) = \emptyset$. $\diamond$

Aus diesem Grund beziehen sich $next(I, a)$ und $next_\varepsilon(I, a)$ immer auf einen Semi-Automaten ohne Worttransitionen. Dies kann in folgendem Sinne o.E. in bezug auf die eingeführten Sprachen stets angenommen werden:

Lemma 2.7.
Es sei $\mathcal{A} = (\Sigma, Q, E)$ ein Semi-Automat (mit Worttransitionen). Zu diesem läßt sich — mit linearem Aufwand — ein Semi-Automat $\mathcal{B} = (\Sigma, Q', E')$ ohne Worttransitionen mit folgenden Eigenschaften konstruieren:

i.) $Q \subseteq Q'$;

ii.) von keinem $q \in Q' \setminus Q$ existiert ein nicht-leerer Pfad zu q, der nur Zustände aus $Q' \setminus Q$ enthält;

iii.) kein $q \in Q' \setminus Q$ liegt auf einem ε-Zyklus;

iv.) für alle $I, J \subseteq Q$ ist $L_\mathcal{A}(I, J) = L_\mathcal{B}(I, J)$;

v.) für alle $q \in Q$ ist $U_\mathcal{A}(q) = U_\mathcal{B}(q)$;

vi.) für alle $I, J \subseteq Q$ ist $B_\mathcal{A}(I, J) = B_\mathcal{B}(I, J)$.

Beweis: Den Automaten $\mathcal{B}$ erhält man aus $\mathcal{A}$, indem jede Worttransition $(p, a_1 \cdots a_n, q) \in E$ mit $n > 1$ durch die Transitionen $(p, a_1, p_1), (p_1, a_2, p_2), \ldots, (p_{n-1}, a_n, q)$ ersetzt wird, wobei $p_1, \ldots, p_{n-1}$ jeweils neue Zustände sind. Die übrigen Transitionen aus E werden unverändert in E' übernommen. Offensichtlich läßt sich $\mathcal{B}$ mit linearem Aufwand aus $\mathcal{A}$ konstruieren. Außerdem sieht man leicht, daß für $\mathcal{B}$ die Aussagen i.) – vi.) gelten. $\square$

Zusätzlich zu Worttransitionen können auch ε-Transitionen eliminiert werden.

Lemma 2.8.
Ist $\mathcal{A} = (\Sigma, Q, E)$ ein Semi-Automat ohne Worttransitionen, so existiert ein (mit polynomialem Aufwand aus $\mathcal{A}$ konstruierbarer) Semi-Automat $\mathcal{B} = (\Sigma, Q, E')$ mit Buchstaben-Transitionen, so daß für alle $I, J \subseteq Q$ gilt:

i.) $L_{\mathcal{A}}(I,J) \setminus \{\varepsilon\} = L_{\mathcal{B}}(I,J) \setminus \{\varepsilon\}$ und

ii.) $B_{\mathcal{A}}(I,J) = B_{\mathcal{B}}(I,J)$.

Beweis: Die Menge E' wird wie folgt definiert: $E' := \{(p,a,q) \in Q \times \Sigma \times Q$; es existiert ein Pfad in $\mathcal{A}$ von p nach q mit Label a$\}$. Wegen $(p,a,q) \in E'$ gdw. $q \in next_\varepsilon(\{p\},a)$, läßt sich E' aus $\mathcal{A}$ mit polynomialem Aufwand konstruieren, indem zu allen Paaren (p,a) aus $Q \times \Sigma$ die Menge $next_\varepsilon(\{p\},a)$ berechnet wird. Man weist die Eigenschaften i.) und ii.) leicht nach. □

Sind $\mathcal{A} = (\Sigma, Q_1, E_1)$ und $\mathcal{B} = (\Sigma, Q_2, E_2)$ zwei Semi-Automaten mit Buchstaben-Transitionen, so besitzt der zugehörige — mit polynomialem Aufwand konstruierbare — *Produktautomat* $\mathcal{C} = (\Sigma, Q, E)$ mit $Q := Q_1 \times Q_2$ und $E := \{(p_1,p_2),a,(q_1,q_2) \in Q \times \Sigma \times Q; (p_1,a,q_1) \in E_1$ und $(p_2,a,q_2) \in E_2\}$ die Eigenschaft:

Satz 2.9.
Für alle $I_1, J_1 \subseteq Q_1$ und $I_2, J_2 \subseteq Q_2$ gilt: $L_{\mathcal{C}}(I_1 \times I_2, J_1 \times J_2) = L_{\mathcal{A}}(I_1,J_1) \cap L_{\mathcal{B}}(I_2,J_2)$. □

Wir werden später außerdem die folgende Aussage zur Sprache $U_{\mathcal{A}}(q)$ benötigen:

Lemma 2.10.
Es sei $\mathcal{A}$ ein Semi-Automat ohne Worttransitionen, q ein Zustand in $\mathcal{A}$ sowie $W = a_1a_2a_3\cdots$ ein ω-Wort. Für $T_i := next_\varepsilon(q, a_1\cdots a_i)$, $i \geq 0$, gilt: $W \notin U(q)$ gdw. ein $k \geq 0$ mit $T_k = \emptyset$ existiert; insbesondere ist dann $T_i = \emptyset$ für alle $i \geq k$.

Beweis: „$\Rightarrow$“: Es sei $T_i \neq \emptyset$ für alle $i \geq 0$. Es existiert also für alle $i \geq 0$ ein Zustand $q_i \in T_i$, so daß $W_i := a_1\cdots a_i \in L(q,q_i)$ gilt. Wir betrachten den folgenden Baum: Die Wurzel des Baumes ist mit q markiert. Nachfolgerknoten von q sind genau die mit q' markierten Knoten, so daß ein $\mathcal{A}$-Pfad von q nach q' mit Label a_1 existiert. Ein solcher mit q' markierte Knoten erhält nun einen mit q'' markierten Nachfolgerknoten genau dann, wenn ein $\mathcal{A}$-Pfad von q' nach q'' mit Label a_2 existiert. Entsprechend führt man dies für $a_3, a_4, a_5, \ldots$ fort. Da Q eine endliche Menge ist, besitzt jeder Knoten des Baumes nur endlich viele Nachfolgerknoten. Wegen $W_i \in L(q,q_i)$ für alle $i \geq 0$ und Lemma 2.5 enthält der Baum beliebig lange Pfade. Nach dem Lemma von König[6] enthält der Baum einen unendlich langen Pfad. Dies liefert $W \in U(q)$.

„$\Leftarrow$“: Es existiere ein $k \geq 0$ mit $T_k = \emptyset$. Gilt $W \in U(q)$, so existieren Zustände $q_1, q_2, \ldots$, so daß $q = q_0, a_1, q_1, a_2, q_2, \ldots$ ein unendlicher Pfad in $\mathcal{A}$ ist. Damit ist nach Lemma 2.5 $q_k \in T_k$ für alle $k \geq 0$, im Widerspruch zur Voraussetzung. □

Nach [GJ79] ist das Inklusionsproblem für reguläre Sprachen PSPACE-vollständig. Wir werden nun einen NPSPACE-Entscheidungsalgorithmus[7] angeben, der das Inklusionsproblem $L_{\mathcal{A}}(I_1,J_1) \subseteq L_{\mathcal{B}}(I_2,J_2)$ für zwei Semi-Automaten $\mathcal{A} = (\Sigma, Q_1, E_1)$ und $\mathcal{B} = (\Sigma, Q_2, E_2)$ sowie Zustandsmengen $I_1, J_1 \subseteq Q_1$ und $I_2, J_2 \subseteq Q_2$ entscheidet.

Es gilt $L_{\mathcal{A}}(I_1,J_1) \subseteq L_{\mathcal{B}}(I_2,J_2)$ gdw. $L_{\mathcal{A}}(I_1,J_1) \cap (\Sigma^* \setminus L_{\mathcal{B}}(I_2,J_2)) = \emptyset$. Der Algorithmus geht deshalb (im Prinzip) vom endlichen Automaten $(\mathcal{A}, I_1, J_1)$ zum Potenzmengenautomaten $\mathcal{A}'$ über. Der Anfangszustand von $\mathcal{A}'$ ist $\varepsilon\text{-}closure_{\mathcal{A}}(I_1)$ und die Endzustände

[6] In einem endlich verzweigten Baum mit beliebig langen Pfaden existiert ein unendlicher Pfad (vgl. [Kö35]).

[7] nicht-deterministischer Algorithmus; jeder Berechnungspfad braucht nur polynomialen Platz (in der Größe der Eingabe) und terminiert stets mit Ausgabe „ja“ bzw. „nein“.

bestehen aus den Mengen $J \subseteq Q_1$, $J = \varepsilon\text{-}closure_{\mathcal{A}}(J)$, für die $J \cap J_1 \neq \emptyset$ gilt. Entsprechend bezeichne $\mathcal{B}'$ den Potenzmengenautomat, den man aus dem endlichen Automaten $(\mathcal{B}, I_2, J_2)$ erhält, wobei jedoch End- und Nicht-Endzustände vertauscht seien. Die Endzustandsmenge besteht somit aus Mengen $J \subseteq Q_2$, $J = \varepsilon\text{-}closure_{\mathcal{A}}(J)$, für die $J \cap J_2 = \emptyset$ gilt. Der deterministische endliche Automat $\mathcal{B}'$ akzeptiert also die Sprache $\Sigma^* \setminus L_{\mathcal{B}}(I_2, J_2)$. Um den Durchschnitt der beiden Sprachen zu $\mathcal{A}'$ und $\mathcal{B}'$ zu erhalten, wird der Produktautomat der beiden deterministischen Automaten gebildet. Dieser besitzt höchstens $2^{2 \cdot n}$ Zustände ($n = max(|Q_1|, |Q_2|)$). Bekanntlich ist die Sprache dieses Automaten genau dann nicht leer, wenn ein Wort der Länge kleiner als $2^{2 \cdot n}$ existiert, welches vom Produktautomaten akzeptiert wird (Pumping-Lemma). Der Algorithmus rät nun nicht-deterministisch ein solches Wort, falls es existiert, und simuliert den Produktautomaten. Die Simulation braucht nur polynomialen Platz, denn es ist nicht nötig, den exponentiell großen Produktautomaten explizit zu konstruieren und abzuspeichern.

Dieser Algorithmus — wie auch alle anderen Algorithmen dieser Arbeit — wird in PASCAL-ähnlicher Notation formuliert. Die Ausgaben „ja" bzw. „nein" schließen jeweils die Terminierung des Algorithmus mit ein. Auf eine explizite „Halt"-Anweisung wird deshalb verzichtet.

Algorithmus 2.11 (Entscheidungsalgorithmus: Inklusion regulärer Sprachen).

Eingabe: Semi-Automaten $\mathcal{A} = (\Sigma, Q_1, E_1)$ und $\mathcal{B} = (\Sigma, Q_2, E_2)$ ohne Worttransitionen; $I_1, J_1 \subseteq Q_1$ und $I_2, J_2 \subseteq Q_2$.

Ausgabe: Es existiert eine Berechnung mit Ausgabe „ja" gdw. $L_{\mathcal{A}}(I_1, J_1) \not\subseteq L_{\mathcal{B}}(I_2, J_2)$

$T_1 := \varepsilon\text{-}closure_{\mathcal{A}}(I_1)$;
$T_2 := \varepsilon\text{-}closure_{\mathcal{B}}(I_2)$;
$n := max(|Q_1|, |Q_2|)$;
$z := 0$;
while $z < 2^{2 \cdot n} - 1$ and not$((T_1 \cap J_1) \neq \emptyset$ and $(T_2 \cap J_2) = \emptyset)$ do begin
  $z := z + 1$;
  Wähle (nicht-det.) ein $a \in \Sigma$;
  $T_1 := next_{\varepsilon\mathcal{A}}(T_1, a)$;
  $T_2 := next_{\varepsilon\mathcal{B}}(T_2, a)$
end;
If $T_1 \cap J_1 \neq \emptyset$ and $T_2 \cap J_2 = \emptyset$ then Ausgabe „ja" else Ausgabe „nein". △

Korrektheit: Gibt der Algorithmus „ja" aus, so existiert ein Wort $W = a_1 \cdots a_m \in \Sigma^*$, wobei a_i der im i-ten Schleifendurchlauf für a gewählte Buchstabe ist, mit $m < 2^{2 \cdot n}$, so daß für $T_1 := next_{\varepsilon\mathcal{A}}(I_1, W)$ und $T_2 := next_{\varepsilon\mathcal{B}}(I_2, W)$ gilt: $T_1 \cap J_1 \neq \emptyset$ und $T_2 \cap J_2 = \emptyset$. Damit gilt nach Lemma 2.5: $W \in L_{\mathcal{A}}(I_1, J_1)$ und $W \notin L_{\mathcal{B}}(I_2, J_2)$, also $L_{\mathcal{A}}(I_1, J_1) \not\subseteq L_{\mathcal{B}}(I_2, J_2)$. Dies zeigt die Korrektheit des Algorithmus 2.11.

Vollständigkeit: Ist $L_{\mathcal{A}}(I_1, J_1) \not\subseteq L_{\mathcal{B}}(I_2, J_2)$, so existiert ein Wort $W = a_1 \cdots a_m \in \Sigma^*$, für das $W \in L_{\mathcal{A}}(I_1, J_1) \setminus L_{\mathcal{B}}(I_2, J_2)$ gilt. Für $T_{1,m} := next_{\varepsilon\mathcal{A}}(I_1, W)$ und $T_{2,m} := next_{\varepsilon\mathcal{B}}(I_2, W)$ gilt damit $T_{1,m} \cap J_1 \neq \emptyset$ und $T_{2,m} \cap J_2 = \emptyset$ nach Lemma 2.5. Es ist $|2^{Q_1} \times 2^{Q_2}| \leq 2^{2 \cdot n}$. Ist $m > 2^{2 \cdot n} - 1$, dann existieren für $T_{1,i} := next_{\varepsilon\mathcal{A}}(I_1, a_1 \cdots a_i)$ und $T_{2,i} := next_{\varepsilon\mathcal{B}}(I_2, a_1 \cdots a_i)$ $(0 \leq i \leq m)$ Zahlen l und r mit $0 \leq l < r \leq 2^{2 \cdot n}(\leq$

m) sowie $T_{1,l} = T_{1,r}$ und $T_{2,l} = T_{2,r}$. Damit gilt $T_{1,m} = next_{\varepsilon\mathcal{A}}(I_1, a_1 \cdots a_l a_{r+1} \cdots a_m)$ und $T_{2,m} = next_{\varepsilon\mathcal{B}}(I_2, a_1 \cdots a_l a_{r+1} \cdots a_m)$. Also gelten auch für $W' = a_1 \cdots a_l a_{r+1} \cdots a_m$, $|W'| < m$, die Aussagen $W' \in L_{\mathcal{A}}(I_1, J_1)$ und $W' \notin L_{\mathcal{B}}(I_2, J_2)$. Für ein Wort W, $|W| = m$, minimaler Länge gilt somit $m < 2^{2 \cdot n}$. Dann existiert aber auch eine Berechnung mit Ausgabe „ja“: Für $W = a_1 \cdots a_m$ wählt der Algorithmus im i-ten Schleifendurchlauf $a = a_i$ für alle $1 \leq i \leq m$. Dies liefert eine Berechnung mit Ausgabe „ja“. Ist schon vor dem m-ten Schleifendurchlauf die while-Bedingung nicht erfüllt, so muß wegen $z < 2^{2 \cdot n}$ sowohl $T_1 \cap J_1 \neq \emptyset$ als auch $T_2 \cap J_2 = \emptyset$ gelten, was ebenfalls zur Ausgabe „ja“ führt.

Komplexität: Die Terminierung des Algorithmus ist trivial. Der Algorithmus muß lediglich T_1, T_2, z und n speichern. Für T_1, T_2 und n ist klar, daß dies nur polynomialen Platz erfordert. Da zur Speicherung von z der Platz $log(2^{2 \cdot n} - 1) \leq 2 \cdot n$ ausreicht, ist auch für z der Speicherplatz polynomial. Außerdem brauchen die arithmitischen Operationen und die Berechnungen von $\varepsilon\text{-}closure(I)$ und $next_\varepsilon(T, a)$ nur polynomiale Zeit. Algorithmus 2.11 ist also ein NPSPACE-Algorithmus.

Nach Savitch's Theorem [HU69] gilt NPSPACE=PSPACE. Insgesamt zeigt dies die Existenz eines PSPACE-Algorithmus für das Inklusionsproblem regulärer Sprachen.

Neben dem Inklusionsproblem für reguläre Sprachen ist auch das Inklusionsproblem für ω-reguläre Sprachen PSPACE-vollständig. In [SVW87] wurde gezeigt, daß das Äquivalenz-Problem für Büchi-Automaten in PSPACE liegt. Wegen $L_1 \subseteq L_2$ gdw. $L_1 \cap L_2 = L_1$, und da der Büchi-Automat für den Durchschnitt von ω-regulären Sprachen in polynomialer Zeit konstruierbar ist (vgl. [Tho90]), erhalten wir auch einen PSPACE-Algorithmus für das Inklusionsproblem. Wie in [Baa96] festgestellt, kann man das Inklusionsproblem für reguläre Sprachen in polynomialer Zeit auf das Inklusionsproblem für ω-reguläre Sprachen reduzieren; für die regulären Sprachen L_1 und L_2 gilt nämlich: $L_1 \subseteq L_2$ gdw. $L_1 \cdot \{\#\}^\omega \subseteq L_2 \cdot \{\#\}^\omega$ (für $\# \notin \Sigma$). $L_1 \cdot \{\#\}^\omega$ und $L_2 \cdot \{\#\}^\omega$ sind ω-reguläre Sprachen, deren zugehörige Büchi-Automaten leicht — in polynomialer Zeit — aus den Automaten für L_1 und L_2 gewonnen werden können. Diese Reduktion impliziert die PSPACE-Härte des Inklusionsproblems ω-regulärer Sprachen.

Kapitel 3

Die terminologische Repräsentationssprache $\mathcal{ALN}$

Die Sprache $\mathcal{ALN}$ erweitert die terminologische Repräsentationssprache $\mathcal{FL}_0$, die nur Konzeptkonjunktion und Werterestriktion erlaubt, um primitive Negation sowie Zahlenrestriktion. Wir definieren außerdem die Teilsprachen $\mathcal{AL}_0$ und $\mathcal{FLN}$ von $\mathcal{ALN}$, die $\mathcal{FL}_0$ um primitive Negation bzw. Zahlenrestriktion ergänzen.

Definition 3.1 (Syntax von $\mathcal{ALN}$).
Es seien N_C und N_P Mengen von Konzeptnamen (Konzepten) sowie N_R eine Menge von Rollennamen (Rollen). Diese Mengen seien paarweise disjunkt. Die Menge der *Konzeptterme* von $\mathcal{ALN}$ ($\mathcal{ALN}$-Konzeptterme) ist induktiv wie folgt definiert:

i.) Für $C \in N_C$ ist C ein Konzeptterm (*atomares Konzept*).

ii.) Für $P \in N_P$ sind P (*atomares Konzept*) und $(\neg P)$ (*primitive Negation*) Konzeptterme.

iii.) Für $n \in \mathbb{N}$ und $R \in N_R$ sind $\exists^{\geq n} R$ (*Maximum-Restriktion*) und $\exists^{\leq n} R$ (*Minimum-Restriktion*) Konzeptterme (*Zahlenrestriktionen*).

Für bereits definierte Konzeptterme C und D sowie $R \in N_R$ sind

iv.) $C \sqcap D$ (*Konzeptkonjunktion*) und

v.) $\forall R.C$ (*Werterestriktion*) ebenfalls Konzeptterme.

Eine *Terminologie* (*T-Box*) T von $\mathcal{ALN}$ ($\mathcal{ALN}$-Terminologie) besteht aus einer endlichen Menge von (T-)Axiomen der Form $A = D$ (Axiom zu A), dabei bezeichnet A ein Konzept aus N_C und D einen Konzeptterm. Zu jedem Konzept $A \in N_C$, das in T vorkommt, existiert stets genau ein T-Axiom. Wir nennen solche Konzepte *definierte Konzepte* von T. Die Konzepte $P \in N_P$, die in T vorkommen, heißen *primitive Konzepte* in T; zu diesen existieren keine T-Axiome. $\diamond$

Neben $\mathcal{ALN}$ betrachten wir auch $\mathcal{ALN}$-Teilsprachen. Die Sprache $\mathcal{FL}_0$ erlaubt neben atomaren Konzepten nur Konzeptkonjunktion und Werterestriktion. Die Sprache $\mathcal{AL}_0$ ergänzt $\mathcal{FL}_0$ um primitive Negation. Schließlich erweitert $\mathcal{FLN}$ die Sprache $\mathcal{FL}_0$ um

Zahlenrestriktionen. Die Terminologien zu diesen Repräsentationssprachen werden analog zu $\mathcal{ALN}$-Terminologien definiert.

In Anlehnung an die Beispiele (1.1) und (1.2) der Einleitung betrachten wir nun Beispiele für $\mathcal{ALN}$-Terminologien, die terminologische Zyklen als natürliche Beschreibung von Konzepten aufweisen sowie Aussagen über (reflexiv-)transitive Hüllen von Relationen enthalten.

Beispiel 3.2.
Es seien Lebewesen, Mensch, männlich, Frau und Mann Konzeptnamen sowie eltern ein Rollenname. Eine $\mathcal{ALN}$-Terminologie kann wie folgt definiert werden:

$$\begin{aligned}
\mathsf{Mensch} &= \mathsf{Lebewesen} \sqcap \exists^{\geq 2}\mathsf{eltern} \sqcap \exists^{\leq 2}\mathsf{eltern} \sqcap \forall\mathsf{eltern}.\mathsf{Mensch} \\
\mathsf{Mann} &= \mathsf{Mensch} \sqcap \mathsf{männlich} \\
\mathsf{Frau} &= \mathsf{Mensch} \sqcap \neg\mathsf{männlich}
\end{aligned}$$

Das Konzept Mensch wird — wie in der Einleitung angesprochen — über die reflexiv-transitive Hülle[1] der Relation eltern definiert. Ein Mensch ist demnach ein Lebewesen, welches genau zwei Elternteile hat und dessen Vorfahren ebenfalls Lebewesen mit zwei Elternteilen sind. Es ist zu beachten, daß die Einführung einer neuen Rolle vorfahren nicht das Gewünschte leistet, da diese Rolle unabhängig von eltern ist. ◇

Ein weiteres Beispiel:

Beispiel 3.3.
Es seien Binärbaum, Ternärbaum, Blätter-Binärbaum und Baum Konzeptnamen sowie direkter-nachfolger ein Rollenname, dann ist

$$\begin{aligned}
\mathsf{Binärbaum} &= \mathsf{Baum} \sqcap \exists^{\leq 2}\mathsf{direkter\text{-}nachfolger} \sqcap \\
&\quad \forall\mathsf{direkter\text{-}nachfolger}.\mathsf{Binärbaum} && (3.1) \\
\mathsf{Ternärbaum} &= \mathsf{Baum} \sqcap \exists^{\leq 3}\mathsf{direkter\text{-}nachfolger} \sqcap \\
&\quad \forall\mathsf{direkter\text{-}nachfolger}.\mathsf{Ternärbaum} && (3.2) \\
\mathsf{Blätter} &= \mathsf{Baum} \sqcap \exists^{\leq 0}\mathsf{direkter\text{-}nachfolger} && (3.3)
\end{aligned}$$

eine $\mathcal{ALN}$-Terminologie mit den definierten Konzepten Binärbaum, Ternärbaum sowie Blätter. Das Konzept Baum ist primitiv. Ein Individuum gehört zum Binärbaum, falls es zum Konzept Baum gehört und höchstens zwei direkte-nachfolger besitzt sowie alle (direkten und indirekten) Nachfolger (transitive Hülle von direkter-nachfolger) auch zum Binärbaum gehören. Das Konzept Ternärbaum ist analog definiert. Man wird erwarten, daß alle Individuen aus Binärbaum auch zum Ternärbaum gehören. Wir werden jedoch sehen, daß nicht alle Semantiken dieser Intuition entsprechen. Durch Blätter werden alle Knoten des Baumes beschrieben, zu denen keine Nachfolger existieren, die also Blätter des Baumes sind. ◇

Als nächstes werden wir die bisher nur intuitiv angegebene Semantik formal modell-theoretisch definieren.

[1] Ist $R \subseteq M \times M$ eine binäre Relation über der Menge M, so sei $R^0 := \{(e,e); e \in M\}$ und $R^{n+1} := R \circ R^n, n \geq 0$, mit „$\circ$" als Komposition binärer Relationen. Damit ist $\bigcup_{n\geq 1} R^n$ die *transitive Hülle* von R und $\bigcup_{n\geq 0} R^n$ die *reflexiv-transitive Hülle* von R.

Definition 3.4 (Interpretation).
Eine *Interpretation* I besteht aus einer Menge $dom(I)$, *Domäne der Interpretation*, und einer Funktion $\cdot^I$, die jedem Konzeptnamen A eine Teilmenge A^I von $dom(I)$ zuordnet sowie jedem Rollennamen R eine binäre Relation über $dom(I)$, d.h. $R^I \subseteq dom(I) \times dom(I)$. Die Mengen A^I und R^I heißen *Extensionen* von A bzw. R bzgl. I.
Die Interpretationsfunktion von I, die damit auf Konzeptnamen und Rollennamen definiert ist, wird folgendermaßen induktiv auf beliebige Konzeptterme erweitert:

i.) Für $P \in N_P$ ist $(\neg P)^I := dom(I) \setminus P^I$.

ii.) Für $R \in N_R$ und $d \in dom(I)$ sei $R^I(d) := \{e; (d,e) \in R^I\}$ die Menge der *Rollenfüller* von d bzgl. R und I. Für $n \in \mathbb{N}$ ist dann $(\exists^{\geq n} R)^I := \{d \in dom(I); |R^I(d)| \geq n\}$ und $(\exists^{\leq n} R)^I := \{d \in dom(I); |R^I(d)| \leq n\}$.

Sind nun C und D Konzeptterme, für die C^I und D^I schon definiert sind, und ist $R \in N_R$, so ist

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

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

Damit ist die Interpretationsfunktion von I für alle Konzeptterme definiert.

Zwei Konzeptterme C und D heißen *äquivalent*, falls $C^I = D^I$ für alle Interpretationen I gilt.

Eine Interpretation I ist *Modell* einer T-Box T (*T-Modell*) gdw. $A^I = D^I$ für alle Axiome $A = D$ in T gilt[2].

Die *Semantik* von T ist durch die Menge der T-Modelle gegeben; genauer bezeichnen wir diese als *deskriptive Semantik*. Zwei Terminologien sind bzgl. der deskriptiven Semantik *äquivalent*, falls sie die gleiche Semantik besitzen. $\diamond$

Wir werden im folgenden Kapitel sehen, daß die deskriptive Semantik nicht immer die intuitive Bedeutung einer T-Box widerspiegelt, d.h. es gibt Modelle, bei denen die definierten Konzepte nicht die erwarteten Individuen enthalten. Für die deskriptive Semantik kann außerdem die Interpretation der primitiven Konzepte und Rollen (primitive Interpretation, Definition 4.2) nicht immer eindeutig auf die definierten Konzepte erweitert werden.

Beide Probleme treten auf, wenn die T-Box terminologische Zyklen enthält. Neben der deskriptiven Semantik betrachtet man für zyklische Terminologien deshalb auch weitere Semantiken.

[2] Offensichtlich genügt es, wenn I nur die in T vorkommenden Konzepte und Rollen interpretiert.

Kapitel 4

Semantik zyklischer Terminologien

Wie im vorangegangenen Kapitel erwähnt, tauchen für die deskriptive Semantik im Zusammenhang mit terminologischen Zyklen verschiedene Probleme auf. Wir werden diese — und andere — Probleme der deskriptiven Semantik in Beispielen genauer beleuchten. Diese führen uns zu alternativen Semantiken, auf deren Vor- und Nachteile wir kurz eingehen werden.

Zunächst werden die schon genannten Begriffe „terminologische Zyklen“ und „primitive Interpretation“ definiert.

Definition 4.1 (terminologische Zyklen).
Ist T eine Terminologie und sind A, B definierte Konzepte in T, so *benutzt* A das Konzept B *direkt*, genau dann wenn B auf der rechten Seite des Axioms zu A auftritt. Es sei „*benutzt*“ die transitive Hülle der Relation „benutzt direkt“. Die Terminologie T enthält nun einen *terminologischen Zyklus* genau dann, wenn ein definiertes Konzept A in T existiert, das sich selbst benutzt, d.h. A benutzt A. Zyklenfreie Terminologien heißen *azyklisch*. $\diamond$

Damit sind die Terminologien aus Beispiel 3.2 und 3.3 zyklisch, da z.B. die definierten Konzepte Mensch bzw. Binärbaum sich selbst benutzen.

Eine Interpretation I kann die primitiven Konzepte und Rollen einer T-Box T beliebig interpretieren. Damit diese Interpretation ein Modell von T darstellt, müssen die Extensionen der definierten Konzepte jedoch die durch T gegebenen Bedingungen erfüllen[1]. Dieser Unterschied zwischen primitiven Konzepten und Rollen auf der einen und definierten Konzepten auf der anderen Seite wird in folgender Definition festgehalten.

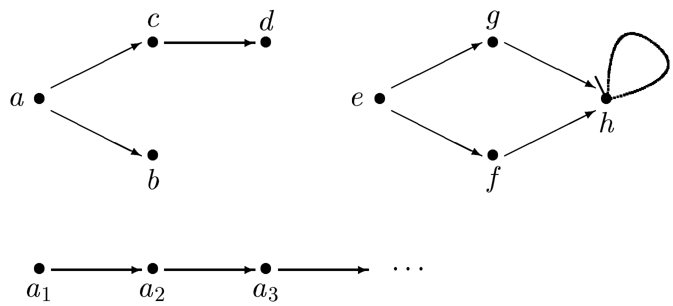
Definition 4.2 (primitive Interpretation und deren Erweiterung).
Es sei T eine T-Box mit den primitiven Konzepten $P_1, \ldots, P_m$, den Rollen $R_1, \ldots, R_k$ sowie den definierten Konzepten $A_1, \ldots, A_n$. Eine *primitive Interpretation* J besteht aus einer Menge $dom(J)$, der *Domäne der primitiven Interpretation*, sowie den Extensionen der primitiven Konzepte $P_1^J, \ldots, P_m^J$ und der Rollen $R_1^J, \ldots, R_k^J$.
Eine Interpretation I zu T *erweitert* die primitive Interpretation J genau dann, wenn $dom(I) = dom(J)$, $P_1^I = P_1^J, \ldots, P_m^I = P_m^J$ und $R_1^I = R_1^J, \ldots, R_k^I = R_k^J$ gilt. Durch J und ein n-Tupel $\underline{A} = (A_1^I, \ldots, A_n^I) \in (2^{dom(J)})^n$ ist I also eindeutig bestimmt. Dabei nennen wir J die zu I gehörende primitive Interpretation. Zu einem definierten Konzept B bezeichnet $index(B) \in \{1, \ldots, n\}$ die Position des Konzeptes B im Tupel $\underline{A}$. Die i-te Komponente des Tupels $\underline{A}$ wird mit $(\underline{A})_i$ bezeichnet. $\diamond$

[1] $A^I = D^I$ für alle Axiome $A = D$ in T.

Da Axiome notwendige und hinreichende Bedingungen an definierte Konzepte stellen, wird man erwarten, daß eine primitive Interpretation J zu einer T-Box T eindeutig zu einem T-Modell I erweitert werden kann. Dies stimmt für azyklische Terminologien[2], ist aber für zyklische Terminologien im allgemeinen falsch.

Beispiel 4.3.
Wir legen die Terminologie T aus Beispiel 3.3 zugrunde und definieren eine primitive Interpretation J wie folgt: $dom(J) = \{a, b, c, d, e, f, g, h, a_1, a_2, \ldots\}$, $\mathsf{Baum}^J := dom(J)$, $\mathsf{direkter\text{-}nachfolger}^J := \{(a,b), (a,c), (c,d), (e,f), (e,g), (f,h), (g,h), (h,h), (a_1,a_2), (a_2,a_3), (a_3,a_4), \ldots\}$.

Graphisch:



Für $index$(Binärbaum):= 1, $index$(Ternärbaum):= 2 und $index$(Blätter-Binärbaum):= 3 bilden z.B. die Tripel $(\{a,b,c,d\}, \{a,b,c,d\}, \{b,d\})$ sowie $(dom(J), dom(J), \{b,d\})$ jeweils zusammen mit J Modelle von T. ◇

Die primitive Interpretation J läßt sich im Beispiel also nicht eindeutig zu einem Modell von T erweitern. Möchte man bei der Definition von Binärbaum bzw. Ternärbaum auch zyklische Graphen oder unendliche Graphen zulassen, so würde als eindeutige Erweiterung von J das Tripel $(dom(J), dom(J), \{b,d\})$ bevorzugt werden. Sollen derartige Graphen jedoch nicht enthalten sein, so würde das Tripel $(\{a,b,c,d\}, \{a,b,c,d\}, \{b,d\})$ als eindeutige Erweiterung von J erwartet werden. Dieses Beispiel legt somit nahe, daß die deskriptive Semantik nicht immer die intuitive Bedeutung einer Terminologie widerspiegelt.

Das zu einem Modell gehörende Tupel der Extensionen definierter Konzepte ist ein Fixpunkt einer Abbildung, die wir nun formalisieren (vgl. [Baa96]). Dies gibt uns die Möglichkeit, neben der deskriptiven Semantik auch weitere Semantiken zu definieren.

Definition 4.4.
Es sei T eine Terminologie, die aus den Axiomen $A_1 = D_1, \ldots, A_n = D_n$ besteht, und J eine primitive Interpretation. Die Abbildung $T_J : (2^{dom(J)})^n \longrightarrow (2^{dom(J)})^n$ ist für ein Tupel $\underline{A} \in (2^{dom(J)})^n$ und die Interpretation I zu J und $\underline{A}$ folgendermaßen definiert: $T_J(\underline{A}) := (D_1^I, \ldots, D_n^I)$. ◇

Man sieht leicht, daß eine aus J und $\underline{A}$ bestehende Interpretation I genau dann Modell einer T-Box T ist, wenn $\underline{A}$ Fixpunkt der Abbildung T_J ist, d.h. $T_J(\underline{A}) = \underline{A}$ gilt. Nach Beispiel 2.1 ist $(2^{dom(J)})^n$ ein vollständiger Verband. Analog zu [Baa96], Proposition 16

[2] Durch sogenanntes „Auffalten“ einer azyklischen Terminologie erhält man eine äquivalente Terminologie, deren Axiome auf der rechten Seite keine definierten Konzepte enthalten. Damit ist eine primitive Interpretation eindeutig zu einem Modell der Terminologie erweiterbar (vgl. [Neb90a], Theorem 3.2).

zeigt man, daß T_J abwärts ω-stetig ist[3]. Insbesondere ist T_J damit monoton, so daß T_J (mindestens) einen Fixpunkt besitzt (vgl. Seite 12). Dies bedeutet, daß eine primitive Interpretation stets zu einem Modell einer T-Box erweitert werden kann. Damit können wir folgendes definieren:

Definition 4.5 (Semantiken (zyklischer) Terminologien).
Es sei T eine (zyklische) Terminologie.

i.) Die *deskriptive Semantik* erlaubt alle Modelle von T als zulässige Modelle.

ii.) Die *Kleinster-Fixpunkt-Semantik* (*lfp-Semantik*) erlaubt nur die Modelle von T als zulässige Modelle, die von einem kleinsten Fixpunkt einer Abbildung T_J stammen (*lfp-Modelle*).

iii.) Die *Größter-Fixpunkt-Semantik* (*gfp-Semantik*) erlaubt nur die Modelle von T als zulässige Modelle, die von einem größten Fixpunkt einer Abbildung T_J stammen (*gfp-Modelle*).

◇

Da bzgl. einer azyklischen Terminologie T jede primitive Interpretation J eindeutig zu einem Modell von T erweitert werden kann, d.h. T_J genau einen Fixpunkt besitzt, stimmen die drei Semantiken für diesen Fall überein. Für zyklische Terminologien läßt sich jede primitive Interpretation eindeutig zu einem lfp- bzw. gfp-Modell von T erweitern; dies gilt nach Beispiel 4.3 für die deskriptive Semantik jedoch nicht.

Da die Abbildung T_J für eine T-Box T und eine primitive Interpretation J abwärts ω-stetig ist, gilt nach Satz 2.3: $gfp(T_J) = \bigcap_{i\geq 0} T_J^i(top) = T_J{\downarrow}^\omega$ für $top = (dom(J))^n$. Satz 2.4 liefert $\underline{A}$-$gfp(T_J) = \bigcap_{i\geq 0} T_J^i(\underline{A})$ für $\underline{A} \subseteq (dom(J))^n$ und $\underline{A} \supseteq T_J(\underline{A})$. Für den kleinsten Fixpunkt $lfp(T_J)$ von T_J kann jedoch $lfp(T_J) \neq \bigcup_{i\geq 0} T_J^i(bottom) = T_J{\uparrow}^\omega$ gelten. Es sei z.B. die Terminologie T durch die Axiome $\mathsf{A} = \mathsf{Q}\sqcap\forall\mathsf{S}.\mathsf{B}$, $\mathsf{B} = \mathsf{P}\sqcap\forall\mathsf{R}.\mathsf{B}$ gegeben[4]. Weiter sei eine primitive Interpretation J dazu durch $dom(J) := \{a_0, a_1, a_2, \ldots\}$, $\mathsf{P}^J := \{a_1, a_2, a_3, \ldots\}$, $\mathsf{Q}^J := \{a_0\}$, $\mathsf{R}^J := \{(a_{i+1}, a_i); i \geq 1\}$ und $\mathsf{S}^J := \{(a_0, a_i); i \geq 1\}$ definiert. Für die Ordinalzahl $\omega + 1$ wird der kleinste Fixpunkt von T_J erreicht, nicht aber vorher, d.h. $lfp(T_J) = T_J{\uparrow}^{\omega+1} \neq T_J{\uparrow}^\omega$. Es gilt nämlich erst für ω die Gleichheit $(T_J{\uparrow}^\omega)_2 = \mathsf{P}^J$ ($index(\mathsf{B}) = 2$) und somit stimmt erst für $\omega+1$ die Menge $(T_J{\uparrow}^{\omega+1})_1$ mit Q^J ($index(\mathsf{A}) = 1$) überein.

Wegen $gfp(T_J) = \bigcap_{i\geq 0} T_J^i(top)$ und $lfp(T_J) = T_J{\uparrow}^\alpha$ für eine passende Ordinalzahl α (Satz 2.2) erhält man in Beispiel 4.3 das Tupel $\underline{A_{lfp}} = (\{a, b, c, d\}, \{a, b, c, d\}, \{b, d\})$ als kleinsten Fixpunkt und $\underline{A_{gfp}} = (dom(J), dom(J), \overline{\{b, d\}})$ als größten Fixpunkt von T_J und damit J zusammen mit $\underline{A_{lfp}}$ als lfp-Modell von T und J mit $\underline{A_{gfp}}$ als gfp-Modell von T. Bzgl. der lfp-Semantik gehören also die Individuen des zyklischen bzw. unendlichen Graphen nicht zum $\mathsf{Binärbaum}$ oder $\mathsf{Ternärbaum}$; bzgl. der gfp-Semantik gehören diese jedoch dazu.

Es stellt sich die Frage, welche Semantik gegenüber den anderen zu bevorzugen ist. Diese Frage wird auch in [Neb91] behandelt. Nebel kommt zu dem Schluß, daß keine der Semantiken grundsätzlich gegenüber den anderen ausgezeichnet ist. Auch die hier aufgeführten Beispiele legen dies nahe: Wie wir auf Seite 85 zeigen werden, folgt mit der

[3] In [Baa96] wurde dies für die Sprache $\mathcal{FL}_0$ gezeigt. Der Beweis für die Sprache $\mathcal{ALN}$ verläuft analog; primitive Negation und Zahlenrestriktion sind im Beweis wie primitive Konzepte zu behandeln.

[4] Beispiel aus [Baa96]

lfp-Semantik in Beispiel 3.2, daß das Konzept Mensch inkonsistent ist, d.h., daß es der betrachteten Terminologie folgend keine Menschen geben kann. Auf die Nachteile der deskriptiven Semantik sind wir in Beispiel 4.3 schon eingegangen. Führt man in Beispiel 3.2 analog zum Konzept Mensch ein Konzept Esel ein, so sind (vgl. Seite 79) die Konzepte Mensch und Esel bzgl. der gfp-Semantik äquivalent, d.h. alle Menschen sind Esel und umgekehrt.

Trotzdem sind in vielen Fällen die gfp-Semantik und die deskriptive Semantik zu bevorzugen. Für die gfp-Semantik bzgl. der Sprache $\mathcal{FL}_0$ hat sich in [Baa96] herausgestellt, daß diese eine natürlichere Charakterisierung durch endliche Automaten zuläßt. Dies wird für die Sprachen $\mathcal{AL}_0$ und $\mathcal{ALN}$ in dieser Arbeit bestätigt werden.

Die Berechnung der Subsumtions-Hierarchie, d.h. die Unterkonzept-Oberkonzept-Relation, die implizit in der T-Box definiert wird, ist ein wichtiger Dienst terminologischer Wissensrepräsentationssysteme. Die Subsumtion ist für die verschiedenen Semantiken folgendermaßen definiert:

Definition 4.6 (Subsumtion).
Es sei T eine Terminologie, A und B seien Konzeptnamen in T.

$$\begin{aligned}
A \sqsubseteq_T B \quad & \text{gdw.} \quad A^I \subseteq B^I \text{ für alle Modelle } I \text{ von } T. \\
A \sqsubseteq_{lfp,T} B \quad & \text{gdw.} \quad A^I \subseteq B^I \text{ für alle lfp-Modell } I \text{ von } T. \\
A \sqsubseteq_{gfp,T} B \quad & \text{gdw.} \quad A^I \subseteq B^I \text{ für alle gfp-Modell } I \text{ von } T.
\end{aligned}$$

Sprich: B subsumiert A in T (T-subsumiert) bzgl. der deskriptiven, lfp- bzw. gfp-Semantik. $\diamond$

Neben der Subsumtion ist auch die Inkonsistenz eines Konzeptes von Interesse.

Definition 4.7 (Inkonsistenz).
Es sei T eine Terminologie. Das Konzept A in T ist *inkonsistent* (*T-inkonsistent*) bzgl. der deskriptiven Semantik (lfp-, gfp-Semantik) gdw. für alle (lfp-, gfp-)Modelle I von T gilt: $A^I = \emptyset$. $\diamond$

Kapitel 5

$\mathcal{AL}_0$ und endliche Automaten

In [Neb87, Neb90a, Neb91] wurde zum erstenmal der Zusammenhang zwischen der terminologischen Sprache $\mathcal{FL}_0$ und endlichen Automaten beobachtet. Damit konnte die co-NP-Vollständigkeit der Subsumtion bzgl. azyklischer Terminologien bewiesen werden. Charakterisierungen der verschiedenen Semantiken und der Subsumtion für (zyklische) $\mathcal{FL}_0$-Terminologien wurden zuerst in [Baa96] angegeben[1]. Daraus konnten Entscheidungsalgorithmen für die Subsumtion abgeleitet werden.

In diesem Kapitel betrachten wir eine Erweiterung von $\mathcal{FL}_0$, die Sprache $\mathcal{AL}_0$. Die primitive Negation kann für alle drei Semantiken zu inkonsistenten Konzepten führen — in $\mathcal{FL}_0$ können nur bzgl. der lfp-Semantik Konzepte inkonsistent sein. Aus diesem Grund werden wir neben der Subsumtion auch die Inkonsistenz mit Hilfe von endlichen Automaten charakterisieren und daraus Entscheidungsalgorithmen sowie Komplexitätsaussagen ableiten.

Wie in der Einleitung erwähnt, führt die Inkonsistenz dazu, daß Konzepte von Wörtern „ausgeschlossen" werden können. Diese Wörter verhindern, daß bestimmte Individuen zur Extension eines Konzeptes gehören, was vor allem bei der Charakterisierung der Subsumtion zu berücksichtigen ist. Algorithmisch werden wir diese Wörter durch „Ausschlußzustände" beschreiben. Mit Hilfe dieser Zustände können aus den jeweiligen Charakterisierungen Entscheidungsalgorithmen für Inkonsistenz und Subsumtion gewonnen werden. Die Begriffe „Ausschluß" und „Ausschlußzustand" werden auch bei der Behandlung der Sprache $\mathcal{ALN}$ in Kapitel 6 von Bedeutung sein. Die Anwesenheit von Zahlenrestriktionen macht dann jedoch eine Anpassung dieser Begriffe nötig, erlaubt aber damit eine einfache Erweiterung der Entscheidungsalgorithmen von $\mathcal{AL}_0$ auf $\mathcal{ALN}$.

In Anlehnung an die Vorgehensweise in [Baa96], werden wir in diesem Kapitel für die Subsumtion außerdem Entscheidungsalgorithmen betrachten, die sich ausschließlich auf die Entscheidung der Inklusion regulärer und ω-regulärer Sprachen stützen. Dazu wird die Menge der Ausschlußwörter eines Konzeptes durch einen (in der Größe der Terminologie) polynomial großen endlichen Automaten beschrieben. Zahlenrestriktionen machen eine Charakterisierung der Menge der Ausschlußwörter eines Konzeptes durch derartige Automaten jedoch schwierig, so daß im 6. Kapitel nur der erstgenannte Ansatz über „Ausschlußzustände" verfolgt wird.

[1] Für die Sprache $\mathcal{FL}^-$, d.h. $\mathcal{FL}_0$ mit einfacher existentieller Restriktion $(\exists^{\geq 1} R)$, wurden auch gfp- und deskriptive Semantik untersucht.

5.1 Der Semi-Automat zur T-Box

Den Zusammenhang von Terminologie und Semi-Automat erhält man durch Zuordnung einer T-Box T zu einem Semi-Automaten $\mathcal{A}_T$. Dazu ist es zunächst nötig, die Terminologie in eine *Normalform* zu überführen:
Man sieht leicht, daß für Konzeptterme C und D sowie einen Rollennamen R die Konzeptterme $\forall R.(C \sqcap D)$ und $\forall R.C \sqcap \forall R.D$ äquivalent sind. Somit ist jeder Konzeptterm äquivalent zu einer Konjunktion von Termen der Form $\forall R_1.\forall R_2.\ldots.\forall R_n.C$ für C Konzeptname, primitive Negation oder Zahlenrestriktion. Wir schreiben für einen solchen Term kurz $\forall W.C$, wobei W das endliche Wort $R_1 \cdots R_n$ bezeichnet; $\forall \varepsilon.C$ steht für C. In einer *normalisierten Terminologie* ist jede rechte Seite eines Axioms also Konjunktion von Termen der Form $\forall W.C$. Für eine Interpretation I bezeichnet W^I die Komposition $R_1^I \circ R_2^I \circ \cdots \circ R_n^I$ der binären Relationen $R_1^I, \ldots, R_n^I$; es ist $\varepsilon^I := \{(d,d); d \in dom(I)\}$ die identische Relation. Damit gilt $(\forall R_1.\forall R_2.\ldots.\forall R_n.C)^I = (\forall W.C)^I$ für alle Interpretationen I. Die Normalform einer Terminologie ist eindeutig bis auf die Anordnung der Terme $\forall W.C$ in den Konjunktionen. Der nun zu definierende Automat $\mathcal{A}_T$ ist invariant unter dieser Anordnung. Deshalb können wir $\mathcal{A}_T$ wie folgt definieren:

Definition 5.1 (der Semi-Automat $\mathcal{A}_T$).
Es sei T eine (o.E., siehe oben) normalisierte $\mathcal{ALN}$-Terminologie. Der zugehörige Semi-Automat (mit Worttransitionen) $\mathcal{A}_T = (\Sigma, Q, E)$ ist folgendermaßen definiert: Das Alphabet Σ von $\mathcal{A}_T$ ist die Menge der in T vorkommenden Rollennamen; die Zustände von $\mathcal{A}_T$ (Q) sind die in T vorkommenden Konzeptnamen, primitiven Negationen (Terme der Form $\neg P$ in T für ein primitives Konzept P) und Zahlenrestriktionen; jedes Axiom $A = \forall W_1.A_1 \sqcap \cdots \sqcap \forall W_n.A_n$ in T liefert die Transitionen $(A, W_1, A_1), \ldots, (A, W_n, A_n) \in E$ in $\mathcal{A}_T$. ◇

Offensichtlich ist $\mathcal{A}_T$ mit linearem Aufwand aus T konstruierbar.

Bemerkung 5.2.
Aus $\mathcal{A}_T$ läßt sich gemäß Lemma 2.7 — mit linearem Aufwand — ein Semi-Automat $\mathcal{A}_T{}'$ ohne Worttransitionen konstruieren, so daß die eingeführten Sprachen (genauer, vgl. Lemma 2.7) für $\mathcal{A}_T$ und $\mathcal{A}_T{}'$ jeweils übereinstimmen. Wir werden deshalb, vor allem bei den Entscheidungsalgorithmen für die Subsumtion, (o.E.) von $\mathcal{A}_T$ als einem Semi-Automaten ohne Worttransitionen ausgehen können.

Ist T' die zu $\mathcal{A}_T{}'$ gehörende normalisierte Terminologie, so entspricht der Konstruktion von $\mathcal{A}_T{}'$ die folgende für T': Jeder Konzeptterm der Form $\forall RW.C$ in T ($R \in \Sigma$ und $W \in \Sigma^+$) wird durch $\forall R.A$ ersetzt, wobei A jeweils ein neues definiertes Konzept ist, für welches das Axiom $A = \forall W.C$ eingeführt wird. Diese Ersetzung wird solange iteriert bis für jeden Konzeptterm $\forall W.C$ der erhaltenen Terminologie gilt, daß $W = \varepsilon$ oder $W = R$ für $R \in \Sigma$ ist. ◇

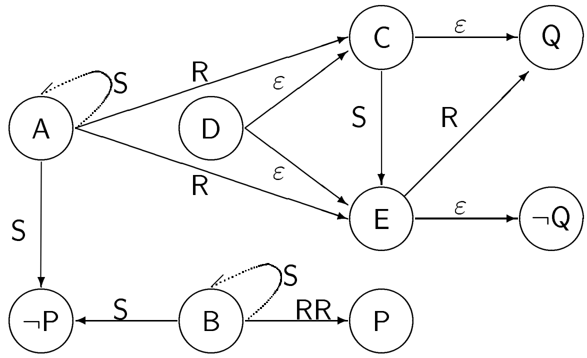
Im nächsten Beispiel werden wir eine konkrete normalisierte Terminologie mit zugehörigem Semi-Automaten angeben.

Beispiel 5.3.
Die Terminologie T sei durch folgende Axiome gegeben:

$$\mathsf{A} = \forall \mathsf{R.C} \sqcap \forall \mathsf{R.E} \sqcap \forall \mathsf{S.}(\neg \mathsf{P}) \sqcap \forall \mathsf{S.A}$$
$$\mathsf{B} = \forall \mathsf{RR.P} \sqcap \forall \mathsf{S.}(\neg \mathsf{P}) \sqcap \forall \mathsf{S.B}$$

$$\mathsf{C} = \forall \mathsf{S}.\mathsf{E} \sqcap \mathsf{Q}$$
$$\mathsf{D} = \mathsf{C} \sqcap \mathsf{E}$$
$$\mathsf{E} = \forall \mathsf{R}.\mathsf{Q} \sqcap \neg\mathsf{Q}.$$

A, B, C, D und E sind definierte Konzepte; P und Q sind primitive Konzepte; R und S Rollen. Wir geben $\mathcal{A}_T$ graphisch wie folgt an:



Die Zustände, von denen keine Transitionen ausgehen, sind primitive Konzepte, primitive Negationen oder (für $\mathcal{ALN}$) Zahlenrestriktionen. $\diamond$

In den folgenden Abschnitten werden wir nun für jede Semantik bzgl. $\mathcal{AL}_0$ mit Hilfe von $\mathcal{A}_T$ die jeweilige Semantik, die Inkonsistenz sowie die Subsumtion charakterisieren und für Inkonsistenz und Subsumtion daraus Entscheidungsalgorithmen und Komplexitätsaussagen ableiten.

Es sei an dieser Stelle angemerkt, daß wir im folgenden — auch im Kapitel 6 — einige Begriffe und Bezeichnungen wie „Ausschluß“, „Ausschlußzustand“, „inkonsistent“, „subsumiert“, „E_A“, „$E_{A,\omega}$“, „kanonisches Modell“ und „erweitertes kanonisches Modell“ einführen bzw. benutzen, die für verschiedene Semantiken und Sprachen gleich benannt werden, aber je nach Semantik und Sprache unterschiedliche Bedeutung haben. Bezugssemantik und -sprache werden jedoch jeweils aus dem Kontext ersichtlich sein, so daß auf zugehörige Sprache und Semantik nicht (immer) explizit hingewiesen wird.

5.2 Charakterisierung der gfp-Semantik in $\mathcal{AL}_0$

Anders als für $\mathcal{FL}_0$ können für $\mathcal{AL}_0$ Konzepte bzgl. der gfp-Semantik inkonsistent sein. Wir werden deshalb neben der Charakterisierung der Subsumtion bzgl. der gfp-Semantik in $\mathcal{AL}_0$ auch die Inkonsistenz charakterisieren. Desweiteren verhindert die Inkonsistenz eine direkte Übertragung der Charakterisierung der Subsumtion von $\mathcal{FL}_0$ auf $\mathcal{AL}_0$. Die Charakterisierung der gfp-Semantik in $\mathcal{FL}_0$ läßt sich jedoch leicht auf die Sprache $\mathcal{AL}_0$ erweitern.

Satz 5.4 (Charakterisierung der gfp-Semantik bzgl. $\mathcal{AL}_0$).
Es bezeichne T eine $\mathcal{AL}_0$-Terminologie, $\mathcal{A}_T$ den zugehörigen Semi-Automaten, I ein gfp-Modell von T sowie A einen Konzeptnamen in T. Für jedes $d \in dom(I)$ gilt:
$d \in A^I$ gdw.

(P1) für alle primitiven Konzepte P in T, alle Wörter $W \in L(A, P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in P^I$; und

(P2) für alle Terme $\neg P$ in T, alle Wörter $W \in L(A, \neg P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\neg P)^I$.

Beweis: Der Beweis kann analog zum Beweis von Proposition 19 in [Baa96] geführt werden. Die primitive Negation $\neg P$ ist im Beweis wie ein primitives Konzept zu behandeln. $\square$

Die Sprachen $L(A, P)$ und $L(A, \neg P)$ für ein primitives Konzept P stellen eine Menge von Bedingungen $\forall W.P$ bzw. $\forall W.\neg P$ dar, die ein Individuum der Extension von A erfüllen muß, d.h. d liegt genau dann in A^I, wenn für alle Wörter W und primitiven Konzepte P mit $W \in L(A, P)$ bzw. $W \in L(A, \neg P)$ gilt: $d \in (\forall W.P)^I$ bzw. $d \in (\forall W.\neg P)^I$. Dabei hängt diese Bedingung nur von der zu I gehörenden primitiven Interpretation ab.

Ist I ein gfp-Modell zur Terminologie T aus Beispiel 5.3, so gilt für ein Individuum $d \in dom(I)$ wegen $L(B, Q) = L(B, \neg Q) = \emptyset$ nach Satz 5.4:

$$\begin{aligned} d \in \mathsf{B}^I \quad \text{gdw.} \quad & d \in (\forall W.\neg P)^I \text{ für alle } W \in \mathsf{S}^*\mathsf{S} = L(B, \neg P) \text{ und} \\ & d \in (\forall W.P)^I \text{ für alle } W \in \mathsf{S}^*\mathsf{RR} = L(B, P). \end{aligned}$$

Für ein Individuum $d \in B^I$ ist also notwendig, daß alle S-Nachfolger (auch indirekten) von d in $(\neg P)^I$ liegen. Zyklische Terminologien erlauben also — wie schon in den Beispielen 3.2 und 3.3 gesehen — Aussagen über transitive Hüllen von Rollen.

5.2.1 Die gfp-Semantik in $\mathcal{AL}_0$ — Inkonsistenz

In einer Terminologie mit Axiom $A = P \sqcap \neg P$ ist A offensichtlich inkonsistent. Auch D in Beispiel 5.3 ist inkonsistent. Ist nämlich I ein gfp-Modell von T, so gilt für ein Individuum $d \in \mathsf{D}^I$ nach Satz 5.4 wegen $\varepsilon \in L(\mathsf{A}, \mathsf{Q}) \cap L(\mathsf{A}, \neg\mathsf{Q})$ und $d\varepsilon^I d$, daß $d \in \mathsf{Q}^I$ und $d \in (\neg\mathsf{Q})^I$ gelten muß. Da dies eine widersprüchliche Bedingung ist, kann D kein Individuum enthalten. Die Existenz eines primitiven Konzeptes P mit $\varepsilon \in L(A, P) \cap L(A, \neg P)$ ist nicht nur hinreichend, sondern auch notwendig für die Inkonsistenz von A. Für den Beweis dieser Aussage ist die folgende Definition nützlich.

Definition 5.5 (kanonisches Modell bzgl. der gfp-Semantik in $\mathcal{AL}_0$).
Es sei T eine $\mathcal{AL}_0$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T. Die *kanonische primitive Interpretation* $J = J(A, d_0)$ zum Konzept A und Individuum d_0 ist wie folgt definiert:
$dom(J) := \{d_0\}$; $R^J := \emptyset$ für alle Rollennamen R aus T; für jedes primitive Konzept P in T gelte $d_0 \in P^J$ gdw. $\varepsilon \in L(A, P)$.

Das *kanonische Modell* $I = I(A, d_o)$ zum Konzept A und Individuum d_0 ist das zu J und T gehörende gfp-Modell[2]. $\diamond$

Um für das kanonische Modell $I = I(A, d_0)$ die Aussage $d_0 \in A^I$ beweisen zu können, ist folgende Bedingung hinreichend[3]:

$$\text{Es existiert kein primitives Konzept } P \text{ mit } \varepsilon \in L(A, P) \cap L(A, \neg P). \tag{5.1}$$

Dies zeigen wir in

[2]Der Bezug von J und I zu T wird stets aus dem Kontext hervorgehen.
[3]und nach Satz 5.8 auch notwendig

Lemma 5.6.
Ist Bedingung (5.1) erfüllt, so gilt: $d_0 \in A^I$.

Beweis: Wir zeigen die Gültigkeit von (P1) und (P2) in Satz 5.4 bzgl. A und d_0, woraus $d_0 \in A^I$ folgt.

Es sei P ein primitives Konzept in T, $W \in L(A,P)$ und $e \in dom(I)$ mit $d_0 W^I e$. Wegen $R^I = \emptyset$ für alle R folgt $W = \varepsilon$ und $e = d_0$. Wegen $\varepsilon \in L(A,P)$ folgt $d_0 \in P^J$ nach Definition von J. Dies zeigt (P1).

Für $W \in L(A,\neg P)$ und $e \in dom(I)$ mit $d_0 W^I e$ folgt wiederum $W = \varepsilon$ und $e = d_0$. Zusammen mit $\varepsilon \in L(A,\neg P)$ liefert $\varepsilon \notin L(A,\neg P) \cap L(A,P)$ auch $\varepsilon \notin L(A,P)$. Die Definition von J impliziert somit $d_0 \notin P^J$, also $d_0 \in (\neg P)^J$. Dies zeigt (P2). □

Für eine algorithmische Charakterisierung der Inkonsistenz definieren wir den Begriff des Ausschlußzustandes. Dieser wird auch für die Subsumtion von Bedeutung sein.

Definition 5.7 (Ausschlußzustand bzgl. der gfp-Semantik in $\mathcal{AL}_0$).
Es bezeichne T eine Terminologie und $\mathcal{A}_T = (\Sigma, Q, E)$ den zugehörigen Semi-Automaten. Dann heißt $F \subseteq Q$ *Ausschlußzustand* bzgl. $\mathcal{A}_T$ (und bzgl. der gfp-Semantik in $\mathcal{AL}_0$), falls ein primitives Konzept P in T existiert mit $P, \neg P \in F$. ◇

Wir sprechen nicht von einer Ausschlußzustandsmenge bzgl. $\mathcal{A}_T$, sondern von einem Ausschlußzustand, da ein Ausschlußzustand einen Zustand in dem zu $\mathcal{A}_T$ gehörenden Potenzmengenautomaten beschreibt (vgl. Ausführungen im Anschluß an Lemma 5.15), der auch den Mengen ε-*closure* und $next_\varepsilon$ implizit zugrunde liegt.

Offensichtlich kann mit linearem Zeitaufwand entschieden werden, ob eine Teilmenge von Q ein Ausschlußzustand ist.

Die Inkonsistenz eines Konzeptes kann nun wie folgt charakterisiert werden:

Satz 5.8 (Charakterisierung der Inkonsistenz bzgl. der gfp-Semantik in $\mathcal{AL}_0$).
Es sei T eine Terminologie in $\mathcal{AL}_0$, $\mathcal{A}_T$ der zugehörige Semi-Automat und A ein Konzept in T. Folgende Aussagen sind äquivalent:

1.) A ist T-inkonsistent bzgl. der gfp-Semantik in $\mathcal{AL}_0$.

2.) Es existiert ein primitives Konzept P in T mit $\varepsilon \in L(A,P) \cap L(A,\neg P)$.

3.) Die Zustandsmenge ε-*closure*$(\{A\})$ ist ein Ausschlußzustand.

Beweis: Die Äquivalenz der letzten beiden Aussagen sollte nach Definition 5.7 und der Definition von „ε-*closure*“ klar sein. Wir zeigen die Äquivalenz der ersten beiden Aussagen:
„1. $\Rightarrow$ 2.“: Existiert kein primitives Konzept P in T mit $\varepsilon \in L(A,P) \cap L(A,\neg P)$, so existiert zu A und einem Individuum d_0 nach Lemma 5.6 das kanonische Modell $I = I(A, d_0)$ mit $d_0 \in A^I$; damit ist A konsistent.
„1. $\Leftarrow$ 2.“: Es existiere ein primitives Konzept P in T mit $\varepsilon \in L(A,P) \cap L(A,\neg P)$. Es sei I ein gfp-Modell von T sowie d ein Individuum in I mit $d \in A^I$. Satz 5.4, (P1) liefert dann $d \in P^I$ zusammen mit $d \varepsilon^I d$. Entsprechend impliziert (P2): $d \in (\neg P)^I$. Dies ist ein Widerspruch und somit ist A inkonsistent. □

Da die Berechnung von ε-*closure*$(\{A\})$ und der Test auf Ausschlußzustand nur polynomialen Zeitaufwand benötigen, ist die Inkonsistenz eines Konzeptes bzgl. der gfp-Semantik in $\mathcal{AL}_0$ mit polynomialem Zeitaufwand entscheidbar.

In Beispiel 5.3 ist ε-*closure*$(\{\mathsf{D}\}) = \{\mathsf{D}, \mathsf{C}, \mathsf{E}, \mathsf{Q}, \neg\mathsf{Q}\}$. Da Q und $\neg\mathsf{Q}$ in dieser Menge enthalten sind, folgt die Inkonsistenz von D.

5.2.2 Die gfp-Semantik in $\mathcal{AL}_0$ — Subsumtion

In [Baa96], Theorem 20 wurde für eine $\mathcal{FL}_0$-Terminologie T die Subsumtion zweier Konzepte A und B wie folgt charakterisiert:

$$A \sqsubseteq_{gfp,T} B \text{ gdw. } L(B,P) \subseteq L(A,P) \text{ für alle primitiven Konzepte } P \text{ in } T. \tag{5.2}$$

Für die Erweiterung der Charakterisierung auf $\mathcal{AL}_0$ reicht die Ergänzung der rechten Seite um die Bedingung $L(B,\neg P) \subseteq L(A,\neg P)$ für alle Terme der Form $\neg P$ in T nicht aus. D.h. im allgemeinen ist die Aussage

$$\begin{aligned} A \sqsubseteq_{gfp,T} B \text{ gdw. } & L(B,P) \subseteq L(A,P) \text{ für alle primitiven Konzepte } P \text{ in } T \text{ und} \\ & L(B,\neg P) \subseteq L(A,\neg P) \text{ für alle Terme der Form } \neg P \text{ in } T \end{aligned} \tag{5.3}$$

falsch. Mit Satz 5.4 sieht man leicht, daß damit jedoch die rechte Seite von (5.3) eine hinreichende Bedingung für $A \sqsubseteq_{gfp,T} B$ ist, da alle Bedingungen ($d \in (\forall W.P)^I$ bzw. $d \in (\forall W.\neg P)^I$), die an ein Individuum aus B^I gestellt werden, auch an Individuen aus A^I gestellt werden.

Aufgrund der Inkonsistenz von Konzepten, die bei $\mathcal{FL}_0$ bzgl. der gfp-Semantik nicht auftrat, ist die rechte Seite aber keine notwendige Bedingung für die Subsumtion. Wir verdeutlichen dies am Beispiel 5.3. Da D inkonsistent ist, gilt im Beispiel die Subsumtionsbeziehung $\mathsf{D} \sqsubseteq_{gfp,T} \mathsf{B}$, obwohl $\mathsf{RR} \in L(\mathsf{B},\mathsf{P})$ und $\mathsf{RR} \notin L(\mathsf{D},\mathsf{P})$, also $L(\mathsf{B},\mathsf{P}) \not\subseteq L(\mathsf{D},\mathsf{P})$. Desweiteren gilt $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$, obwohl $\mathsf{WRR} \in L(\mathsf{B},\mathsf{P})$ und $\mathsf{WRR} \notin L(\mathsf{A},\mathsf{P})$ für ein beliebiges Wort $\mathsf{W} \in \mathsf{S}^*$ gilt. Ist nämlich I ein gfp-Modell von T und $d \in dom(I)$ mit $d \in \mathsf{A}^I$, so besitzt d keine WR-Nachfolger. Wegen $\mathsf{WR} \in L(\mathsf{A},\mathsf{Q}) \cap L(\mathsf{A},\neg\mathsf{Q})$ müßten diese nach Satz 5.4 sonst Elemente der Extensionen zu Q und $\neg\mathsf{Q}$ sein; dies ist aber ein Widerspruch. Damit gilt $d \in (\forall\mathsf{WRR.P})^I$. Zusammen mit $L(A,\neg\mathsf{P}) = L(B,\neg\mathsf{P})$, $L(\mathsf{B},\mathsf{Q}) = L(\mathsf{B},\neg\mathsf{Q}) = \emptyset$ und $d \in \mathsf{A}^I$ folgt dann mit Satz 5.4 leicht: $d \in \mathsf{B}^I$.

Woran liegt es also, daß $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$ gilt, obwohl die rechte Seite der Äquivalenzaussage (5.3) nicht erfüllt ist? Ist $\mathsf{A} \not\sqsubseteq_{gfp,T} \mathsf{B}$, so existiert ein gfp-Modell I und ein Individuum d_0 mit $d_0 \in \mathsf{A}^I$ und $d_0 \notin \mathsf{B}^I$. Bis auf die Inklusion von $L(\mathsf{B},\mathsf{P})$ und $L(\mathsf{A},\mathsf{P})$ gelten die Inklusionen zu $\neg\mathsf{P}$, Q und $\neg\mathsf{Q}$. Wegen $d_0 \in \mathsf{A}^I$ und $d_0 \notin \mathsf{B}^I$ muß deshalb zu d_0 ein WRR-Nachfolger existieren, der nicht in P^I liegt. Dann hätte aber d_0 auch einen WR-Nachfolger, der, wie erwähnt, in der Extension von Q und $\neg\mathsf{Q}$ liegen müßte. Dies kann aber nicht sein. Es ist also nicht möglich, ein gfp-Modell zu T anzugeben, welches die Subsumtion von A und B widerlegt. Der Grund sind die Wörter WR, die derartige Modelle nicht zulassen. Diese Wörter schließen A aus. Der Begriff „Ausschluß“ ist in dieser Arbeit ein zentraler Begriff, der auch in den folgenden Abschnitten und im folgenden Kapitel eine wichtige Rolle bei der Charakterisierung der Subsumtion spielen wird. Nun werden wir diesen formal definieren.

Definition 5.9 (Ausschluß bzgl. der gfp-Semantik in $\mathcal{AL}_0$).
Es sei T eine Terminologie, $\mathcal{A}_T = (\Sigma, Q, E)$ der zugehörige Semi-Automat sowie A ein Konzept in T. Das Wort $W \in \Sigma^* \cup \Sigma^\omega$ *schließt A aus*[4], falls ein primitives Konzept P in T existiert und ein endliches Präfix V von W mit $V \in L(A,P) \cap L(A,\neg P)$. ◇

Das folgende Lemma macht die Bedeutung dieser Definition deutlich.

[4] Der Fall $W \in \Sigma^\omega$ wird erst im nächsten Abschnitt benötigt.

Lemma 5.10.
Ist I ein gfp-Modell zur Terminologie T sowie W ein A-ausschließendes Wort, und sind $d, f \in dom(I)$ Individuen mit $dW^I f$, dann gilt $d \notin A^I$.

Beweis: Wegen $dW^I f$ existiert ein Individuum $e \in dom(I)$ mit $dV^I e$ (V wie in Definition 5.9 gewählt). Nach Voraussetzung gilt $V \in L(A, P) \cap L(A, \neg P)$. Gilt $d \in A^I$, so folgt nach Satz 5.4: $e \in P^I$ und $e \in (\neg P)^I$; also kann d kein Element von A^I sein. □

Es sei $E_A := \{W \in \Sigma^*; W \text{ schließt } A \text{ aus}\}$ die Menge der A-ausschließenden Wörter; auch hier geht jeweils der Bezug zur Terminologie, zur Semantik sowie zur Sprache aus dem Kontext hervor. Da mit einem A-ausschließenden Wort W auch alle Wörter A ausschließen, die W als Präfix enthalten, ist für ein inkonsistentes Konzept A wegen $\varepsilon \in E_A$ (Satz 5.8) die Menge E_A gleich Σ^*.

In Beispiel 5.3 ist $E_\mathsf{A} = \mathsf{S}^*\mathsf{R}\Sigma^*$. Wie erwähnt gilt die Inklusion zwischen $L(\mathsf{B}, \mathsf{P})$ und $L(\mathsf{A}, \mathsf{P})$ nicht, da $L(\mathsf{B}, \mathsf{P})$ A-ausschließende Wörter enthält, die in $L(\mathsf{A}, \mathsf{P})$ nicht enthalten sind. Es gilt aber $L(\mathsf{B}, \mathsf{P}) \subseteq L(\mathsf{A}, \mathsf{P}) \cup E_\mathsf{A}$. Diese Inklusion ist der Grund dafür, daß kein gfp-Modell existiert, welches die Subsumtion von A und B widerlegt. Wir werden im folgenden sehen, daß ein solches Modell existiert, falls eine derartige Inklusion nicht gilt.

Definition 5.11 (erweitertes kanonisches gfp-Modell bzgl. $\mathcal{AL}_0$).
Es sei T eine Terminologie in $\mathcal{AL}_0$, $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T und $W = R_1 \cdots R_n$ ein endliches Wort über Σ. Weiter sei $I = I(A, d_0)$ das kanonische Modell zu A und Individuum d_0 mit zugehöriger primitiver Interpretation J. Wir definieren nun die Erweiterungen $J_{min} = J_{min}(A, d_0, W)$ und $J_{max} = J_{max}(A, d_0, W)$ zu J wie folgt: $dom(J_{min}) = dom(J_{max}) := \{d_0, \ldots, d_n\}$ und $R^{J_{min}} = R^{J_{max}} := \{(d_{i-1}, d_i); 1 \le i \le n$ und $R = R_i\}$ für alle Rollennamen R in T. Für alle primitiven Konzepte P in T und alle $d \in dom(J_{min}) = dom(J_{max})$ sei weiter:

$$
\begin{array}{ll}
d \in P^{J_{min}} & \text{gdw. ein } V \in \Sigma^* \text{ mit } V \in L(A, P) \text{ und } d_0 V^{J_{min}} d \text{ existiert; und} \\
d \in (\neg P)^{J_{max}} & \text{gdw. ein } V \in \Sigma^* \text{ mit } V \in L(A, \neg P) \text{ und } d_0 V^{J_{max}} d \text{ existiert.}
\end{array}
$$

Die *erweiterten kanonischen gfp-Modelle* $I_{min}(A, d_0, W)$ und $I_{max}(A, d_0, W)$ sind nun die zu $J_{min}(A, d_0, W)$ bzw. $J_{max}(A, d_0, W)$ gehörenden gfp-Modelle zu T. ◇

Die primitiven Interpretationen J_{min} und J_{max} unterscheiden sich dadurch, daß in J_{min} die Extension eines primitiven Konzeptes P nur die Individuen enthält, die nach Satz 5.4, (P1) enthalten sein müssen. Bzgl. J_{max} enthält dagegen die Extension zu P die Individuen, die nach Satz 5.4, (P2) nicht zwingend in der Extension zu $\neg P$ liegen müssen.

Um nun für die erweiterten kanonischen gfp-Modelle $I_{min} = I_{min}(A, d_0, W)$ bzw. $I_{max} = I_{max}(A, d_0, W)$ die Aussagen $d_0 \in A^{I_{min}}$ bzw. $d_0 \in A^{I_{max}}$ zeigen zu können, ist die folgende Bedingung hinreichend[5]:

Das Konzept A ist konsistent und wird vom endlichen Wort W nicht ausgeschlossen. (5.4)

Das nächste Lemma faßt Eigenschaften der erweiterten kanonischen primitiven Interpretationen bzw. der erweiterten kanonischen Modelle zusammen.

Lemma 5.12 (Eigenschaften von J_{min} und I_{min} bzw. J_{max} und I_{max}).
Mit den Bezeichnungen von Definition 5.11 gelten die folgenden Aussagen:

[5] und, wie man mit Satz 5.8 und Lemma 5.10 leicht zeigt, auch notwendig

1.) $d_0 W^{J_{min}} d_n$ und $d_0 W^{J_{max}} d_n$;

2.) zu jedem Individuum $d \in dom(J_{min}) = dom(J_{max})$ existiert eindeutig ein V, welches Präfix von W ist und für das $d_0 V^{J_{min}} d$ (bzw. $d_0 V^{J_{max}} d$) gilt;

3.) ist $V \notin L(A, P)$ und $d_0 V^{J_{min}} d$ für ein Wort V, ein primitives Konzept P und ein Individuum d, dann gilt $d \notin P^{J_{min}}$;

4.) ist $V \notin L(A, \neg P)$ und $d_0 V^{J_{max}} d$ für ein Wort V, ein primitives Konzept P und ein Individuum d, dann gilt $d \notin (\neg P)^{J_{max}}$;

5.) ist Bedingung (5.4) erfüllt, so sind (P1) und (P2) von Satz 5.4 bzgl. A, d_0 und J_{min} bzw. J_{max} gültig; insbesondere gilt für die zugehörigen gfp-Modelle: $d_0 \in A^{I_{min}}$ bzw. $d_0 \in A^{I_{max}}$.

Beweis: Die Punkte 1.) und 2.) folgen direkt aus Definition 5.11. Die Eigenschaften 3.) und 4.) folgen leicht aus 2.) und den Definitionen zu J_{min} bzw. J_{max}.

Zu 5.): Es gelte Bedingung (5.4). Eigenschaft (P1) gilt nach Definition von J_{min} für A und d_0. Es sei nun $V \in L(A, \neg P)$ für ein primitives Konzept P und $d_0 V^{J_{min}} d$ für ein $d \in dom(J_{min})$. Nach 2.) ist V Präfix von W. Da A von W nicht ausgeschlossen wird, gilt $V \notin L(A, P)$. Aus 3.) folgt somit $d \notin P^{J_{min}}$, also $d \in (\neg P)^{J_{min}}$. Nach Definition von J_{max} gilt (P2) für A und d_0. Man zeigt analog zu J_{min} und (P2) mit Hilfe von 4.), daß (P1) für A, d_0 und J_{max} gilt. □

Wir können nun den folgenden Satz zeigen:

Satz 5.13 (Charakterisierung: Subsumtion bzgl. gfp-Semantik in $\mathcal{AL}_0$).
Es bezeichne T eine $\mathcal{AL}_0$-Terminologie sowie $\mathcal{A}_T$ den zugehörigen Semi-Automaten, A und B bezeichnen Konzepte in T. Es gilt $A \sqsubseteq_{gfp,T} B$ gdw.

1.) $L(B, P) \subseteq L(A, P) \cup E_A$ für alle primitiven Konzepte P in T; und

2.) $L(B, \neg P) \subseteq L(A, \neg P) \cup E_A$ für alle Terme der Form $\neg P$ in T.

Beweis: „$\Rightarrow$": Wir nehmen an, daß eine der Inklusionen nicht gilt und zeigen $A \not\sqsubseteq_{gfp,T} B$.

(1) Annahme: $L(B, P) \not\subseteq L(A, P) \cup E_A$ für ein primitives Konzept P.
Es existiert demnach ein Wort $W \in L(B, P) \setminus (L(A, P) \cup E_A)$. Wäre A inkonsistent, so wäre $E_A = \Sigma^*$, im Widerspruch zu $W \notin E_A$. Das Konzept A ist also konsistent. Wegen $W \notin E_A$ existiert somit nach Lemma 5.12, 5.) das erweiterte kanonische Modell $I_{min} = I_{min}(A, d_0, W)$ mit $d_0 \in A^{I_{min}}$ und $d_0 W^{I_{min}} d$ für ein $d \in dom(I_{min})$. Aus Lemma 5.12, 3.) folgt wegen $W \notin L(A, P)$: $d \notin P^{I_{min}}$. Damit impliziert $W \in L(B, P)$ zusammen mit (P1) von Satz 5.4 für B und d_0: $d_0 \notin B^{I_{min}}$. Dies zeigt $A \not\sqsubseteq_{gfp,T} B$.

(2) Annahme: $L(B, \neg P) \not\subseteq L(A, \neg P) \cup E_A$ für einen Term $\neg P$ in T.
Es existiert also ein Wort $W \in \Sigma^*$ mit $W \in L(B, \neg P) \setminus (L(A, \neg P) \cup E_A)$. Man schließt wie in (1) die Existenz des erweiterten kanonischen Modells $I_{max} = I_{max}(A, d_0, W)$ mit $d_0 \in A^{I_{max}}$ und $d_0 W^{I_{max}} d$ für ein $d \in dom(I_{max})$. Wegen $W \notin L(A, \neg P)$ und Lemma 5.12, 4.) folgt $d \notin (\neg P)^{I_{max}}$, was zusammen mit $W \in L(B, \neg P)$ nach Satz 5.4, (P2) bzgl. B und d_0 die Aussage $d_0 \notin B^{I_{max}}$ liefert. Damit gilt $A \not\sqsubseteq_{gfp,T} B$.

„$\Leftarrow$" Es gelte die rechte Seite der Behauptung. Wir nehmen $A \not\sqsubseteq_{gfp,T} B$ an. Damit existiert ein gfp-Modell I zu T und ein Individuum $d_0 \in dom(I)$ mit $d_0 \in A^I \setminus B^I$. Wegen $d_0 \notin B^I$ gelten also (P1) oder (P2) nicht.

(3) Gilt (P1) nicht, so existiert ein primitives Konzept P, ein Wort $W \in L(B,P)$ und ein Individuum $e \in dom(I)$ mit $d_0 W^I e$ und $e \notin P^I$. Wegen $L(B,P) \subseteq L(A,P) \cup E_A$ gilt $W \in L(A,P)$ oder $W \in E_A$. Im Fall $W \in L(A,P)$ folgt mit (P1) für A und d_0 sofort $d_0 \notin A^I$, im Widerspruch zur Annahme. Im Fall $W \in E_A$ folgt nach Lemma 5.10 und mit $d_0 W^I e$ direkt $d_0 \notin A^I$, im Widerspruch zur Annahme.

(4) Gilt (P2) nicht, so existiert ein primitives Konzept P, ein $W \in L(B,\neg P)$ sowie ein Individuum $e \in dom(I)$ mit $d_0 W^I e$ und $e \notin (\neg P)^I$. Man folgert analog zu (3): $d_0 \notin A^I$, im Widerspruch zur Annahme. □

In Beispiel 5.3 ist $E_\mathsf{A} = \mathsf{S}^*\mathsf{R}\Sigma^*$, $L(\mathsf{B},\neg\mathsf{P}) = L(\mathsf{A},\neg\mathsf{P})$, $L(\mathsf{B},\mathsf{P}) = \mathsf{S}^*\mathsf{RR}$ und $L(\mathsf{B},\mathsf{Q}) = L(\mathsf{B},\neg\mathsf{Q}) = \emptyset$; damit ist $L(\mathsf{B},\mathsf{Q}) \subseteq L(\mathsf{A},\mathsf{Q}) \cup E_\mathsf{A}$, $L(\mathsf{B},\neg\mathsf{Q}) \subseteq L(\mathsf{A},\neg\mathsf{Q}) \cup E_\mathsf{A}$, $L(\mathsf{B},\mathsf{P}) \subseteq L(\mathsf{A},\mathsf{P}) \cup E_\mathsf{A}$ sowie $L(\mathsf{B},\neg\mathsf{P}) \subseteq L(\mathsf{A},\neg\mathsf{P}) \cup E_\mathsf{A}$. Satz 5.13 impliziert also $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$.

Satz 5.13 ermöglicht es, einen Entscheidungsalgorithmus für die Subsumtion bzgl. der gfp-Semantik in $\mathcal{AL}_0$ zu formulieren. Dazu werden wir die Menge E_A mit Hilfe der eingeführten Ausschlußzustände charakterisieren. Zunächst aber

Definition 5.14 (Erreichen eines Ausschlußzustandes).
Es sei T eine Terminologie und $\mathcal{A}_T$ der zugehörige Semi-Automat ohne Worttransitionen (vgl. Bemerkung 5.2). Mit einem Wort $W \in \Sigma^* \cup \Sigma^\omega$ *erreicht* man von einem Konzept A aus *einen Ausschlußzustand*, falls ein endliches Präfix V von W existiert, so daß $next_\varepsilon(A,V)$ ein Ausschlußzustand ist[6]. ◇

Eine Charakterisierung von E_A liefert nun

Lemma 5.15.
Für eine Terminologie T, den zugehörigen Semi-Automaten $\mathcal{A}_T$ ohne Worttransitionen und ein Konzept A in T gilt: $E_A = \{W \in \Sigma^*;$ mit W erreicht man von A aus einen Ausschlußzustand$\}$.

Beweis: „$\subseteq$“: Es sei $W \in E_A$, d.h. W schließt A aus. Dann existiert ein Präfix V von W und ein primitives Konzept P mit $V \in L(A,P) \cap L(A,\neg P)$. Damit gilt aber $P, \neg P \in next_\varepsilon(A,V)$ nach Lemma 2.5, also ist $next_\varepsilon(A,V)$ ein Ausschlußzustand. Das Wort W ist somit Element der rechten Seite.

„$\supseteq$“: Erreicht man von A mit W einen Ausschlußzustand, dann existiert ein Präfix V von W, so daß $next_\varepsilon(A,V)$ Ausschlußzustand ist. D.h., es existiert ein primitives Konzept P mit $P, \neg P \in next_\varepsilon(A,V)$. Nach Lemma 2.5 gilt also $V \in L(A,P) \cap L(A,\neg P)$, womit $W \in E_A$ gezeigt ist. □

Aus dieser Charakterisierung von E_A kann ein E_A-akzeptierender endlicher Automat gewonnen werden; E_A ist also eine reguläre Sprache. Den Automaten erhält man wie folgt: Wir betrachten zu $\mathcal{A}_T = (\Sigma, Q, E)$ den zugehörigen Potenzmengenautomaten (vgl. Seite 14), der durch den neuen Zustand q ergänzt wird. Die Zustandsmenge $\varepsilon\text{-}closure(\{A\})$ ist Anfangszustand und q ist Endzustand dieses Automaten. In diesen fügt man zu jedem Ausschlußzustand $F \subseteq Q$, $F = \varepsilon\text{-}closure(F)$, eine Transition (F, ε, q) ein und für alle $R \in \Sigma$ eine Transition (q, R, q). Man sieht leicht, daß der so konstruierte endliche Automat genau die Sprache E_A akzeptiert. Die Größe des Automaten ist i. allg. exponentiell in der Größe von T bzw. $\mathcal{A}_T$.

[6] Der Fall $W \in \Sigma^\omega$ wird erst im nächsten Abschnitt von Bedeutung sein.

Der folgende nicht-deterministische Algorithmus zur Entscheidung der Bedingung $L(B,P) \subseteq L(A,P) \cup E_A$ für ein primitives Konzept P arbeitet ähnlich wie Algorithmus 2.11. Er muß nur zusätzlich die reguläre Sprache E_A berücksichtigen. Dazu wird der obige Automat zu E_A simuliert. Wie Algorithmus 2.11 rät der nun anzugebende Algorithmus ein Wort W, welches die Inklusion widerlegt. Dabei wird ausgenutzt: $W \notin L(A,P) \cup E_A$ gdw. $P \notin next_\varepsilon(A,W)$ und von A aus wird mit W kein Ausschlußzustand erreicht. Diese Aussage ist mit Lemma 2.5 und Lemma 5.15 leicht einzusehen.

Algorithmus 5.16.

Eingabe: Semi-Automat $\mathcal{A}_T = (\Sigma, Q, E)$ ohne Worttransitionen zu einer Terminologie T; Konzepte A,B und primitives Konzept P aus T. ($n = |Q|$)

Ausgabe: Es existiert eine Berechnung mit Ausgabe „ja“ gdw. $L(B,P) \not\subseteq L(A,P) \cup E_A$.

$T_1 := \varepsilon\text{-}closure(\{B\})$;
$T_2 := \varepsilon\text{-}closure(\{A\})$;
$z := 0$;
while $z < 2^{2\cdot n} - 1$ and $T_2 \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ and ($P \notin T_1$ or $P \in T_2$) do
    $z := z + 1$;
    Wähle (nicht-deterministisch) ein $R \in \Sigma$;
    $T_1 := next_\varepsilon(T_1, R)$;
    $T_2 := next_\varepsilon(T_2, R)$
end;
If $T_2 \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ and $P \in T_1$ and $P \notin T_2$
    then Ausgabe „ja“
    else Ausgabe „nein“ $\triangle$

Korrektheit: Gibt der Algorithmus „ja“ aus, so existiert ein Wort $W \in \Sigma^*$, $W = R_1 \cdots R_m$, $m < 2^{2\cdot n}$, für das gilt: $P \in next_\varepsilon(B,W)$ wegen $T_1 = next_\varepsilon(B,W)$ und $P \in T_1$; entsprechend gilt $P \notin next_\varepsilon(A,W)$; nach Konstruktion gilt $next_\varepsilon(A,V) \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ für alle Präfixe V von W. Daraus folgt $W \in L(B,P)$ und $W \notin L(A,P)$ nach Lemma 2.5 sowie $W \notin E_A$ nach Lemma 5.15. Damit ist Algorithmus 5.16 korrekt.

Vollständigkeit: Gilt $L(B,P) \not\subseteq L(A,P) \cup E_A$, so existiert ein Wort $W = R_1 \cdots R_m \in \Sigma^*$ mit $W \in L(B,P)$ und $W \notin L(A,P) \cup E_A$. Es seien $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ und $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$ für alle $0 \le i \le m$. Ist $m \ge 2^{2\cdot n}$, so existieren Zahlen l und r mit $0 \le l < r \le 2^{2\cdot n}$ sowie $T_{1,l} = T_{1,r}$ und $T_{2,l} = T_{2,r}$. Damit werden für $W' = R_1 \cdots R_l R_{r+1} \cdots R_m$ ab $T_{1,0}$ bzw. $T_{2,0}$ die Zustandsmengen $T_{1,0}, \ldots, T_{1,l}, T_{1,r+1}, \ldots, T_{1,m}$ bzw. $T_{2,0}, \ldots, T_{2,l}, T_{2,r+1}, \ldots, T_{2,m}$ durchlaufen. Wegen $P \in T_{1,m}$ ist nach Lemma 2.5 das Wort W' in $L(B,P)$ enthalten, entsprechend gilt $W' \notin L(A,P)$ wegen $P \notin T_{2,m}$. Außerdem folgt aus Lemma 5.15 und $T_{2,i} \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ für alle $0 \le i \le m$, daß W' nicht von A ausgeschlossen wird. Für das im Vergleich zu W kürzere Wort W' gilt also $W' \in L(B,P) \setminus (L(A,P) \cup E_A)$. Für ein Wort $W \in L(B,P) \setminus (L(A,P) \cup E_A)$ minimaler Länge gilt somit $m < 2^{2\cdot n}$, da zu einem Wort, das größer oder gleich $2^{2\cdot n}$ ist, nach obiger Argumentation ein kürzeres Wort existiert mit entsprechenden Eigenschaften.

Wählt der Algorithmus im i-ten Schleifendurchlauf $R = R_i$, so gilt nach Lemma 2.5: $P \in T_1 = next_\varepsilon(B,W)$ und $P \notin T_2 = next_\varepsilon(A,W)$. Wegen $W \notin E_A$ gilt nach Lemma 5.15 für alle Präfixe V von W: $next_\varepsilon(A,V) \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$. Insgesamt liefert dies eine Berechnung mit Ausgabe „ja“. Damit ist Algorithmus 5.16 vollständig.

Komplexität: Die Terminierung des Algorithmus ist trivial. Die Bedingung $T_2 \notin \{F \subseteq Q;$ F Ausschlußzustand$\}$ kann nach Seite 31 mit linearem Aufwand entschieden werden. Wie Algorithmus 2.11 so ist auch Algorithmus 5.16 ein NPSPACE-Algorithmus.

Gibt man in Algorithmus 5.16 die primitive Negation $\neg P$ statt P ein, so entscheidet dieser das Problem $L(B, \neg P) \not\subseteq L(A, \neg P) \cup E_A$.

Nach Savitch's Theorem [HU69] gilt NPSPACE=PSPACE. Zusammen mit Algorithmus 5.16 kann somit die rechte Seite der Äquivalenz von Satz 5.4 mit einem PSPACE-Algorithmus entschieden werden. Nach [Baa96], Korollar 21 ist die Subsumtion bzgl. der gfp-Semantik in $\mathcal{FL}_0$ PSPACE-hart. Also ist sie insbesondere auch bzgl. $\mathcal{AL}_0$ PSPACE-hart. Insgesamt erhalten wir

Korollar 5.17.
Das Subsumtionsproblem bzgl. der gfp-Semantik in $\mathcal{AL}_0$ ist PSPACE-vollständig. □

Die Charakterisierung der Subsumtion bzgl. der gfp-Semantik in $\mathcal{FL}_0$ (vgl. (5.2)) ermöglicht es, die Subsumtion direkt auf das Inklusionsproblem für reguläre Sprachen zu reduzieren, so daß sofort das PSPACE-Resultat folgt. Auch $L(B, P) \not\subseteq L(A, P) \cup E_A$ ist ein Inklusionsproblem regulärer Sprachen; wir haben bisher jedoch nur einen exponentiell großen endlichen Automaten zu E_A angegeben, so daß das PSPACE-Resultat nicht direkt folgt. Aus der Definition von E_A läßt sich jedoch leicht folgende Charakterisierung ablesen:

Lemma 5.18 (alternative Charakterisierung von E_A).
Für eine Terminologie T, den zugehörigen Semi-Automaten $\mathcal{A}_T = (\Sigma, Q, E)$ sowie für ein Konzept A und die Menge S_P der primitiven Konzepte in T ist

$$E_A = \left[\bigcup_{P \in S_P} L(A, P) \cap L(A, \neg P) \right] \cdot \Sigma^*.$$

□

Es sei an dieser Stelle skizziert, wie aus dieser Charakterisierung in polynomialer Zeit ein endlicher Automat zu E_A konstruiert werden kann:

Ist A inkonsistent, so ist $E_A = \Sigma^*$ und man gibt leicht einen zugehörigen endlichen Automaten an. Ist A konsistent, so ist nach Satz 5.8 das leere Wort ε für kein primitives Konzept P in $L(A, P) \cap L(A, \neg P)$ enthalten. Man kann deshalb statt $L(A, P)$ und $L(A, \neg P)$ die Sprachen $L(A, P) \setminus \{\varepsilon\}$ und $L(A, \neg P) \setminus \{\varepsilon\}$ betrachten; nach Lemma 2.8 sind zu diesen beiden Sprachen in polynomialer Zeit Semi-Automaten mit Buchstabentransitionen konstruierbar. Der Produktautomat der beiden Automaten (vgl. Satz 2.9) akzeptiert nun die Sprache $L(A, P) \cap L(A, \neg P)$. Zu den Produktautomaten der primitiven Konzepte bildet man nun den Vereinigungsautomaten und modifiziert diesen so, daß die Konkatenation mit Σ^* berücksichtigt wird.

Somit können die endlichen Automaten zu den Sprachen $L(B, P)$ und $L(A, P) \cup E_A$ in polynomialer Zeit aus $\mathcal{A}_T$ konstruiert werden und mit Algorithmus 2.11 ist die Inklusion der Sprachen entscheidbar, was insgesamt ebenfalls die Existenz eines PSPACE-Algorithmus für die Entscheidung der Subsumtion bzgl. der gfp-Semantik in $\mathcal{AL}_0$ liefert.

Der Vorteil von Algorithmus 5.16 ist der, daß dieser — wie wir sehen werden — zum einen für die anderen Semantiken und zum anderen auch für die Sprache $\mathcal{ALN}$ verwendet werden kann. Dies liegt daran, daß E_A dort jeweils auch durch (dann anders definierte) Ausschlußzustände charakterisiert werden kann. Eine Charakterisierung von E_A durch einen polynomial großen endlichen Automaten ist für $\mathcal{ALN}$ nicht mehr so leicht möglich, so daß die polynomiale Reduktion auf das Inklusionsproblem für reguläre Sprachen schwer fällt.

5.3 Charakterisierung der deskriptiven Semantik in $\mathcal{AL}_0$

Wie im vorherigen Abschnitt kann auch bzgl. der deskriptiven Semantik in $\mathcal{AL}_0$ ein Konzept inkonsistent sein, so daß auch hier die Charakterisierung der Subsumtion nicht direkt von $\mathcal{FL}_0$ auf $\mathcal{AL}_0$ übertragbar ist. Die Charakterisierung der deskriptiven Semantik selbst läßt sich jedoch leicht von $\mathcal{FL}_0$ auf $\mathcal{AL}_0$ verallgemeinern.

Satz 5.19 (Charakterisierung der deskriptiven Semantik bzgl. $\mathcal{AL}_0$).
Es sei T eine $\mathcal{AL}_0$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat, J eine primitive Interpretation und $\underline{A}$ ein Tupel mit $T_J(\underline{A}) \subseteq \underline{A}$; weiter bezeichne I das durch J und $\underline{A}$-$gfp(T_J)$ definierte Modell (vgl. Satz 2.4). Für jedes Konzept A und alle Individuen $d \in dom(I)$ gilt: $d \in A^I$ gdw.

(P1) für alle primitiven Konzepte P in T, alle Wörter $W \in L(A, P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in P^I$; und

(P2) für alle Terme $\neg P$ in T, alle Wörter $W \in L(A, \neg P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\neg P)^I$; und

(P3) für alle definierten Konzepte B, alle Wörter $W \in L(A, B)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\underline{A})_j$ $(j = index(B))$.

Beweis: analog zum Beweis von Proposition 28 in [Baa96]. Die primitive Negation $\neg P$ ist im Beweis wie ein primitives Konzept zu behandeln. □

5.3.1 Die deskriptive Semantik in $\mathcal{AL}_0$ — Inkonsistenz

Die Ergebnisse zur Inkonsistenz aus Abschnitt 5.2.1 übertragen sich aufgrund des folgenden Zusammenhangs:

Lemma 5.20 (Inkonsistenz bzgl. der deskriptiven und der gfp-Semantik).
Es sei T eine ($\mathcal{ALN}$-)Terminologie und A ein Konzept aus T. Es gilt: A ist inkonsistent bzgl. der gfp-Semantik gdw. A bzgl. der deskriptiven Semantik inkonsistent ist.

Beweis: Es sei I ein Modell von T. Durch I werde die primitive Interpretation J und das Tupel $\underline{A}$, für das $\underline{A} = T_J(\underline{A})$ gilt, bestimmt. Es gilt $\underline{A} \subseteq gfp(T_J)$. Durch J und $gfp(T_J)$ ist das gfp-Modell bzgl. J und T gegeben.

Ist A bzgl. der gfp-Semantik konsistent, so auch bzgl. der deskriptiven Semantik, da jedes gfp-Modell von T insbesondere ein Modell von T ist. Ist A bzgl. der deskriptiven Semantik konsistent, so wegen $\underline{A} \subseteq gfp(T_J)$ auch bzgl. der gfp-Semantik. □

Die Charakterisierung der Inkonsistenz bzgl. der gfp-Semantik in Satz 5.8 kann also unverändert auf die deskriptive Semantik übertragen werden. Dazu übernehmen wir die Begriffe „kanonisches Modell“ (Definition 5.5) und „Ausschlußzustand“ (Definition 5.7). Insbesondere ist auch hier die Inkonsistenz mit polynomialem Zeitaufwand entscheidbar.

5.3.2 Die deskriptive Semantik in $\mathcal{AL}_0$ — Subsumtion

Analog zu Abschnitt 5.2.2, Seite 32 wird man auch für die Subsumtion bzgl. der deskriptiven Semantik in $\mathcal{AL}_0$ Wörter berücksichtigen müssen, die ein Konzept ausschließen. Wir übernehmen dazu für die deskriptive Semantik den Begriff „Ausschluß“ (Definition 5.9) sowie die Definition von E_A (Seite 33).

Lemma 5.10 überträgt sich auf die deskriptive Semantik wie folgt:

Lemma 5.21.
Ist I ein Modell zur Terminologie T, W ein A-ausschließendes Wort, und sind $d, f \in dom(I)$ Individuen mit $dW^I f$, dann gilt $d \notin A^I$.

Beweis: analog zum Beweis von Lemma 5.10 unter Verwendung von Satz 5.19 (statt Satz 5.4). □

In [Baa96], Theorem 29 wird die Subsumtion bzgl. der deskriptiven Semantik zweier Konzepte A und B einer $\mathcal{FL}_0$-Terminologie T wie folgt charakterisiert:

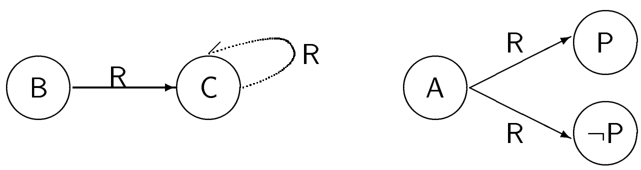
$A \sqsubseteq_T B$ gdw.

1.) $L(B, P) \subseteq L(A, P)$ für alle primitiven Konzepte P in T ist; und

2.) für alle definierten Konzepte C und alle unendlichen Pfade der Form $B, U_0, C, U_1, C, U_2, \ldots$ ein $k \geq 0$ existiert mit $U_0 \cdots U_k \in L(A, C)$. (5.5)

Aus denselben Gründen wie bei der gfp-Semantik wird man in 1.) die Bedingung $L(B, P) \subseteq L(A, P)$ in $L(B, P) \subseteq L(A, P) \cup E_A$ ändern und die primitive Negation entsprechend behandeln. Wie das folgende Beispiel zeigt, wird auch die Bedingung 2.) in dieser Form nicht beibehalten werden können.

Beispiel 5.22.
Die Terminologie T sei durch den zugehörigen Semi-Automaten $\mathcal{A}_T$ wie folgt gegeben:



In $\mathcal{A}_T$ existiert der unendliche Pfad $\mathsf{B}, \mathsf{R}, \mathsf{C}, \mathsf{R}, \mathsf{C}, \mathsf{R}, \mathsf{C}, \ldots$, und es gilt $\mathsf{R}^n \notin L(\mathsf{A}, \mathsf{C}) = \emptyset$ für alle $n \geq 1$. Wegen 2.) dürfte dann $\mathsf{A} \sqsubseteq_T \mathsf{B}$ nicht gelten. Trotzdem gilt $\mathsf{A} \sqsubseteq_T \mathsf{B}$, da in einem Modell I von T für $d \in \mathsf{A}^I$ das Individuum d keinen R-Nachfolger haben kann. Ein R-Nachfolger müßte nach Satz 5.19 nämlich Element von P^I und $(\neg\mathsf{P})^I$ sein. Mit Satz 5.19 folgt also $d \in \mathsf{B}^I$. ◇

Die Wörter R^n, $n \geq 1$, in Beispiel 5.22 schließen A aus, so daß kein Modell I existiert mit $d \in \mathsf{A}^I \setminus \mathsf{B}^I$ für ein Individuum d. Wird A nicht von Wörtern, die die Inklusion widerlegen, ausgeschlossen, so ist die Konstruktion eines derartigen Modells jedoch möglich.

Definition 5.23 (erweiterte kanonische primitive Interpretation).
Es sei T eine $\mathcal{AL}_0$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T und W ein Wort aus $\Sigma^* \cup \Sigma^\omega$. Zu einem Individuum d_0 und $W \in \Sigma^*$ wird die

erweiterte kanonische primitive Interpretation $J_{min} = J_{min}(A, d_0, W)$ wie in Definition 5.11 erklärt. Für $W = R_1R_2R_3 \cdots \in \Sigma^\omega$ definieren wir $J_{min} = J_{min}(A, d_0, W)$ durch: $dom(J_{min}) := \{d_0, d_1, d_2, \ldots\}$; $R^{J_{min}} := \{(d_{i-1}, d_i);\ i \geq 1 \text{ und } R = R_i\}$ für alle Rollen R in T; für alle primitiven Konzepte P in T und Individuen $d \in dom(J_{min})$ sei $d \in P^{J_{min}}$ gdw. ein $V \in \Sigma^*$ mit $V \in L(A, P)$ und $d_0 V^{J_{min}} d$ existiert. $\diamond$

Um für die erweiterte kanonische primitive Interpretation $J_{min} = J_{min}(A, d_0, W)$ auf die Eigenschaften (P1) und (P2) von Satz 5.19 bzgl. A, d_0 T und J_{min} schließen zu können, ist die folgende Bedingung hinreichend[7]:

Das Konzept A ist konsistent und wird nicht vom (endlichen oder unendlichen) Wort W ausgeschlossen (Definition 5.9). (5.6)

Das folgende Lemma faßt Eigenschaften von J_{min} zusammen.

Lemma 5.24.
Mit den Bezeichnungen von Definition 5.23 gelten die folgenden Aussagen:

1.) zum endlichen Wort W (vgl. Definition 5.23) existiert ein Individuum $d \in dom(J_{min})$ mit $d_0 W^{J_{min}} d$. Zum unendlichen Wort W (vgl. Definition 5.23) existieren $d_1, d_2, d_3, \ldots \in dom(J_{min})$ mit $d_0 R_1^{J_{min}} d_1 R_2^{J_{min}} d_2 R_3^{J_{min}} d_3 \cdots$;

2.) zu jedem Individuum $d \in dom(J_{min})$ existiert eindeutig ein Wort $V \in \Sigma^*$, das Präfix von W ist, und für das $d_0 V^{J_{min}} d$ gilt;

3.) ist $V \notin L(A, P)$ und $d_0 V^{J_{min}} d$ für ein endliches Wort V, ein primitives Konzept P und ein Individuum d, dann gilt $d \notin P^{J_{min}}$;

4.) ist Bedingung (5.6) erfüllt, so gelten (P1) und (P2) von Satz 5.19 bzgl. A, d_0, T und J_{min}.

Beweis: Die Punkte 1.) und 2.) folgen direkt aus der Definition von J_{min}. Aussage 3.) folgt leicht aus 2.) und der Definition von J_{min}.

Die Bedingungen (5.4) und (5.6) stimmen für $W \in \Sigma^*$ überein. Damit folgt für $W \in \Sigma^*$ Aussage 4.) sofort aus Lemma 5.12, 5.). Für das ω-Wort W zeigt man (P1) und (P2) von Satz 5.19 bzgl. A, d_0, T und J_{min} genauso wie für den Fall $W \in \Sigma^*$. $\square$

Damit können wir den folgenden Satz zeigen:

Satz 5.25 (Charakterisierung: Subsumtion bzgl. deskr. Semantik in $\mathcal{AL}_0$).
Es bezeichne T eine $\mathcal{AL}_0$-Terminologie und $\mathcal{A}_T$ den zugehörigen Semi-Automaten, A und B bezeichnen Konzepte in T. Es gilt $A \sqsubseteq_T B$ gdw.

1.) $L(B, P) \subseteq L(A, P) \cup E_A$ für alle primitiven Konzepte P in T; und

2.) $L(B, \neg P) \subseteq L(A, \neg P) \cup E_A$ für alle Terme der Form $\neg P$ in T; und

3.) für alle definierten Konzepte C und alle unendlichen Pfade der Form $B, U_0, C, U_1, C, U_2, C, \ldots$ existiert ein $k \geq 0$, so daß $U_0 \cdots U_k \in L(A, C) \cup E_A$.

[7] und notwendig (nach Satz 5.8 und Lemma 5.21)

Beweis: „$\Leftarrow$“: Es gelte die rechte Seite der Behauptung. Die Interpretation I sei das durch die primitive Interpretation J und den Fixpunkt $\underline{A}$ von T_J definierte Modell von T. Offensichtlich gilt $T_J(\underline{A}) \subseteq \underline{A}$ und $\underline{A} = \underline{A}\text{-}gfp(T_J)$. Es sei $d \in dom(I)$ mit $d \notin B^I$. Zu zeigen ist $d \notin A^I$.

Wegen $d \notin B^I$ gilt nach Satz 5.19, daß (P1), (P2) oder (P3) nicht gelten. Die Fälle, daß (P1) oder (P2) nicht gelten, zeigt man unter Verwendung von Satz 5.19 (statt Satz 5.4) und Lemma 5.21 (statt Lemma 5.10) analog zum Satz 5.13 („$\Leftarrow$“).

Gilt (P3) nicht, so existiert ein definiertes Konzept C_1, ein Wort $W_1 \in L(B, C_1)$ sowie ein Individuum $e_1 \in dom(I)$ mit $(d, e_1) \in W_1^I$ und $e_1 \notin (\underline{A})_{i_1}$ ($i_1 = index(C_1)$). Nach Voraussetzung ist $(\underline{A})_{i_1} = C_1^I$, und wir können mit C_1 anstelle von B fortfahren. Wir nehmen an, daß wir bereits eine Folge $C_1, W_1, e_1, \ldots, C_k, W_k, e_k$ zu $e_0 := d$ und $C_0 := B$ erhalten haben mit $e_i \notin C_i^I$, $e_{i-1}W_i^I e_i$ und $W_i \in L(C_{i-1}, C_i)$ für alle $1 \leq i \leq k$.

Wegen $e_k \notin C_k^I$ gelten (P1), (P2) oder (P3) von Satz 5.19 nicht. Im Fall (P1) existiert also ein primitives Konzept P, ein Wort $W \in L(C_k, P)$ sowie ein Individuum $e \in dom(I)$ mit $(e_k, e) \in W^I$ und $e \notin P^I$. Es gilt also $W_1 \cdots W_k W \in L(B, P) \subseteq L(A, P) \cup E_A$, $e \notin P^I$ und $d(W_1 \cdots W_k W)^I e$. Im Fall $W_1 \cdots W_k W \in L(A, P)$ ist (P1) von Satz 5.19 verletzt, damit gilt $d \notin A^I$. Im Fall $W_1 \cdots W_k W \in E_A$ gilt $d \notin A^I$ nach Lemma 5.21. Für den Fall (P2) folgert man $d \notin A^I$ analog.

Für den Fall, daß (P3) nicht gilt, können wir für alle $k \in \mathbb{N}$ annehmen, daß (P3) nicht gilt, da sonst die schon betrachteten Fälle (P1) und (P2) eintreten. Wir haben also einen unendlichen Pfad $B, W_1, C_1, W_2, C_2, W_3, C_3, \ldots$ und Individuen $e_1, e_2, e_3, \ldots$ mit den oben beschriebenen Eigenschaften. Es muß demnach ein Konzept C existieren mit $C = C_i$ für unendlich viele Indizes i, somit auch ein unendlicher Pfad der Form $B, U_0, C, U_1, C, U_2, \ldots$ Nach 3.) existiert demnach ein $k \geq 0$ mit $U_0 \cdots U_k \in L(A, C) \cup E_A$. Außerdem existiert ein Index i mit $d(U_0 \cdots U_k)^I e_i$ und $e_i \notin C^I = (\underline{A})_j$ ($j = index(C)$). Für den Fall $U_0 \cdots U_k \in L(A, C)$ liefert (P3) von Satz 5.19 für A und d: $d \notin A^I$; für $U_0 \cdots U_k \in E_A$ impliziert Lemma 5.21 ebenfalls $d \notin A^I$.

„$\Rightarrow$“: Es gelte $A \sqsubseteq_T B$; insbesondere gilt damit $A \sqsubseteq_{gfp,T} B$. Da E_A für die gfp-Semantik und die deskriptive Semantik gleich definiert sind, stimmen die Aussagen 1.) und 2.) von Satz 5.13 und Satz 5.25 überein. Wegen $A \sqsubseteq_{gfp,T} B$ ist somit nur noch 3.) zu zeigen. Wir nehmen dazu an, daß 3.) nicht gilt und konstruieren ein Modell, das im Widerspruch zu $A \sqsubseteq_T B$ steht. Es existiert also ein unendlicher Pfad der Form $B, U_0, C, U_1, C, U_2, \ldots$, so daß $U_0 \cdots U_k \notin L(A, C) \cup E_A$ für alle $k \geq 0$. Wegen $U_0 \cdots U_k \notin E_A$ ist A konsistent, da ansonsten $E_A = \Sigma^*$ gilt. Außerdem wird A durch das (endliche oder unendliche) Wort $W = U_0 U_1 U_2 \cdots$ (nach Definition) nicht ausgeschlossen. Insgesamt liefert dies die Existenz der erweiterten kanonischen primitiven Interpretation $J = J_{min}(A, d_0, W)$ zu A, dem Individuum d_0 und dem Wort W mit den in Lemma 5.24 aufgeführten Eigenschaften. Es seien $j_1 \leq j_2 \leq \cdots$ Indizes, so daß $d_0 U_0^J d_{j_1} U_1^J d_{j_2} U_2^J \cdots$ gilt. Das Tupel $\underline{A}$ sei wie folgt definiert: Für ein definiertes Konzept D in T mit Index m ist $(\underline{A})_m := dom(J) \setminus \{e;$ es existieren Wörter X, Y und ein $k \geq 0$ mit $XY = U_0 \cdots U_k$, $X \in L(B, D)$, $Y \in L(D, C)$, $d_0 X^J e$ und $eY^J d_{j_{k+1}}\}$.

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

Beweis der Behauptung: Sei D ein definiertes Konzept in T und $m = index(D)$. Wir nehmen $e \notin (\underline{A})_m$ an. Zu zeigen ist $e \notin (T_J(\underline{A}))_m$.

Nach Definition von $\underline{A}$ bedeutet $e \notin (\underline{A})_m$, daß endliche Wörter X, Y und ein Index $k \geq 0$ existieren mit $XY = U_0 \cdots U_k$, $X \in L(B, D)$, $Y \in L(D, C)$ sowie $d_0 X^J e$ und $eY^J d_{j_{k+1}}$. O.E. können wir annehmen, daß der Pfad von D nach C nicht leer ist (sonst betrachte $k+1$ statt k). Wir können deshalb $Y = Y_1Y_2$ wählen, so daß ein Individuum e' mit eY_1^Je' und $e'Y_2^Jd_{j_{k+1}}$ existiert, und so daß das definierende Axiom von D die Form $D = \cdots \sqcap \forall Y_1.D' \sqcap \cdots$ besitzt. Der Index von D' sei mit m' bezeichnet. Die Definition von $\underline{A}$ liefert $e' \notin (\underline{A})_{m'}$ und somit $e \notin (T_J(\underline{A}))_m$. Damit ist die Behauptung gezeigt.

Es sei nun I das durch J und $\underline{A}\text{-}gfp(T_J)$ (wohldefiniert nach Behauptung) definierte Modell zu T. Der Index zu B sei j, d.h. $B^I = (\underline{A}\text{-}gfp(T_J))_j$. Wegen $d_0\varepsilon^Id_0$, $d_0U_0^Jd_{j_1}$ und $U_0 \in L(B,C)$ ist $d_0 \notin (\underline{A})_j$, also $d_0 \notin (\underline{A}\text{-}gfp(T_J))_j = B^I$.
Annahme: $d_0 \notin A^I$. Nach Lemma 5.24, 4.), da A konsistent ist und nicht von W ausgeschlossen wird, ist (P1) und (P2) für A, d_0, T und J erfüllt. Nach Satz 5.19 ist also (P3) nicht erfüllt. Es existiert somit ein definiertes Konzept D, ein Wort $U \in L(A,D)$ und ein Individuum $e \in dom(I)$ mit d_0U^Ie und $e \notin (\underline{A})_l$ ($l = index(D)$). Nach Definition von $\underline{A}$ existieren demnach Wörter X, Y und ein Index $k \geq 0$ mit $XY = U_0 \cdots U_k$, $X \in L(B,D)$, $Y \in L(D,C)$ sowie d_0X^Je und $eY^Jd_{j_{k+1}}$. Wegen d_0U^Je und d_0X^Je sind nach Lemma 5.24, 2.) die Wörter U und X identisch. Also ist $UY = XY = U_0 \cdots U_k \in L(A,C)$, im Widerspruch dazu, daß 3.) nicht gilt. □

In Beispiel 5.22 ist $E_{\mathsf{A}} = \mathsf{R}\mathsf{R}^*$. Offensichtlich gilt damit $\emptyset = L(\mathsf{B},\mathsf{P}) \subseteq L(\mathsf{A},\mathsf{P}) \cup E_A$ und $\emptyset = L(\mathsf{B},\neg\mathsf{P}) \subseteq L(\mathsf{A},\mathsf{P}) \cup E_{\mathsf{A}}$. Die Bedingungen 1.) und 2.) von Satz 5.25 sind also erfüllt. Der einzige unendliche Pfad, der von B ausgeht, ist $\mathsf{B},\mathsf{R},\mathsf{C},\mathsf{R},\mathsf{C},\ldots$. Wegen $\mathsf{R} \in L(\mathsf{A},\mathsf{C}) \cup E_A$ ist damit auch Bedingung 3.) erfüllt, was insgesamt $\mathsf{A} \sqsubseteq_T \mathsf{B}$ liefert.

Die Charakterisierung der Subsumtion in Satz 5.25 kann herangezogen werden, um die Subsumtion bzgl. der deskriptiven Semantik in $\mathcal{AL}_0$ zu entscheiden. Dazu müssen die Bedingungen 1.), 2.) und 3.) des Satzes 5.25 getestet werden. Für 1.) und 2.) haben wir schon im Abschnitt 5.2.2 die Existenz eines PSPACE-Algorithmus nachgewiesen. Es bleibt, einen Entscheidungsalgorithmus für Bedingung 3.) zu formulieren. Um die Existenz eines PSPACE-Algorithmus nachzuweisen, der 3.) entscheidet, reicht offensichtlich die Angabe eines NPSPACE-Entscheidungsalgorithmus, der zu fest vorgegebenen Konzepten A, B und C folgendes Problem entscheidet:

Es existiert ein unendlicher Pfad der Form $B, U_0, C, U_1, C, U_2, C, \ldots$, so daß für alle $k \geq 0$ gilt: $U_0 \cdots U_k \notin L(A,C) \cup E_A$. (5.7)

Wir geben dazu einen (nicht-deterministischen) Algorithmus an, der einen solchen unendlichen Pfad rät, falls er existiert. Ist n die Größe der Zustandsmenge von $\mathcal{A}_T$, so rät der Algorithmus genauer zwei Wörter $U_0 \in \Sigma^*$, $|U_0| < 2^{2\cdot n}$, und $U_1 \in \Sigma^+$, $|U_1| \leq 2^{2\cdot n}$, so daß $B, U_0, C, U_1, C, U_1, C, U_1, C \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$ ist mit $U_0U_1^k \notin L(A,C) \cup E_A$ für alle $k \geq 0$. Liegt C auf einem ε-Zyklus, so reicht es, ein Wort $U_0 \in \Sigma^*$ mit $U_0 \in L(B,C)$ und $U_0 \notin L(A,C) \cup E_A$ zu finden, um Bedingung (5.7) zu erfüllen.

Algorithmus 5.26.

Eingabe: Semi-Automat $\mathcal{A}_T = (\Sigma, Q, E)$ ohne Worttransitionen zu einer Terminologie T; Konzepte A, B und C aus T

Ausgabe: Es existiert eine Berechnung mit Ausgabe „ja" gdw. Bedingung (5.7) erfüllt ist.

Es bezeichne n die Größe der Zustandsmenge, M die Menge der Konzepte in T, die auf einem ε-Zyklus liegen sowie L die Menge der Ausschlußzustände. Ist $S \subseteq Q$, so sei $\overline{S}$ das Komplement von S, also $\overline{S} := Q \setminus S$.

T_1:=ε-*closure*($\{B\}$);
T_2:=ε-*closure*($\{A\}$);
z:=0;
stop:=false;
(* Raten von U_0 *)
(1) If $C \in T_1 \cap \overline{T_2}$ then begin
    Setze (nicht-det.) *stop* auf *false* oder *true* (* $U_0 = \varepsilon$ *)
end;
while $z < 2^{2 \cdot n} - 1$ and (not *stop*) and $T_2 \notin L$ do begin
    z:=z+1;
    Wähle (nicht-det.) ein $R \in \Sigma$;
    $T_1 := next_\varepsilon(T_1, R)$;
    $T_2 := next_\varepsilon(T_2, R)$;
(2)   If $C \in T_1 \cap \overline{T_2}$ then begin
        Setze *stop* (nicht-det.) auf *true* oder *false*
    end
end;
(3) If $T_2 \notin L$ and $C \in T_1 \cap \overline{T_2}$ and $C \in M$ then Ausgabe „ja";
(4) If $T_2 \notin L$ and $C \in T_1 \cap \overline{T_2}$ then begin
    (* Raten von U_1 *)
    $z := 0$;
    $T_1 := \varepsilon$-*closure*($\{C\}$);
    $T_2' := T_2$;
    $stop := false$;
    while $z < 2^{2 \cdot n}$ and (not *stop*) and $T_2 \notin L$ do begin
        $z := z + 1$;
        Wähle (nicht-det.) ein $R \in \Sigma$;
        $T_1 := next_\varepsilon(T_1, R)$;
        $T_2 := next_\varepsilon(T_2, R)$;
(5)   If $C \in T_1$ and $T_2 = T_2'$ (* insb. $C \notin T_2$ *) then begin
            Setze (nicht-det.) *stop* auf *false* oder *true*
        end
    end; (*while*)
(6)   If $T_2 \notin L$ and $C \in T_1$ and $T_2 = T_2'$ then Ausgabe „ja"
end; (*if*)
Ausgabe „nein" △

Korrektheit: Gibt der Algorithmus „ja" aus, so existiert nach Definition des Algorithmus ein $U_0 \in \Sigma^*$, $U_0 = R_1 \cdots R_{m_0}$, $m_0 < 2^{2 \cdot n}$, so daß für $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ und $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$ gilt: $T_{2,i} \notin L$ für alle $0 \le i \le m_0$ und $C \in T_{1,m_0} \cap \overline{T_{2,m_0}}$. Außerdem unterscheiden wir folgende Fälle:

i) $C \in M$ (If-Bedingung von (3) ist erfüllt.)

Wegen $C \in T_{1,m_0}$ ist nach Lemma 2.5 das Wort U_0 ein Element der Sprache $L(B, C)$. Damit und wegen $C \in M$ ist für $U_i = \varepsilon$, $i \ge 1$, der Pfad $B, U_0, C, U_1, C, U_2, \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$. Weiter ist $U_0 \notin L(A, C)$ wegen $C \notin T_{2,m_0}$ (Lemma 2.5). Nach Lemma 5.15 ist $U_0 \notin E_A$ aufgrund von $T_{2,i} \notin L$ für alle $0 \le i \le m_0$. Somit existiert zum unendlichen Pfad $B, U_0, C, U_1, C, U_2, \ldots$ kein $k \ge 0$ mit $U_0 \cdots U_k \in L(A, C) \cup E_A$.

ii) (If-Bedingung von (6) ist erfüllt)
Es existiert ein nicht leeres Wort $U_1 \in \Sigma^+$, $U_1 = R'_1 \cdots R'_{m_1}$, $1 \leq m_1 \leq 2^{2 \cdot n}$, so daß mit $T'_{1,i} := next_\varepsilon(C, R'_1 \cdots R'_i)$ und $T'_{2,i} := next_\varepsilon(T_{2,m_0}, R'_1 \cdots R'_i)$ für alle $0 \leq i \leq m_1$ gilt: $C \in T'_{1,m_1}$ und $T'_{2,m_1} = T_{2,m_0}$ (insb. $C \notin T'_{2,m_1}$) und $T'_{2,i} \notin L$ für alle $0 \leq i \leq m_1$. Damit existiert in $\mathcal{A}_T$ der unendliche Pfad $B, U_0, C, U_1, C, U_2, \ldots$ mit $U_i = U_1$ für alle $i \geq 1$, da $U_0 \in L(B, C)$ (wegen $C \in T_{1,m_0}$) und $U_1 \in L(C, C)$ (wegen $C \in T'_{1,m_1}$ und $T'_{1,0} = \varepsilon\text{-}closure(\{C\})$). Wegen $T_{2,i} \notin L$ für alle $0 \leq i \leq m_0$ erreicht man mit U_0 von A aus keinen Ausschlußzustand, und wegen $T'_{2,i} \notin L$ für alle $0 \leq i \leq m_1$ erreicht man von $T'_{2,0} = T_{2,m_0}$ mit U_1 keinen Ausschlußzustand. Somit gilt $U_0 U_1 \notin E_A$. Da T'_{2,m_1} und T_{2,m_0} übereinstimmen, erreicht man auch mit $U_0 U_1 \cdots U_k$ von A aus für beliebige $k \geq 0$ keinen Ausschlußzustand, also $U_0 U_1 \cdots U_k \notin E_A$ für alle $k \geq 0$. Aus $C \notin T_{2,m_0}$ folgt $U_0 \notin L(A, C)$. Wegen $C \notin T'_{2,m_1}$ und $T'_{2,0} = T_{2,m_0}$ folgt $U_0 U_1 \notin L(A, C)$. Zusammen mit $T_{2,m_0} = T'_{2,m_1}$ liefert dies $U_0 \cdots U_k \notin L(A, C)$ für alle $k \geq 0$. Zum unendlichen Pfad $B, U_0, C, U_1, C, U_2, \ldots$ existiert also kein $k \geq 0$ mit $U_0 \cdots U_k \in L(A, C) \cup E_A$. Damit ist die Korrektheit des Algorithmus nachgewiesen.

Vollständigkeit: Es sei $B, U_0, C, U_1, C, U_2, \ldots$ ein unendlicher Pfad, so daß kein $k \geq 0$ mit $U_0 \cdots U_k \in L(A, C) \cup E_A$ existiert. Wir zeigen die Existenz einer Berechnung mit Ausgabe „ja“ und unterscheiden dazu die Fälle iii) und iv).

iii) Für den Fall, daß $U_0 U_1 U_2 \cdots$ ein endliches Wort ist, folgt $C \in M$. Es sei $W = U_0 U_1 U_2 \cdots$ das endliche Wort $R_1 \cdots R_m$. Wir definieren $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ und $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$ für alle $0 \leq i \leq m$. Auch $B, W, C, \varepsilon, C, \varepsilon, \ldots$ ist ein unendlicher Pfad in $\mathcal{A}_T$ mit $W \notin L(A, C)$ und $W \notin E_A$ (denn: wähle k so, daß $W = U_0 \cdots U_k$).

O.E. ist $m < 2^{2 \cdot n}$, denn sonst existieren l und r mit $0 \leq l < r \leq 2^{2 \cdot n}$ sowie $T_{1,l} = T_{1,r}$ und $T_{2,l} = T_{2,r}$. Für $W' = R_1 \cdots R_l R_{r+1} \cdots R_m$ werden ab $T_{1,0}$ bzw. $T_{2,0}$ die Zustandsmengen $T_{1,0}, T_{1,1}, \ldots, T_{1,l}, T_{1,r+1}, \ldots, T_{1,m}$ bzw. $T_{2,0}, T_{2,1}, \ldots, T_{2,l}, T_{2,r+1}, \ldots, T_{2,m}$ durchlaufen. Damit gilt wie für W auch für das kürzere Wort W': $W' \in L(B, C)$ und $W' \notin L(A, C) \cup E_A$.

Es sei also $W = R_1 \cdots R_m$ ein Wort mit $m < 2^{2 \cdot n}$, so daß $B, W, C, \varepsilon, C, \varepsilon, \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$ ist und außerdem $W \notin L(A, C) \cup E_A$ gilt. Damit zeigen wir nun die Existenz einer Berechnung mit Ausgabe „ja“. Wir unterscheiden die Fälle $W = \varepsilon$ und $W \neq \varepsilon$.

Für $W = \varepsilon$ kann in (1) von Algorithmus 5.26 die Variable *stop* gleich *true* gewählt werden, so daß die erste while-Schleife nicht durchlaufen wird. Wegen $\varepsilon \notin L(A, C)$ und $\varepsilon \in L(B, C)$ ist $C \in T_1 \cap \overline{T_2}$. Das Konzept C ist nach Voraussetzung in M enthalten. Wegen $\varepsilon \notin E_A$ ist $T_2 = \varepsilon\text{-}closure(\{A\}) \notin L$. Die Bedingung in (3) ist also erfüllt, und es wird „ja“ ausgegeben.

Für $W \neq \varepsilon$ setzt man in (1), falls die If-Bedingung erfüllt ist, $stop = false$. Wegen $W \notin E_A$ gilt vor und auch nach jedem Schleifendurchlauf $T_2 \notin L$. Insbesondere wird nach (1) die while-Schleife durchlaufen. Im i-ten Schleifendurchlauf wählt man für R den Buchstaben R_i. Ist die Bedingung von (2) erfüllt, so wird, falls $i < m$, *stop* auf *false* gesetzt. Wegen $m < 2^{2 \cdot n}$ und $W \notin E_A$ ist die while-Bedingung auch am Ende des $(m-1)$-ten Schleifendurchlaufes noch erfüllt. Im m-ten Schleifendurchlauf wird R auf R_m gesetzt. Wegen $W \in L(B, C)$ und $W \notin L(A, C)$ ist dann die Bedingung $C \in T_1 \cap \overline{T_2}$ in (2) erfüllt. Nun wird *stop* auf *true* gesetzt. Die while-Schleife wird somit verlassen. Da die Bedingungen in (3) ebenfalls erfüllt sind ($W \notin E_A$), wird „ja“ ausgegeben.

iv) Nun sei $W = U_0 U_1 U_2 \cdots$ das ω-Wort $W = R_1 R_2 R_3 \cdots \in \Sigma^\omega$. O.E. gilt $U_i \neq \varepsilon$ für alle $i \geq 1$. Es seien $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ und $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$ für alle $i \geq 0$. Für die Indizes $i_0, i_1, i_2, \ldots$ gelte $R_1 \cdots R_{i_k} = U_0 \cdots U_k$ für alle $k \geq 0$. Wegen $|2^Q| = 2^n$

existieren zur Folge $T_{2,i_0}, T_{2,i_1}, \ldots$ die Zahlen l und r mit $0 \leq l < r$ sowie $T_{2,i_l} = T_{2,i_r}$. Für die Wörter $U_0 \cdots U_l$ und $U_{l+1} \cdots U_r$ gelten also die Aussagen: $U_0 \cdots U_l \in L(B,C)$, $U_0 \cdots U_l \notin L(A,C) \cup E_A$, $U_{l+1} \cdots U_r \in L(C,C)$, $T_{2,i_l} = T_{2,i_r} = next_\varepsilon(T_{2,i_l}, U_{l+1} \cdots U_r)$ und $U_0 \cdots U_l \cdots U_r \notin E_A$.

Wir werden nun aus $U_0 \cdots U_l$ und $U_{l+1} \cdots U_r$ Wörter U'_0 bzw. U'_1 bestimmen, so daß $B, U'_0, C, U'_1, C, U'_1, \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$ ist mit $U'_0 {U'_1}^k \notin L(A,C) \cup E_A$ für alle $k \geq 0$ sowie $|U'_0| < 2^{2\cdot n}$ und $0 < |U'_1| \leq 2^{2\cdot n}$.

Ist $i_l \geq 2^{2\cdot n}$, so existieren s und t mit $0 \leq s < t \leq 2^{2\cdot n}$, $T_{1,s} = T_{1,t}$ und $T_{2,s} = T_{2,t}$. Wir setzen $U'_0 := R_1 \cdots R_s R_{t+1} \cdots R_{i_l}$. Für U'_0 werden also ab $T_{1,0}$ bzw. $T_{2,0}$ die Zustandsmengen $T_{1,0}, T_{1,1}, \ldots, T_{1,s}, T_{1,t+1}, \ldots, T_{1,i_l}$ bzw. $T_{2,0}, T_{2,1}, \ldots, T_{2,s}, T_{2,t+1}, \ldots, T_{2,i_l} \notin L$ durchlaufen. Damit gelten, wie für $U_0 \cdots U_l$, für U'_0 die Aussagen $U'_0 \notin L(A,C) \cup E_A$ und $U'_0 \in L(B,C)$. Für ein U'_0 minimaler Länge gilt somit $|U'_0| < 2^{2\cdot n}$, da zu einem Wort aus $L(B,C) \setminus (L(A,C) \cup E_A)$, das größer oder gleich $2^{2\cdot n}$ ist, nach obiger Argumentation ein kürzeres Wort existiert mit entsprechenden Eigenschaften.

Wegen $U_i \neq \varepsilon$ für alle $i \geq 1$ und $l < r$ ist $U_{l+1} \cdots U_r \neq \varepsilon$. Ist $|U_{l+1} \cdots U_r| > 2^{2\cdot n}$, so wird für $T'_{1,j} := next_\varepsilon(C, R_{i_l+1} \cdots R_j)$ $(i_l \leq j \leq i_r)$ zum Wort $U_{l+1} \cdots U_r$ die Folge $(T'_{1,i_l}, T_{2,i_l}), (T'_{1,i_l+1}, T_{2,i_l+1}), \ldots, (T'_{1,i_r}, T_{2,i_r})$ von Paaren aus $2^Q \times 2^Q$ durchlaufen. Wegen $|2^Q \times 2^Q| = 2^{2\cdot n}$ existieren also s und t mit $i_l \leq s < t \leq i_l + 2^{2\cdot n}$ sowie $T'_{1,s} = T'_{1,t}$ und $T_{2,s} = T_{2,t}$. Für $U'_1 := R_{i_l+1} \cdots R_s R_{t+1} \cdots R_{i_r}$ werden somit ab T'_{1,i_l} bzw. T_{2,i_l} die Zustandsmengen $T'_{1,i_l}, \ldots, T'_{1,s}, T'_{1,t+1}, \ldots, T'_{1,i_r}$ bzw. $T_{2,i_l}, \ldots, T_{2,s}, T_{1,t+1}, \ldots, T_{2,i_r} \notin L$ durchlaufen. Damit gelten, wie zu $U_{l+1} \cdots U_r$, für U'_1 die Aussagen: $U'_1 \in L(C,C)$, $T_{2,i_l} = T_{2,i_r} = next_\varepsilon(T_{2,i_l}, U'_1)$, $U'_0 U'_1 \notin E_A$ sowie $|U'_1| > 0$. Nach obiger Argumentation gilt für ein nicht leeres Wort U'_1 minimaler Länge, für das die obigen Aussagen erfüllt sind: $0 < |U'_1| \leq 2^{2\cdot n}$.

Für $U'_i := U'_1$ für alle $i \geq 2$ ist nun insgesamt wegen $U'_0 \in L(B,C)$, $U'_1 \in L(C,C)$ und $|U'_1| > 0$ der Pfad $B, U'_0, C, U'_1, C, U'_2, C, \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$. Wegen $U'_0 \notin L(A,C) \cup E_A$, $U'_0 U'_1 \notin L(A,C) \cup E_A$ und $C \notin T_{2,i_l} = T_{2,i_r} = next_\varepsilon(T_{2,i_l}, U'_1)$ ist $U'_0 \cdots U'_k \notin L(A,C) \cup E_A$ für alle $k \geq 0$.

Damit kann wie folgt eine Berechnung des Algorithmus 5.26 mit Ausgabe „ja" angeben werden: Ist $U'_0 = \varepsilon$, so ist $\varepsilon \in L(B,C)$ und $\varepsilon \notin L(A,C)$, also $C \in T_1 \cap \overline{T_2}$. In (1) wählt man für *stop* die Belegung *true*. Somit wird die erste while-Schleife übersprungen. Ist dagegen $U'_0 \neq \varepsilon$, so wird, falls die If-Bedingung in (1) erfüllt ist, *stop* mit *false* belegt. Zusammen mit $U'_0 \notin E_A$ ist die Bedingung der while-Schleife erfüllt. In dieser Schleife ist dann R gemäß U'_0 zu wählen. Außer im letzten Durchlauf der Schleife setzt man, falls die If-Bedingung von (2) erfüllt ist, *stop* auf *false*. Wegen $|U'_0| < 2^{2\cdot n}$, $stop = false$ (außer nach Wahl des letzten Zeichens von U'_0) sowie $U'_0 \notin E_A$, d.h. $T_2 \notin L$, wird die Schleife $|U'_0|$-mal durchlaufen. Beim letzten Durchlauf ist wegen $U'_0 \in L(B,C)$ und $U'_0 \notin L(A,C)$ die Bedingung in (2) erfüllt, so daß *stop* auf *true* gesetzt werden kann.

Ist nach Verlassen der while-Schleife die Bedingung in (3) erfüllt, so sind wir fertig. Ansonsten ist nach obigen Überlegungen die Bedingung in (4) erfüllt. In der zweiten while-Schleife wird R nun gemäß U'_1 gewählt. Falls (5) erfüllt ist, setzt man, außer beim letzten Durchlauf, d.h. letztem Zeichen von U'_1, *stop* auf *false*. Wegen $|U'_1| \leq 2^{2\cdot n}$, $stop = false$ und $U'_0 U'_1 \notin E_A$ ist die Bedingung der while-Schleife bis zum letzten Durchlauf erfüllt. Außerdem ist aufgrund von $U'_0 U'_1 \in L(B,C)$ und $T_{2,i_l} = T_{2,i_r}$ die Bedingung in (5) im letzten Durchlauf erfüllt, so daß *stop* auf *true* gesetzt werden kann, was zum Verlassen der while-Schleife führt. Schließlich ist wegen $U'_0 U'_1 \notin E_A$, und da die Bedingung in (5) erfüllt war, die Bedingung in (6) erfüllt, so daß „ja" ausgegeben wird.

Damit konnten wir in allen Fällen die Existenz einer Berechnung mit Ausgabe „ja" nachweisen.

Komplexität: Man sieht leicht, daß Algorithmus 5.26 immer terminiert und mit polynomialen Platz auskommt.

Algorithmus 5.26 zeigt, daß 3.) von Satz 5.25 mit einem PSPACE-Algorithmus entschieden werden kann. Da auch die Probleme 1.) und 2.) von Satz 5.25 in PSPACE liegen, folgt

Korollar 5.27.
Das Subsumtionsproblem bzgl. der deskriptiven Semantik in $\mathcal{AL}_0$ ist in PSPACE enthalten. □

In [Neb91], Proposition 10 wurde die Äquivalenz von Konzepten in $\mathcal{FL}_0$ bzgl. der deskriptiven Semantik charakterisiert. Dies lieferte ebenfalls einen PSPACE-Entscheidungsalgorithmus für Subsumtion und Äquivalenz. Es konnte bisher aber weder mit Hilfe dieser noch mit der Charakterisierung der Subsumtion (vgl. (5.5)) der Nachweis für die PSPACE-Härte erbracht werden.

Die Bedingung (5.5), 2.) wird in [Baa96] mit Hilfe von Büchi-Automaten entschieden. Dies ist auch für das Problem (5.7) in ähnlicher Weise möglich. Wir skizzieren dazu eine polynomiale Reduktion der Negation von (5.7), im folgenden $\neg$(5.7) genannt, auf das Inklusionsproblem ω-regulärer Sprachen. Dazu sei $\Sigma_\# := \Sigma \dot\cup \{\#\}$, $L\# := \{w\#; w \in L\}$ für ein $L \subseteq \Sigma^*$. Die Abbildung $\psi : \Sigma_\#^* \longrightarrow \Sigma^*$ sei der durch $\psi(\sigma) = \sigma$ für alle $\sigma \in \Sigma$ und $\psi(\#) = \varepsilon$ eindeutig definierte Homomorphismus. Für einen Semi-Automaten $\mathcal{A}$, der die Zustände p und q enthält, sei $L_{p,q}(\mathcal{A}) = L_{p,q} := \{w; w \in \Sigma^*$ ist Label eines nicht-leeren Pfades von p nach $q\}$. Nach [HU79], Theorem 3.5 sind die Sprachen $\psi^{-1}(L_{p,q}) := \{w \in \Sigma_\#^*; \psi(w) \in L_{p,q}\}$ und $\psi^{-1}(L_{p,q})\#$ regulär. Die zugehörigen Automaten können jeweils in polynomialer Zeit aus den Ausgangsautomaten konstruiert werden.

Für die Sprache $L_{\mathcal{A}_T}(A,C) \cup E_A$ haben wir auf Seite 37 die Konstruktion eines in der Größe von $\mathcal{A}_T$ polynomialen endlichen Automaten skizziert. Man konstruiert — in polynomialer Zeit — auch zu den Sprachen $L_{B,C}(\mathcal{A}_T)$ und $L_{C,C}(\mathcal{A}_T)$ aus $\mathcal{A}_T$ leicht endliche Automaten. Diese Automaten werden zum Automaten $\mathcal{B}_{T,A,B,C}$ vereinigt, so daß für passende Zustände p_1, q_1, p_2, q_2, p_3 und q_3 gilt: $L_{\mathcal{A}_T}(A,C) \cup E_A = L_{p_1,q_1}(\mathcal{B}_{T,A,B,C})$, $L_{B,C}(\mathcal{A}_T) = L_{p_2,q_2}(\mathcal{B}_{T,A,B,C})$ und $L_{C,C}(\mathcal{A}_T) = L_{p_3,q_3}(\mathcal{B}_{T,A,B,C})$.

Wir zeigen damit die folgende Aussage:

Lemma 5.28.
Die Bedingung $\neg$(5.7) gilt für Konzepte A, B, C aus der Terminologie T gdw.

$$(L_{p_2,q_2}\#)(L_{p_3,q_3}\#)^\omega \subseteq \psi^{-1}(L_{p_1,q_1})\#(L_{p_3,q_3}\#)^\omega.$$

Dabei beziehen sich die Sprachen und Zustände auf den oben definierten Automaten $\mathcal{B}_{T,A,B,C}$.

Beweis: „$\Rightarrow$“: Es gelte $\neg$(5.7). Das Wort W sei ein Element in $(L_{p_2,q_2}\#)(L_{p_3,q_3}\#)^\omega$, d.h. W ist von der Form $U_0\#U_1\#U_2\#\cdots$, wobei U_0 Label eines nicht-leeren Pfades von p_2 nach q_2 ist und U_i, $i \geq 1$, Label eines nicht-leeren Pfades von p_3 nach q_3. Nach $\neg$(5.7) existiert ein $k \geq 0$ mit $U_0 \cdots U_k \in L_{\mathcal{A}_T}(A,C) \cup E_A = L_{p_1,q_1}$. Dann ist $U_0\# \cdots \#U_k \in \psi^{-1}(L_{p_1,q_1})$ und somit $W = U_0\#U_1\# \cdots U_k\#U_{k+1}\# \cdots \in (\psi^{-1}(L_{p_1,q_1})\#)(L_{p_3,q_3}\#)^\omega$.

„$\Leftarrow$“: Es gelte die rechte Seite. Es sei $B, U_0, C, U_1, C, \ldots$ ein unendlicher Pfad von B aus, der C unendlich oft erreicht. Dann ist $U_0\#U_1\# \cdots \in L_{p_2,q_2}\#(L_{p_3,q_3}\#)^\omega$, denn o.E.

ist der Pfad von p_2 nach q_2 mit Label U_0 nicht leer. Ansonsten existiert ein k, so daß der Pfad von p_2 nach q_2 mit Label $U_0 \cdots U_k$ nicht leer ist. Wähle dann $U_0 \cdots U_k$ statt U_0. Da das letzte Symbol der Wörter aus $L_{p_1,q_1}\#$ jeweils das Zeichen '#' ist, existiert ein $k \geq 0$ mit $U_0\# \cdots U_k\# \in \psi^{-1}(L_{p_1,q_1})\#$. Also $U_0\# \cdots \#U_k \in \psi^{-1}(L_{p_1,q_1})$ und $U_0 \cdots U_k \in L_{p_1,q_1}$. □

Damit ist $\neg(5.7)$ mit polynomialem Aufwand auf das Inklusionsproblem für ω-reguläre Sprachen reduziert. Wie in Kapitel 2, Seite 18 erwähnt, ist das Inklusionsproblem für ω-reguläre Sprachen PSPACE-vollständig. Dies zeigt — wie auch Algorithmus 5.26 —, daß Bedingung (5.7) bzw. $\neg(5.7)$ mit einem PSPACE-Algorithmus entscheidbar sind. Wie Algorithmus 5.16 so hat auch Algorithmus 5.26 den Vorteil, daß sich dieser leicht auf die Sprache $\mathcal{ALN}$ verallgemeinern läßt; es ist lediglich die Definition des Ausschlußzustandes anzupassen. Eine polynomiale Charakterisierung von E_A durch einen endlichen Automaten fällt für $\mathcal{ALN}$ dagegen schwer, so daß eine polynomiale Reduktion auf das Inklusionsproblem ω-regulärer Sprachen nicht so leicht möglich ist.

5.4 Charakterisierung der lfp-Semantik in $\mathcal{AL}_0$

Es fällt, wie bei der gfp- und der deskriptiven Semantik, auch hier leicht, die Charakterisierung der lfp-Semantik in $\mathcal{FL}_0$ auf $\mathcal{AL}_0$ zu übertragen. Anders als für die gfp- und die deskriptive Semantik spielen — wie bei $\mathcal{FL}_0$ (vgl. [Baa96], Proposition 22) — unendliche Ketten für die Charakterisierung der Inkonsistenz, der Subsumtion sowie der lfp-Semantik selbst eine wichtige Rolle.

Satz 5.29 (Charakterisierung der lfp-Semantik bzgl. $\mathcal{AL}_0$).
Es sei T eine $\mathcal{AL}_0$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat, I ein lfp-Modell von T sowie A ein Konzept in T. Dann gilt für alle Individuen $d \in dom(I)$: $d \in A^I$ gdw.

(P1) für alle primitiven Konzepte P in T, alle Wörter $W \in L(A, P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in P^I$; und

(P2) für alle Terme $\neg P$ in T, alle Wörter $W \in L(A, \neg P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\neg P)^I$; und

(P3) für alle unendlichen Pfade der Form $A, W_1, C_1, W_2, C_2, \ldots$ und alle Individuen $d_1, d_2, d_3, \ldots \in dom(I)$ existiert ein $n \geq 1$ mit $(d_{n-1}, d_n) \notin W_n^I$. („Unendliche Ketten sind verboten.")

Beweis: analog zum Beweis von Proposition 22 in [Baa96]. Die primitive Negation $\neg P$ ist im Beweis wie ein primitives Konzept zu behandeln. □

5.4.1 Die lfp-Semantik in $\mathcal{AL}_0$ — Inkonsistenz

Zusätzlich zu den in Satz 5.8 genannten Gründen für Inkonsistenz liefern bzgl. der lfp-Semantik unendliche Ketten weitere. Diese verursachen auch schon in $\mathcal{FL}_0$ inkonsistente Konzepte (vgl. [Baa96], Korollar 23). Geht nämlich von einem Konzept ein unendlicher Pfad mit Label ε aus, so ist nach (P3) klar, daß dieses Konzept inkonsistent sein muß.

Für die Charakterisierung der Inkonsistenz sind im Beweis bzw. in der algorithmischen Behandlung wiederum das kanonische Modell bzw. der Begriff des Ausschlußzustandes nützlich.

Wir definieren die *kanonische primitive Interpretation* $J = J(A, d_0)$ zu einem Konzept A und einem Individium d_0 wie in Defintion 5.5. Das *kanonische Modell* $I = I(A, d_0)$ ist nun das zu J und einer Terminologie T gehörende lfp-Modell. Um für dieses Modell die Aussage $d_0 \in A^I$ beweisen zu können, ist, wie oben angedeutet, Bedingung (5.1) nicht ausreichend.

Lemma 5.30.
Es sei T eine Terminologie und $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T. Weiterhin existiere weder ein primitives Konzept P in T mit $\varepsilon \in L(A, P) \cap L(A, \neg P)$ noch ein Konzept C, das auf einem ε-Zyklus liegt, mit $\varepsilon \in L(A, C)$. Für das kanonische Modell $I = I(A, d_0)$ zu A und Individuum d_0 ist dann $d_0 \in A^I$.

Beweis: Wir zeigen, daß (P1), (P2) und (P3) von Satz 5.29 bzgl. A, d_0, T und $J = J(A, d_0)$ gelten. Dies haben wir für (P1) und (P2) in Lemma 5.6 bereits getan, da mit den Voraussetzungen dieses Lemmas insbesondere Bedingung (5.1) erfüllt ist. Zudem beziehen sich (P1) und (P2) nur auf die primitive Interpretation nicht auf ein (gfp-)Modell und können auch unabhängig davon nachgewiesen werden.

Gilt (P3) nicht, so existieren ein unendlicher Pfad $A, W_1, C_1, W_2, C_2, W_3, C_3, \ldots$ in $\mathcal{A}_T$ und Individuen $d_1, d_2, d_3, \ldots$ mit $(d_{n-1}, d_n) \in W_n^I$ für alle $n \geq 1$. Nach Definition von J gilt dann $d_n = d_0$ und $W_n = \varepsilon$ für alle $n \geq 1$. Dies liefert einen unendlichen Pfad von A aus mit Label ε. Es existiert demnach ein Konzept C mit $C = C_i$ für unendlich viele Indizes i; insbesondere liegt C auf einem ε-Zyklus. Zusammen mit $\varepsilon \in L(A, C)$ ist dies ein Widerspruch zur Voraussetzung. □

Die Bedeutung unendlicher Ketten macht auch eine andere Definition des Auschlußzustandes als die für die gfp- und die deskriptive Semantik nötig.

Definition 5.31 (Ausschlußzustand bzgl. der lfp-Semantik in $\mathcal{AL}_0$).
Es bezeichne T eine Terminologie und $\mathcal{A}_T = (\Sigma, Q, E)$ den zugehörigen Semi-Automaten. Dann heißt $F \subseteq Q$ *Ausschlußzustand* bzgl. $\mathcal{A}_T$ (und bzgl. der lfp-Semantik in $\mathcal{AL}_0$), falls

1.) ein primitives Konzept P in T existiert mit $P, \neg P \in F$; oder

2.) ein definiertes Konzept C in T existiert, das auf einem ε-Zyklus liegt, und für das $C \in F$ gilt.

◇

Offensichtlich läßt sich mit in der Größe von T polynomialem Aufwand entscheiden, ob eine gegebene Menge $F \subseteq Q$ Auschlußzustand ist.

Man zeigt nun

Satz 5.32 (Charakterisierung: Inkonsistenz bzgl. lfp-Semantik in $\mathcal{AL}_0$).
Es sei T eine $\mathcal{AL}_0$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat und A ein Konzept in T. Folgende Aussagen sind äquivalent:

1.) A ist T-inkonsistent bzgl. der lfp-Semantik in $\mathcal{AL}_0$.

2.) (a) Es existiert ein primitives Konzept P in T mit $\varepsilon \in L(A, P) \cap L(A, \neg P)$; oder

(b) es existiert ein (definiertes) Konzept C in T, das auf einem ε-Zyklus liegt, und für das $\varepsilon \in L(A, C)$ gilt.

3.) ε-$closure(\{A\})$ ist ein Ausschlußzustand.

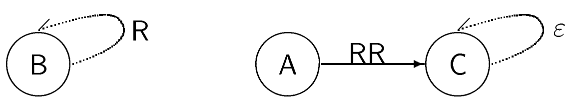
Beweis: Die Äquivalenz von 2.) und 3.) sollte klar sein. Es bleibt die Äquivalenz von 1.) und 2.) zu zeigen:
„1. $\Leftarrow$ 2.“: Es existiere ein primitives Konzept P in T mit $\varepsilon \in L(A, P) \cap L(A, \neg P)$. Ist nun I ein lfp-Modell von T mit $d \in A^I$ für ein Individuum d, so folgt nach (P1) und (P2) von Satz 5.29 wegen $d\varepsilon^I d$ der Widerspruch $d \in P^I$ und $d \in (\neg P)^I$. Existiert ein definiertes Konzept C, das auf einem ε-Zyklus liegt, mit $\varepsilon \in L(A, C)$, so gilt für ein beliebiges lfp-Modell I und Individuum $d \in dom(I)$ aufgrund des unendlichen Pfades $A, \varepsilon, C, \varepsilon, C, \varepsilon, C, \ldots$ und wegen $d\varepsilon^I d$ nach (P3) die Aussage $d \notin A^I$. Dies zeigt die Inkonsistenz von A.
„1. $\Rightarrow$ 2.“: Wir nehmen an, daß sowohl (a) als auch (b) nicht gelten. Damit existiert zu A und einem Individuum d_0 nach Lemma 5.30 das kanonische Modell $I = I(A, d_0)$ mit $d_0 \in A^I$. Somit ist A konsistent. □

Da die Berechnung von ε-$closure(\{A\})$ und der Test auf Ausschlußzustand mit polynomialem Zeitaufwand möglich sind, ist die Inkonsistenz bzgl. der lfp-Semantik in $\mathcal{AL}_0$ mit polynomialem Zeitaufwand entscheidbar.

5.4.2 Die lfp-Semantik in $\mathcal{AL}_0$ — Subsumtion

Satz 5.29 impliziert, daß für die Subsumtion $A \sqsubseteq_{lfp,T} B$ zweier Konzepte A und B folgende Bedingung hinreichend ist: $L(B, P) \subseteq L(A, P)$ für alle primitiven Konzepte P in T, $L(B, \neg P) \subseteq L(A, \neg P)$ für alle Terme der Form $\neg P$ in T und $U(B) \subseteq U(A)$. Um auch eine notwendige Bedingung für die Subsumtion zu erhalten, ist aufgrund widersprüchlicher Bedingungen durch Paare P, $\neg P$ bzw. aufgrund unendlicher Ketten die Berücksichtigung von Ausschlußwörtern zu A nötig. Für die Bedingung $U(B) \subseteq U(A)$ macht dies das folgende Beispiel deutlich:

Beispiel 5.33.
Die Terminologie T sei durch den zugehörigen Automaten $\mathcal{A}_T$ wie folgt gegeben:



Obwohl $\mathsf{RRR}\cdots \in U(\mathsf{B})$ und $\mathsf{RRR}\cdots \notin U(\mathsf{A})$, also $U(\mathsf{B}) \not\subseteq U(\mathsf{A})$, gilt $\mathsf{A} \sqsubseteq_{lfp,T} \mathsf{B}$. Ist nämlich I ein lfp-Modell von T und d ein Individuum mit $d \in \mathsf{A}^I$, so kann zu d kein RR-Nachfolger existieren. Wäre das Individuum e ein RR-Nachfolger von d, so ist mit dem unendlichen Pfad $\mathsf{A}, \mathsf{RR}, \mathsf{C}, \varepsilon, \mathsf{C}, \varepsilon, \mathsf{C}, \varepsilon, \mathsf{C}, \ldots$ sowie mit $d(\mathsf{RR})^I e$ und $e\varepsilon^I e$ nach (P3) das Individuum d nicht in A^I enthalten. Für B und d sind (P1) und (P2) trivial erfüllt. Der einzige unendliche Pfad, der von B ausgeht, ist $\mathsf{B}, \mathsf{R}, \mathsf{B}, \mathsf{R}, \mathsf{B}, \mathsf{R}, \ldots$ Da d also keinen RR-Nachfolger besitzt, ist auch (P3) bzgl. B und d erfüllt, was insgesamt $d \in \mathsf{B}^I$ liefert. ◇

Daß in Beispiel 5.33 die Subsumtionsbeziehung $\mathsf{A} \sqsubseteq_{lfp,T} \mathsf{B}$ gilt, ist darauf zurückzuführen, daß die Wörter aus RRR^* das Konzept A in folgendem Sinn ausschließen:

Definition 5.34 (Ausschluß bzgl. der lfp-Semantik in $\mathcal{AL}_0$).
Es sei T eine $\mathcal{AL}_0$-Terminologie, $\mathcal{A}_T = (\Sigma, Q, E)$ der zugehörige Semi-Automat sowie A ein Konzept in T. Das endliche Wort $W \in \Sigma^*$ *schließt A aus*, falls ein Präfix $V \in \Sigma^*$ von W existiert, für das gilt:

1.) Es existiert ein primitives Konzept P in T mit $V \in L(A, P) \cap L(A, \neg P)$; oder

2.) es existiert ein (definiertes) Konzept C, so daß C auf einem ε-Zyklus liegt und $V \in L(A, C)$ gilt.

Ein ω-Wort $W \in \Sigma^\omega$ *schließt A aus*, falls A von einem endlichen Präfix von W ausgeschlossen wird oder $W \in U(A)$ gilt. $\diamond$

Die Bedeutung von Definition 5.34 halten wir im folgendem Lemma fest:

Lemma 5.35.
Mit den Bezeichnungen aus Definition 5.34 gilt für ein lfp-Modell I zur Terminologie T:

1.) Ist $W \in \Sigma^* \cup \Sigma^\omega$ sowie $V \in \Sigma^*$ Präfix von W mit den Eigenschaften 1.) oder 2.) von Definition 5.34, und sind $d, e \in dom(I)$ Individuen mit $dV^I e$, so gilt $d \notin A^I$.[8]

2.) Für ein ω-Wort $W \in U(A)$, $W = R_1 R_2 R_3 \cdots$, und für Individuen $d_0, d_1, d_2, \ldots \in dom(I)$ mit $d_0 R_1^I d_1 R_2^I d_2 \cdots$, ist $d_0 \notin A^I$.

Beweis: Die zweite Aussage folgt sofort mit (P3) von Satz 5.29.

Für $d \in A^I$ folgt aus Satz 5.29 wegen $dV^I e$ und $V \in L(A, P) \cap L(A, \neg P)$ der Widerspruch: $e \in P^I$ und $e \in (\neg P)^I$. Ist $V \in L(A, C)$ für ein Konzept C, das auf einem ε-Zyklus liegt, so ist mit dem unendlichen Pfad $A, V, C, \varepsilon, C, \varepsilon, C, \ldots$ sowie $dV^I e$ und $e\varepsilon^I e$ die Bedingung (P3) für A und d nicht erfüllt, also $d \notin A^I$. $\square$

Mit E_A bezeichnen wir wiederum die Menge der endlichen A-ausschließenden Wörter (vgl. Definition 5.34) zu einer Terminologie T. Zusätzlich werden wir auch die Menge $E_{A,\omega} := \{W \in \Sigma^* \cup \Sigma^\omega;\ A \text{ wird von } W \text{ ausgeschlossen}\}$ zur Charakterisierung der Subsumtion benötigen. Für den Entscheidungsalgorithmus zur Subsumtion betrachten wir die Menge $E^f_{A,\omega} := \{W \in \Sigma^* \cup \Sigma^\omega;\ \text{zu } W \text{ existiert ein endliches Präfix, welches } A \text{ ausschließt}\}$.

Für ein inkonsistentes Konzept A liefert Satz 5.32: $\varepsilon \in E_A$, $\varepsilon \in E_{A,\omega}$ und $\varepsilon \in E^f_{A,\omega}$. Aus Definition 5.34 folgt damit $E_A = \Sigma^*$ und $E_{A,\omega} = E^f_{A,\omega} = \Sigma^* \cup \Sigma^\omega$.

Für den Beweis der Charakterisierung der Subsumtion bzgl. der lfp-Semantik in $\mathcal{AL}_0$ benötigen wir wiederum den Begriff des erweiterten kanonischen Modells.

Definition 5.36 (erweitertes kanonisches lfp-Modell bzgl. $\mathcal{AL}_0$).
Es sei T eine Terminologie in $\mathcal{AL}_0$, $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T, W ein Wort in $\Sigma^* \cup \Sigma^\omega$ und d_0 ein Individuum.

Für $W \in \Sigma^*$ definieren wir die *erweiterten kanonischen primitiven Interpretationen* $J_{min} = J_{min}(A, d_0, W)$ und $J_{max} = J_{max}(A, d_0, W)$ wie in Definition 5.11. Für $W \in \Sigma^\omega$ wurde $J_{min} = J_{min}(A, d_0, W)$ bereits in Definition 5.23 definiert. Analog dazu kann $J_{max} = J_{max}(A, d_0, W)$ definiert werden:
$dom(J_{max}) := dom(J_{min})$; $R^{J_{max}} := R^{J_{min}}$ für alle Rollen R in T; für alle primitiven Konzepte P in T und alle Individuen $d \in dom(J_{max})$ sei weiter:

[8] Wird statt $dV^I e$ die Beziehung $dW^I f$ für ein Individuum f vorausgesetzt, so folgt, da V Präfix von W ist, die Existenz eines Individuums e mit $dV^I e$. Nach Aussage 1.) gilt somit ebenfalls $d \notin A^I$.

$d \in (\neg P)^{J_{max}}$ gdw. ein $V \in \Sigma^*$ mit $V \in L(A, \neg P)$ und $d_0 V^{J_{max}} d$ existiert.

Die zu $J_{min}(A, d_0, W)$ bzw. $J_{max}(A, d_0, W)$ gehörenden *erweiterten kanonischen lfp-Modelle* von T seien $I_{min}(A, d_0, W)$ bzw. $I_{max}(A, d_0, W)$. $\diamond$

Um nun für die erweiterten kanonischen Modelle $I_{min} = I_{min}(A, d_0, W)$ und $I_{max} = I_{max}(A, d_0, W)$ die Aussage $d_0 \in A^{I_{min}}$ bzw. $d_0 \in A^{I_{max}}$ zeigen zu können, ist die folgende Bedingung hinreichend[9]:

Das Konzept A ist konsistent und wird von W nicht ausgeschlossen (Definition 5.34). (5.8)

Eigenschaften der erweiterten kanonischen primitiven Interpretationen und der erweiterten kanonischen lfp-Modelle werden im folgenden Lemma zusammengefaßt.

Lemma 5.37 (Eigenschaften von J_{min} und I_{min} bzw. J_{max} und I_{max}).
Mit den Bezeichnungen von Definition 5.36 gelten die folgenden Aussagen:

1.) zu $W \in \Sigma^*$ existiert ein Individuum d mit $d_0 W^{J_{min}} d$ und $d_0 W^{J_{max}} d$; zu $W = R_1 R_2 R_3 \cdots \in \Sigma^\omega$ existieren $d_1, d_2, d_3, \ldots \in dom(J_{min}) = dom(J_{max})$ mit $d_0 R_1^{J_{min}} d_1 R_2^{J_{min}} d_2 \cdots$ und $d_0 R_1^{J_{max}} d_1 R_2^{J_{max}} d_2 \cdots$;

2.) zu jedem Individuum $d \in dom(J_{min}) = dom(J_{max})$ existiert ein eindeutiges Wort $V \in \Sigma^*$, welches Präfix von W ist, und für das $d_0 V^{J_{min}} d$ bzw. $d_0 V^{J_{max}} d$ gilt;

3.) ist $V \notin L(A, P)$ und $d_0 V^{J_{min}} d$ für ein Wort V, ein primitives Konzept P und ein Individuum d, dann gilt $d \notin P^{J_{min}}$;

4.) ist $V \notin L(A, \neg P)$ und $d_0 V^{J_{max}} d$ für ein Wort V, ein primitives Konzept P und ein Individuum d, dann gilt $d \notin (\neg P)^{J_{max}}$;

5.) ist Bedingung (5.8) erfüllt, so gelten für A, d_0 und J_{min} (bzw. J_{max}) die Bedingungen (P1), (P2) und (P3) von Satz 5.29; insbesondere gilt $d_0 \in A^{I_{min}}$ (bzw. $d_0 \in A^{I_{max}}$).

Beweis: Die Aussagen 1.) und 2.) folgen direkt aus den Definitionen von J_{min} bzw. J_{max}. Die Eigenschaften 3.) und 4.) folgen leicht aus Aussage 2.) und den Definitionen von J_{min} und J_{max}.

Zu 5.): Es sei Bedingung (5.8) erfüllt. Die Gültigkeit von (P1) und (P2) behandelt man analog zum Beweis von Lemma 5.12 — auch für den Fall $W \in \Sigma^\omega$.

Wir zeigen nun die Gültigkeit von (P3) bzgl. J_{min}. Für J_{max} zeigt man dies ebenso. Gilt (P3) bzgl. A, d_0, T und J_{min} nicht, so existiert ein unendlicher Pfad der Form $A, W_1, C_1, W_2, C_2, \ldots$, und so existieren Individuen $d_1, d_2, \ldots \in dom(J_{min})$ mit $(d_{i-1}, d_i) \in W_i^{J_{min}}$ für alle $i \geq 1$.

Ist $V = W_1 W_2 W_3 \cdots$ ein endliches Wort, so existiert ein $d \in dom(J_{min})$ mit $d_0 V^{J_{min}} d$; nach 2.) ist damit V Präfix von W. Außerdem existiert ein C, so daß $C = C_i$ für unendlich viele Indizes gilt; insbesondere muß damit C auf einem ε-Zyklus liegen. In $\mathcal{A}_T$ existiert also der unendliche Pfad $A, V, C, \varepsilon, C, \varepsilon, C, \ldots$ Somit wird A von V, also auch von W, ausgeschlossen, im Widerspruch zur Voraussetzung.

[9] und, wie man mit Satz 5.32 und Lemma 5.35 leicht zeigt, auch notwendig

Ist dagegen V ein ω-Wort, so müssen nach Definition von J_{min} die Wörter V und W übereinstimmen. Wegen $V \in U(A)$ wird damit im Widerspruch zur Voraussetzung ebenfalls A von W ausgeschlossen. Dies liefert die Gültigkeit von (P3) bzgl. A, d_0, T und J_{min}. □

Damit zeigen wir

Satz 5.38 (Charakterisierung: Subsumtion bzgl. lfp-Semantik in $\mathcal{AL}_0$). Es bezeichne T eine $\mathcal{AL}_0$-Terminologie und $\mathcal{A}_T$ den zugehörigen Semi-Automaten; A und B bezeichnen Konzepte in T. Es gilt $A \sqsubseteq_{lfp,T} B$ gdw.

1.) $L(B,P) \subseteq L(A,P) \cup E_A$ für alle primitiven Konzepte P in T; und

2.) $L(B,\neg P) \subseteq L(A,\neg P) \cup E_A$ für alle Terme der Form $\neg P$ in T; und

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

Beweis: „$\Rightarrow$“: Wir nehmen an, daß jeweils eine der Bedingungen 1.), 2.) oder 3.) nicht gilt und zeigen dann $A \not\sqsubseteq_{lfp,T} B$.

Mit Hilfe der erweiterten kanonischen Modelle $I_{min}(A,d_0,W)$ und $I_{max}(A,d_0,W)$ aus Definition 5.36 sowie mit Lemma 5.37 und Satz 5.29 zeigt man analog zum Beweis von Satz 5.13 („$\Rightarrow$“, (1), (2)), daß, falls 1.) oder 2.) nicht gilt, A nicht von B subsumiert wird.

Annahme: $U(B) \not\subseteq U(A) \cup E_{A,\omega}$.
Es existiert demnach ein $W \in U(B) \setminus (U(A) \cup E_{A,\omega})$. Wegen $W \notin E_{A,\omega}$ muß A konsistent sein, da sonst $E_{A,\omega} = \Sigma^* \cup \Sigma^\omega$ wäre. Außerdem wird A von W nicht ausgeschlossen, was nach Lemma 5.37, 5.) die Existenz des lfp-Modells $I_{min} = I_{min}(A,d_0,W)$ mit $d_0 \in A^{I_{min}}$ sichert[10] .

Ist W endlich, so exisitiert nach Lemma 5.37, 1.) ein $d \in dom(I_{min})$ mit $d_0 W^I d$. Wegen $W \in U(B)$ existiert ein (definiertes) Konzept C, das auf einem ε-Zyklus liegt, so daß $A, W, C, \varepsilon, C, \varepsilon, C, \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$ ist. Zusammen mit $d_0 W^{I_{min}} d$ und $d\varepsilon^{I_{min}} d$ liefert (P3) aus Satz 5.38 bzgl. B und d_0: $d_0 \notin B^{I_{min}}$.

Ist W das ω-Wort $R_1 R_2 R_3 \cdots$, so existieren nach Lemma 5.37, 1.) Individuen $d_1, d_2, d_3, \ldots \in dom(I_{min})$ mit $d_0 R_1^{I_{min}} d_1 R_2^{I_{min}} d_2 \cdots$, womit ebenfalls zusammen mit $W \in U(B)$ und (P3) bzgl. B und d_0 die Aussage $d_0 \notin B^I$ folgt. Insgesamt gilt also $A \not\sqsubseteq_{lfp,T} B$.

„$\Leftarrow$“: Es gelte die rechte Seite der Äquivalenzaussage. Wir nehmen $A \not\sqsubseteq_{lfp,T} B$ an. Es existiert also ein lfp-Modell I zu T und ein Individuum $d_0 \in dom(I)$ mit $d_0 \in A^I \setminus B^I$. Wegen $d_0 \notin B^I$ gelten also (P1), (P2) oder (P3) von Satz 5.29 bzgl. B, d_0, T und I nicht.

Die beiden Fälle (P1) und (P2) führt man mit Hilfe von Satz 5.29 (statt Satz 5.4) und Lemma 5.35 (statt Lemma 5.10) analog zum Beweis von Satz 5.13 („$\Leftarrow$“) zum Widerspruch.

Gilt (P3) nicht, so existiert ein unendlicher Pfad der Form $B, W_1, C_1, W_2, C_2, \ldots$ in $\mathcal{A}_T$, und so existieren Individuen $d_1, d_2, d_3, \ldots$ mit $(d_{n-1}, d_n) \in W_n^I$ für alle $n \geq 1$ (*). Wegen $W = W_1 W_2 W_3 \cdots \in U(B)$ und $U(B) \subseteq U(A) \cup E_{A,\omega}$ ist $W \in U(A)$ oder $W \in E_{A,\omega}$. Die Aussage $W \in U(A)$ zusammen mit (*) und (P3) bzgl. A und d_0 liefert $d_0 \notin A^I$, im Widerspruch zur Annahme. Weiter impliziert $W \in E_{A,\omega}$ zusammen mit (*) und Lemma 5.35 ebenfalls $d_0 \notin A^I$, im Widerspruch zur Annahme. Insgesamt zeigt dies, daß A von B subsumiert wird. □

[10]Statt I_{min} könnte hier auch I_{max} verwendet werden. Für ein ω-Wort W werden $J_{max} = J_{max}(A,d_0,W)$ bzw. $I_{max} = I_{max}(A,d_0,W)$ ansonsten nicht genötigt.

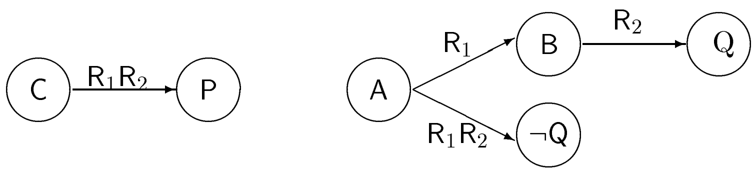
In Beispiel 5.33 kommen keine primitiven Konzepte und keine primitive Negation vor, weshalb die Bedingungen 1.) und 2.) von Satz 5.38 trivial erfüllt sind. Es ist $E_{A,\omega} = \mathsf{R}^\omega \cup \mathsf{RRR}^*$ und $\mathsf{B}, \mathsf{R}, \mathsf{C}, \mathsf{R}, \mathsf{C}, \ldots$ der einzige unendliche Pfad, der von B ausgeht. Damit ist $U(\mathsf{B}) \subseteq U(\mathsf{A}) \cup E_{A,\omega}$ erfüllt, und Satz 5.38 impliziert $\mathsf{A} \sqsubseteq_{lfp,T} \mathsf{B}$.

Wie schon erwähnt, können auch für $\mathcal{FL}_0$ bzgl. der lfp-Semantik Konzepte inkonsistent sein (unendliche Ketten). Aus diesem Grund ist entsprechend für $\mathcal{FL}_0$ die Bedingung: $L_{\mathcal{A}_T}(B,P) \subseteq L_{\mathcal{A}_T}(A,P)$ für alle primitiven Konzepte P und $U_{\mathcal{A}_T}(B) \subseteq U_{\mathcal{A}_T}(A)$ keine notwendige Bedingung für $A \sqsubseteq_{lfp,T} B$. In [Baa96] wird deshalb $\mathcal{A}_T$ zu einem Semi-Automaten $\mathcal{B}_T$ erweitert. Dazu wird $\mathcal{A}_T$ um einen Zustand Q_{loop}, eine Transition mit Label ε von Q_{loop} nach Q_{loop} sowie für jede Rolle R aus T um eine Transition von Q_{loop} nach Q_{loop} mit Label R ergänzt; für jeden Zustand B aus $\mathcal{A}_T$, der auf einem ε-Zyklus liegt, fügt man eine Transition mit Label ε von B nach Q_{loop} ein. Die Hinzunahme der letztgenannten Transitionen ist gleichbedeutend damit, daß man jedes inkonsistente Konzept B durch eine ε-Transition mit Q_{loop} verbindet. Nach [Baa96], Korollar 23 sind nämlich inkonsistente Konzepte gerade diejenigen, von denen aus ein endlicher Pfad mit Label ε zu einem Konzept existiert, das auf einem ε-Zyklus liegt. Schließlich wird zu jedem primitiven Konzept P eine Transition mit Label ε von Q_{loop} nach P eingefügt.

Statt $L_{\mathcal{A}_T}(B,P) \subseteq L_{\mathcal{A}_T}(A,P)$ und $U_{\mathcal{A}_T}(B) \subseteq U_{\mathcal{A}_T}(A)$ zu betrachten, geht man nun zu $L_{\mathcal{B}_T}(B,P) \subseteq L_{\mathcal{B}_T}(A,P)$ und $U_{\mathcal{B}_T}(B) \subseteq U_{\mathcal{B}_T}(A)$ über. Dies entspricht der Charakterisierung in Satz 5.38 (ohne 2.)). In $\mathcal{FL}_0$ würde man nämlich den Begriff „Ausschluß“ wie in Defintion 5.34 bestimmen, jedoch ohne 1.). Damit verifiziert man leicht: (*) $L_{\mathcal{B}_T}(A,P) = L_{\mathcal{A}_T}(A,P) \cup E_A$, $L_{\mathcal{B}_T}(B,P) = L_{\mathcal{A}_T}(B,P) \cup E_A$, $U_{\mathcal{B}_T}(A) = U_{\mathcal{B}_T}(A) \cup E_{A,\omega}$ und $U_{\mathcal{B}_T}(B) = U_{\mathcal{B}_T}(B) \cup E_{A,\omega}$.

Für $\mathcal{AL}_0$ kann $\mathcal{B}_T$ analog zur obigen Konstruktion definiert werden, wobei nun jedes bzgl. der lfp-Semantik in $\mathcal{AL}_0$ inkonsistente Konzept mit Q_{loop} über eine ε-Transition verbunden wird. Das folgende Beispiel zeigt, daß damit (*) i. allg. jedoch nicht gilt. Somit kann — anders als in $\mathcal{FL}_0$ — die Subsumtion nicht allein mit $\mathcal{B}_T$ charakterisiert werden.

Beispiel 5.39.
Die Terminologie T sei durch den Semi-Automaten $\mathcal{A}_T$ wie folgt gegeben:



Mit Satz 5.32 sieht man leicht, daß T kein inkonsistentes Konzept enthält. Somit gilt für den wie oben definierten Automaten $\mathcal{B}_T$ Folgendes: $L_{\mathcal{A}_T}(\mathsf{A},\mathsf{P}) = L_{\mathcal{B}_T}(\mathsf{B},\mathsf{P}) = \emptyset$, $\{\mathsf{R}_1\mathsf{R}_2\} = L_{\mathcal{A}_T}(\mathsf{C},\mathsf{P}) = L_{\mathcal{B}_T}(\mathsf{C},\mathsf{P})$, $L_{\mathcal{A}_T}(\mathsf{C},\mathsf{Q}) = L_{\mathcal{A}_T}(\mathsf{C},\neg\mathsf{Q}) = L_{\mathcal{B}_T}(\mathsf{C},\mathsf{Q}) = L_{\mathcal{B}_T}(\mathsf{C},\neg\mathsf{Q}) = \emptyset$, $E_A = \mathsf{R}_1\mathsf{R}_2\Sigma^*$ und $E_{A,\omega} = \mathsf{R}_1\mathsf{R}_2 \cdot (\Sigma^* \cup \Sigma^\omega)$. Demnach ist $L_{\mathcal{A}_T}(\mathsf{C},\mathsf{P}) \subseteq L_{\mathcal{A}_T}(\mathsf{A},\mathsf{P}) \cup E_A$, $L_{\mathcal{A}_T}(\mathsf{C},\mathsf{Q}) \subseteq L_{\mathcal{A}_T}(\mathsf{A},\mathsf{Q}) \cup E_A$, $L_{\mathcal{A}_T}(\mathsf{C},\neg\mathsf{Q}) \subseteq L_{\mathcal{A}_T}(\mathsf{A},\neg\mathsf{Q}) \cup E_A$ sowie $\emptyset = U_{\mathcal{A}_T}(\mathsf{C}) \subseteq U_{\mathcal{A}_T}(\mathsf{A}) \cup E_{A,\omega}$. Satz 5.38 liefert also $\mathsf{A} \sqsubseteq_{lfp,T} \mathsf{C}$.

Andererseits gilt $L_{\mathcal{B}_T}(\mathsf{C},\mathsf{P}) \subseteq L_{\mathcal{B}_T}(\mathsf{A},\mathsf{P})$ nicht. Dies liegt an der echten Inklusion $L_{\mathcal{B}_T}(\mathsf{A},\mathsf{P}) \subset L_{\mathcal{A}_T}(\mathsf{A},\mathsf{P}) \cup E_A$. Durch $\mathcal{B}_T$ werden die A-ausschließenden Wörter also nicht vollständig erfaßt. ◇

Der Automat $\mathcal{B}_T$ beschreibt lediglich Ausschluß aufgrund von Inkonsistenz, das Beispiel zeigt jedoch, daß auch Ausschluß ohne Inkonsistenz auftreten kann.

Um Satz 5.38 zur Entscheidung der Subsumtion heranzuziehen, werden wir zunächst E_A und $E^f_{A,\omega}$ mit Hilfe von Ausschlußzuständen charakterisieren.

Lemma 5.40.
Für die Terminologie T, den Semi-Automaten $\mathcal{A}_T$ ohne Worttransitionen und für ein Konzept A in T gilt:

1.) $E_A = \{W \in \Sigma^*;$ mit W erreicht man von A aus einen Ausschlußzustand$\}$,

2.) $E^f_{A,\omega} = \{W \in \Sigma^* \cup \Sigma^\omega;$ mit W erreicht man von A aus einen Ausschlußzustand$\}$.

Beweis: zu 1.) „$\subseteq$": Sei W ein Wort in E_A. Dann existiert ein Präfix $V \in \Sigma^*$ von W und ein primitives Konzept P oder ein definiertes Konzept C, das auf einem ε-Zyklus liegt, mit $V \in L(A,P) \cap L(A,\neg P)$ bzw. $V \in L(A,C)$. Also gilt $P, \neg P \in next_\varepsilon(A,V)$ bzw. $C \in next_\varepsilon(A,V)$. Damit ist $next_\varepsilon(A,V)$ ein Ausschlußzustand.
„$\supseteq$": Sei $W \in \Sigma^*$, so daß mit W von A aus einen Auschlußzustand erreicht wird, d.h. es existiert ein Präfix $V \in \Sigma^*$ von W und ein primitives Konzept P oder ein definiertes Konzept C, das auf einem ε-Zyklus liegt, mit $P, \neg P \in next_\varepsilon(A,V)$ bzw. $C \in next_\varepsilon(A,V)$. Damit impliziert Lemma 2.5: $V \in L(A,P) \cap L(A,\neg P)$ bzw. $V \in L(A,C)$. Also schließt W das Konzept A aus.

Aussage 2.) zeigt man analog zu 1.). □

Diese Charakterisierung von E_A erlaubt es, die Bedingung $L(B,P) \subseteq L(A,P) \cup E_A$ für ein primitives Konzept P mit Algorithmus 5.16 zu entscheiden. Dabei bezieht sich der Begriff „Ausschlußzustand" im Algorithmus nun auf Definition 5.31. Wird P durch $\neg P$ ersetzt, so erlaubt Algorithmus 5.16 auch die Entscheidung von $L(A,\neg P) \subseteq L(A,\neg P) \cup E_A$. Die Ausschlußzustandsbedingung $T_2 \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ kann auch bzgl. der lfp-Semantik in $\mathcal{AL}_0$ mit polynomialem Aufwand entschieden werden. Algorithmus 5.16 ist also auch hier ein NPSPACE-Algorithmus. Die Entscheidungsprobleme 1.) und 2.) in Satz 5.38 liegen somit in PSPACE.

Es bleibt, einen (NPSPACE-)Entscheidungsalgorithmus für die Bedingung 3.), also $U(B) \subseteq U(A) \cup E_{A,\omega}$, von Satz 5.38 anzugeben. Wegen $E_{A,\omega} = E^f_{A,\omega} \cup U(A)$ ist 3.) äquivalent zu $U(B) \subseteq U(A) \cup E^f_{A,\omega}$.

Der Algorithmus wählt nicht-deterministisch ein Wort $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$, falls ein solches existiert. Existiert ein endliches Wort $U_0 = W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$, so auch ein Wort der Länge $|U_0| < 2^{2\cdot n}$. Für ein ω-Wort $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$ rät der Algorithmus zunächst ein Wort U_0, $|U_0| < 2^{2\cdot n}$, mit $next_\varepsilon(A,U_0) = \emptyset$ und $next_\varepsilon(B,U_0) \neq \emptyset$. Dann rät der Algorithmus ein Wort U_1 der Länge $2^n - 1$ mit $next_\varepsilon(next_\varepsilon(B,U_0),U_1) \neq \emptyset$. Aus U_0 und U_1 läßt sich ein ω-Wort W mit $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$ konstruieren.

Algorithmus 5.41.

Eingabe: Semi-Automat $\mathcal{A}_T = (\Sigma, Q, E)$ ohne Worttransitionen zu einer Terminologie T; Konzepte A, B aus T

Ausgabe: Es existiert eine Berechnung mit Ausgabe „ja" gdw. $U(B) \not\subseteq U(A) \cup E_{A,\omega} (= U(A) \cup E^f_{A,\omega})$

Es bezeichne n die Größe von Q, M die Menge der Konzepte in T, die auf einem ε-Zyklus liegen, sowie L die Menge der Ausschlußzustände.

```
    T₁:=ε-closure({B});
    T₂:=ε-closure({A});
    z:=0;
    (* Raten von U₀ *)
    while z < 2^(2·n) − 1 and T₂ ∉ L and (T₁ ∩ M = ∅ or T₂ ∩ M ≠ ∅) do begin
        z:=z+1;
        Wähle (nicht-det.) ein R ∈ Σ;
        T₁ := next_ε(T₁, R);
        T₂ := next_ε(T₂, R)
    end;
(1) If T₂ ∉ L then begin
(2)     If T₁ ∩ M ≠ ∅ and T₂ ∩ M = ∅ then Ausgabe „ja“
        else
(3)         if (* z = 2^(2·n) − 1 and *) T₂ = ∅ then begin
                (* Raten von U₁ *)
                z := 0;
                while z < 2^n − 1 do begin
                    z := z + 1;
                    Wähle (nicht-det.) ein R ∈ Σ;
                    T₁ := next_ε(T₁, R);
                    (*T₂ := next_ε(T₂, R) = ∅;*)
                end; (*while*)
(4)             If T₁ ≠ ∅ then Ausgabe „ja“
                else Ausgabe „nein“
            end
            else Ausgabe „nein“
        end(*if*)
    end
    else Ausgabe „nein“.
```
△

Korrektheit: Gibt der Algorithmus „ja“ aus, so sind zwei Fälle zu unterscheiden.

i) (Ausgabe von „ja“ in (2)) Es existiert ein endliches Wort $W = R_1 \cdots R_m \in \Sigma^*$, $m < 2^{2\cdot n}$, so daß für $T_1 := next_\varepsilon(B, W)$ und $T_2 := next_\varepsilon(A, W)$ gilt: $T_1 \cap M \neq \emptyset$ und $T_2 \cap M = \emptyset$. Es existiert demnach ein Konzept $C \in T_1 \cap M$. Damit existiert ein unendlicher Pfad der Form $B, W, C, \varepsilon, C, \varepsilon, C, \ldots$, es gilt also $W \in U(B)$. Wäre $W \in U(A)$, so müßte auch ein (definiertes) Konzept C' existieren, welches auf einem ε-Zyklus liegt, so daß $A, W, C', \varepsilon, C', \varepsilon, C', \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$ ist. Damit wäre aber $C' \in T_2 \cap M$ im Widerspruch zu $T_2 \cap M = \emptyset$. Also ist $W \notin U(A)$. Der Algorithmus sichert für $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$, $0 \leq i \leq m$, ebenfalls $T_{2,i} \notin L$ für alle $0 \leq i \leq m$ (aufgrund der Bedingung in der ersten while-Schleife und wegen (1)). Nach Lemma 5.40 folgt daraus $W \notin E^f_{A,\omega}$. Insgesamt gilt $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$, also $U(B) \not\subseteq U(A) \cup E^f_{A,\omega}$.

Oder:

ii) (Ausgabe von „ja“ in (4)) Es existiert ein Wort $W = R_1 \cdots R_m \in \Sigma^*$, $m = 2^{2\cdot n} - 1 + 2^n - 1$, so daß für $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ und $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$, $0 \leq i \leq m$, gilt: $T_{1,i} \neq \emptyset$ für alle $0 \leq i \leq m$, $T_{2,j} = \emptyset$ für alle $2^{2\cdot n} - 1 \leq j \leq m$ sowie $T_{2,i} \notin L$ für alle $0 \leq i \leq$

m (beachte: $\emptyset \notin L$). Wegen $|2^Q \setminus \{\emptyset\}| = 2^n - 1$ existieren l und r mit $2^{2\cdot n} - 1 \leq l < r \leq m$ und $T_{1,l} = T_{1,r}$. Da für das ω-Wort $\alpha := R_1 \cdots R_{2^{2\cdot n}-1} \cdots R_l(R_{l+1} \cdots R_r)^\omega$ ausgehend von $T_{1,0}$ die Zustandsmengen $T_{1,0}, T_{1,1}, \ldots, T_{1,l}, T_{1,l+1}, \ldots, T_{1,r-1}, T_{1,r}, T_{1,l+1}, \ldots, T_{1,r}, \ldots$ durchlaufen werden, die alle ungleich der leeren Menge sind, folgt $\alpha \in U(B)$ (Lemma 2.10). Für α werden zudem ausgehend von $T_{2,0}$ die Mengen $T_{2,0}, T_{2,1}, \ldots, T_{2,2^{2\cdot n}-2}, \emptyset, \emptyset, \emptyset, \ldots$ durchlaufen. Nach Lemma 2.10 liefert dies $\alpha \notin U(A)$. Da diese Folge von Zustandsmengen keine Ausschlußzustände enthält, gilt nach Lemma 5.40 außerdem $\alpha \notin E^f_{A,\omega}$. Insgesamt gilt somit $\alpha \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$, also $U(B) \not\subseteq U(A) \cup E^f_{A,\omega}$.

Vollständigkeit: Gilt $U(B) \not\subseteq U(A) \cup E^f_{A,\omega}$, so unterscheiden wir die Fälle iii) und iv).

iii) (Nachweis einer Berechnung mit Ausgabe „ja“ in (2)) Es existiert ein endliches Wort $W = R_1 \cdots R_m \in \Sigma^*$ mit $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$. Es seien $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ und $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$ für alle $0 \leq i \leq m$. Ist $m \geq 2^{2\cdot n}$, so existieren wegen $|2^Q \times 2^Q| = 2^{2\cdot n}$ Zahlen l und r mit $0 \leq l < r \leq 2^{2\cdot n}$ sowie $T_{1,l} = T_{1,r}$ und $T_{2,l} = T_{2,r}$. Für das Wort $W' = R_1 \cdots R_l R_{r+1} \cdots R_m$ werden die Zustandsmengenpaare $(T_{1,0}, T_{2,0}), (T_{1,1}, T_{2,1}), \ldots, (T_{1,l}, T_{2,l}), (T_{1,r+1}, T_{2,r+1}), \ldots, (T_{1,m}, T_{2,m})$ durchlaufen. Wegen $T_{1,m} \cap M \neq \emptyset$ (da $W \in U(B)$) folgt analog zu i): $W' \in U(B)$. Analog zu i) liefert $T_{2,m} \cap M = \emptyset$ (wegen $W \notin U(A)$): $W' \notin U(A)$. Da $T_{2,i}$ für kein i, $0 \leq i \leq m$, in L enthalten ist (wegen $W \notin E^f_{A,\omega}$ und Lemma 5.40) folgt $W' \notin E^f_{A,\omega}$. Für W' folgt also $W' \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$. O.E. können wir deshalb von $m = |W| < 2^{2\cdot n}$ ausgehen. In der ersten while-Schleife des Algorithmus wählt man nun im i-ten Durchlauf $R = R_i$. Nach Wahl von W gilt in jedem Durchlauf $T_2 \notin L$. Kann die while-Schleife m-mal durchlaufen werden, so gelten am Ende des m-ten Durchlaufes nach Wahl von W die Aussagen $T_1 \cap M \neq \emptyset$, $T_2 \cap M = \emptyset$ und $T_2 \notin L$. In (2) wird also „ja“ ausgegeben. Wird die while-Schleife nicht m-mal durchlaufen, so muß schon vor dem m-ten Durchlauf $T_1 \cap M \neq \emptyset$ und $T_2 \cap M = \emptyset$ gelten, was in (2) wiederum zur Ausgabe „ja“ führt.
Oder:

iv) (Nachweis einer Berechnung mit Ausgabe „ja“ in (4)) Es existiert ein ω-Wort $W = R_1 R_2 R_3 \cdots \in \Sigma^\omega$ mit $W \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$. Für $T_{1,i} := next_\varepsilon(B, R_1 \cdots R_i)$ und $T_{2,i} := next_\varepsilon(A, R_1 \cdots R_i)$, $i \geq 0$, gilt: (*) $T_{1,i} \neq \emptyset$ für alle $i \geq 0$ (wegen $W \in U(B)$ und Lemma 2.10); außerdem existiert eine Zahl $k \geq 0$ mit $T_{2,k} = \emptyset$ (wegen $W \notin U(A)$ und Lemma 2.10); schließlich gilt $T_{2,i} \notin L$ für alle $i \geq 0$ aufgrund von $W \notin E^f_{A,\omega}$ und Lemma 5.40.

Wegen $|2^Q \times 2^Q| = 2^{2\cdot n}$ existieren l und r mit $0 \leq l < r \leq 2^{2\cdot n}$ sowie $T_{1,l} = T_{1,r}$ und $T_{2,l} = T_{2,r}$. Für $W' := R_1 \cdots R_l R_{r+1} R_{r+2} R_{r+3} \cdots \in \Sigma^\omega$ werden also die Paare $(T_{1,0}, T_{2,0}), (T_{1,1}, T_{2,1}), \ldots, (T_{1,l}, T_{2,l}), (T_{1,r+1}, T_{2,r+1}), (T_{1,r+2}, T_{2,r+2}), \ldots$ durchlaufen. Aussage (*) impliziert $W' \in U(B) \setminus (U(A) \cup E^f_{A,\omega})$. Man kann deshalb für W o.E. $k < 2^{2\cdot n}$ annehmen.

Wir geben nun zu W eine Berechnung des Algorithmus mit Ausgabe „ja“ an. In der ersten while-Schleife wird im i-ten Durchlauf $R = R_i$ gewählt. Wegen $T_{2,i} \notin L$ für alle $i \geq 0$ gilt jeweils $T_2 \notin L$. Tritt der Fall $T_1 \cap M \neq \emptyset$ und $T_2 \cap M = \emptyset$ ein, so wird wegen $T_2 \notin L$ in (2) „ja“ ausgegeben, und wir sind fertig. Ansonsten wird der Algorithmus in (3) fortgesetzt. Nach Voraussetzung gilt $k < 2^{2\cdot n}$. Da die erste while-Schleife $(2^{2\cdot n} - 1)$-mal durchlaufen wurde, gilt somit $T_2 = \emptyset$. Es wird also die zweite Schleife $(2^n - 1)$-mal durchlaufen. Im i-ten Durchlauf wird dabei $R = R_{2^{2\cdot n}-1+i}$ gesetzt. Nach Verlassen der Schleife gilt $T_1 \neq \emptyset$ wegen $T_{1,i} \neq \emptyset$ für alle $i \geq 0$. In (4) wird demnach „ja“ ausgegeben.

Komplexität: Wie erwähnt, kann $T_2 \notin L$ mit polynomialem Zeitaufwand entschieden werden. Damit sieht man leicht, daß Algorithmus 5.41 immer terminiert und mit polynomialem Platz auskommt.

Algorithmus 5.41 zeigt, daß das Problem $U(B) \subseteq U(A) \cup E_{A,\omega}$ mit einem PSPACE-Algorithmus entschieden werden kann. Zusammen mit der PSPACE-Vollständigkeit der Subsumtion bzgl. der lfp-Semantik in $\mathcal{FL}_0$ (vgl. [Baa96], Korollar 27) erhalten wir insgesamt das

Korollar 5.42.
Das Subsumtionsproblem bzgl. der lfp-Semantik in $\mathcal{AL}_0$ ist PSPACE-vollständig. □

Alternative Charakterisierungen von E_A und $E^f_{A,\omega}$ ermöglichen es, die Bedingungen 1.), 2.) und 3.) von Satz 5.38 auf das Inklusionsproblem für reguläre und ω-reguläre Sprachen zu reduzieren. Wir werden dies im folgenden skizzieren.

Lemma 5.43 (alternative Charakterisierung von E_A und $E^f_{A,\omega}$).
Für eine $\mathcal{AL}_0$-Terminologie T, den zugehörige Semi-Automaten $\mathcal{A}_T = (\Sigma, Q, E)$, ein Konzept A sowie die Menge S_P der primitiven Konzepte in T und die Menge M der definierten Konzepte, die auf einem ε-Zyklus liegen, ist

1.) $E_A = \left[\bigcup_{P\in S_P}(L(A,P) \cap L(A,\neg P)) \cup \bigcup_{C\in M} L(A,C)\right] \cdot \Sigma^*$ und

2.) $E^f_{A,\omega} = \left[\bigcup_{P\in S_P}(L(A,P) \cap L(A,\neg P)) \cup \bigcup_{C\in M} L(A,C)\right] \cdot (\Sigma^* \cup \Sigma^\omega)$.

Beweis: Die Aussagen folgen leicht mit Hilfe der Definitionen von E_A und $E^f_{A,\omega}$. □

Die Charakterisierung von E_A in Lemma 5.43 ermöglicht es — ähnlich wie auf Seite 37 —, aus $\mathcal{A}_T$ mit polynomialem Zeitaufwand einen endlichen Automaten zu konstruieren, der E_A bzw. $L_{\mathcal{A}_T}(A,P) \cup E_A$ akzeptiert. Somit läßt sich das Problem $L_{\mathcal{A}_T}(C,P) \subseteq L_{\mathcal{A}_T}(A,P) \cup E_A$ und analog $L_{\mathcal{A}_T}(C,\neg P) \subseteq L_{\mathcal{A}_T}(A,\neg P) \cup E_A$ auf das Inklusionsproblem für reguläre Sprachen reduzieren, ist also, wie erwähnt, in PSPACE enthalten.

Die Charakterisierung von $E^f_{A,\omega}$ in Lemma 5.43 ermöglicht die Entscheidung des Problems $U_{\mathcal{A}_T}(C) \subseteq U_{\mathcal{A}_T}(A) \cup E^f_{A,\omega}$ mit Hilfe von Büchi-Automaten. Die Mengen $U_{\mathcal{A}_T}(C)$, $U_{\mathcal{A}_T}(A)$ und $E^f_{A,\omega}$ enthalten nicht nur unendliche Wörter. Durch Anhängen von $\#^\omega$ (neues Symbol '$\#$') an die endlichen Wörter erhält man die Sprachen $U_{\mathcal{A}_T}(C)^\#$, $U_{\mathcal{A}_T}(A)^\#$ und ${E^f_{A,\omega}}^\#$ unendlicher Wörter über $\Sigma_\# := \Sigma \mathbin{\dot\cup} \{\#\}$. Offensichtlich gilt $(U_{\mathcal{A}_T}(A) \cup E^f_{A,\omega})^\# = U_{\mathcal{A}_T}(A)^\# \cup {E^f_{A,\omega}}^\#$ und $U_{\mathcal{A}_T}(C) \subseteq U_{\mathcal{A}_T}(A) \cup E^f_{A,\omega}$ gdw. $U_{\mathcal{A}_T}(C)^\# \subseteq U_{\mathcal{A}_T}(A)^\# \cup {E^f_{A,\omega}}^\#$.

Um aus $\mathcal{A}_T$ einen Automaten zu konstruieren, der $U_{\mathcal{A}_T}(C)^\#$ bzw. $U_{\mathcal{A}_T}(A)^\#$ akzeptiert, betrachtet man zu $\mathcal{A}_T$ den zugehörigen Semi-Automat mit Buchstaben-Transitionen (vgl. Lemma 2.8). Dieser wird ergänzt durch: einen neuen Zustand Q_{loop}; eine Transition $(Q_{loop}, \#, Q_{loop})$; eine Transition $(q, \#, Q_{loop})$ für jeden Zustand q, von dem aus im ursprünglichen Automaten (mit ε-Transitionen) ein endlicher Pfad mit Label ε zu einem Zustand führt, der auf einem ε-Zyklus liegt. Die Sprachen $U(C)$ bzw. $U(A)$ des so konstruierten Automaten sind $U_{\mathcal{A}_T}(C)^\#$ bzw. $U_{\mathcal{A}_T}(A)^\#$. Ist Q' die Zustandsmenge des Automaten, so ist (vgl. Seite 14) $U_{\mathcal{A}_T}(C)^\# = B(C, Q')$ und $U_{\mathcal{A}_T}(A)^\# = B(A, Q')$.

Offensichtlich entspricht ${E^f_{A,\omega}}^\#$ nach Lemma 5.43 der folgenden Menge:
$\left[\bigcup_{P\in S_P}(L(A,P) \cap L(A,\neg P)) \cup \bigcup_{D\in M} L(A,D)\right] \cdot (\Sigma^\omega \cup \Sigma^* \cdot \{\#\}^\omega)$. Auch zu dieser Sprache

kann in polynomialer Zeit ein akzeptierender Büchi-Automaten konstruiert werden. Dazu wird die Konkatenation von $\{\#\}^\omega$ wie für die Sprachen $U_{\mathcal{A}_T}(C)^\#$ bzw. $U_{\mathcal{A}_T}(A)^\#$ realisiert.

Damit können aus $\mathcal{A}_T$ in polynomialer Zeit Büchi-Automaten konstruiert werden, die $U_{\mathcal{A}_T}(C)^\#$ und $U_{\mathcal{A}_T}(A)^\# \cup {E^f_{A,\omega}}^\#$ akzeptieren. Da das Inklusionsproblem für ω-reguläre Sprachen, gegeben durch Büchi-Automaten, PSPACE-vollständig ist, zeigt dies — wie auch Algorithmus 5.41 —, daß das Problem $U_{\mathcal{A}_T}(C) \subseteq U_{\mathcal{A}_T}(A) \cup E_{A,\omega}$ in PSPACE liegt.

Auch hier hat der Ansatz mit Büchi-Automaten den Nachteil, daß er sich nicht so leicht wie die auf Ausschlußzustände basierenden Algorithmen auf die lfp-Semantik bzgl. $\mathcal{ALN}$ übertragen läßt, da für $\mathcal{ALN}$ die Beschreibung der Mengen E_A und $E^f_{A,\omega}$ durch (in der Größe der Terminologie) polynomiale Automaten schwer fällt.

Kapitel 6

$\mathcal{ALN}$ und endliche Automaten

In diesem Kapitel werden wir Semantik, Inkonsistenz und Subsumtion bzgl. $\mathcal{ALN}$ charakterisieren sowie zugehörige Entscheidungsalgorithmen und Komplexitätsaussagen angeben.

Die Anwesenheit von Zahlenrestriktionen führt in Beweisen im Vergleich zum vorherigen Kapitel zu aufwendigeren Modellen, da zu einem Individuum Nachfolger „gefordert“ werden können. Daneben können widersprüchliche Zahlenrestriktionen Ursache für inkonsistente Konzepte sein. Sie werden deshalb eine Anpassung der in Kapitel 5 eingeführten Begriffe „Ausschlußzustand“ und „Ausschluß“ nötig machen. Wir werden weiterhin sehen, daß die verschiedenen Charakterisierungen und die auf Ausschlußzuständen basierenden Algorithmen jeweils ähnliche Gestalt haben wie diejenigen in Kapitel 5.

6.1 Reduktion von $\mathcal{ALN}$ auf $\mathcal{FLN}$

Wir beschränken uns hier auf die Betrachtung von $\mathcal{FLN}$, d.h. $\mathcal{ALN}$ ohne primitive Negation, da zu einer $\mathcal{ALN}$-Terminologie T eine $\mathcal{FLN}$-Terminologie T' angegeben werden kann, die auf den definierten Konzepten mit T bzgl. der Aussagen zu Inkonsistenz und Subsumtion übereinstimmt; es kann sogar ganz auf primitive Konzepte verzichtet werden.

Die Terminologie T' wird wie folgt (mit linearem Aufwand) aus T konstruiert:
Jedes Vorkommen einer primitiven Negation $\neg P$ in T wird durch eine Maximum-Restriktion $\exists^{\geq 1} R_P$ ersetzt, wobei R_P ein Rollenname ist, der für das primitive Konzept P neu eingeführt wird. Jedes primitive Konzept P in T wird durch die Minimum-Restriktion $\exists^{\leq 0} R_P$ ersetzt. Damit ist T' eine Terminologie ohne primitive Konzepte und ohne primitive Negation[1]. In T' muß nicht jedes primitive Konzept P und jede zugehörige primitive Negation ersetzt werden, jedoch ist wichtig, daß, wenn P und $\neg P$ ersetzt werden, *alle* Vorkommen von P und $\neg P$ ersetzt werden (*konsistentes Ersetzen*). Beachte: T und T' besitzen dieselben definierten Konzepte.

Wird in der Terminologie T von Beispiel 3.2 das primitive Konzept $\mathsf{männlich}$ durch $\exists^{\leq 0}\mathsf{R}_{\mathsf{männlich}}$ konsistent ersetzt, so erhält man die Terminologie T':

$$\begin{array}{rcl} \mathsf{Mensch} & = & \mathsf{Lebewesen} \sqcap \exists^{\geq 2}\mathsf{eltern} \sqcap \exists^{\leq 2}\mathsf{eltern} \sqcap \forall \mathsf{eltern.Mensch} \\ \mathsf{Mann} & = & \mathsf{Mensch} \sqcap \exists^{\leq 0}\mathsf{R}_{\mathsf{männlich}} \end{array}$$

[1] Um auch in T' für primitive Konzepte P Subsumtionsaussagen machen zu können, führt man in T ein neues definiertes Konzept D_P mit zugehörigem Axiom $D_P = \exists^{\leq 0} R_P$ ein.

$$\mathsf{Frau} \quad = \quad \mathsf{Mensch} \sqcap \exists^{\geq 1} \mathsf{R}_{\mathsf{männlich}}$$

Bzgl. T und T' ist der Konzeptterm $\mathsf{Mann} \sqcap \mathsf{Frau}$ inkonsistent. Hätte man das Axiom zu Mann jedoch unverändert beibehalten, d.h. $\mathsf{Mann} = \mathsf{Mensch} \sqcap \mathsf{männlich}$, und das Axiom zu Frau wie angegeben geändert, so wäre T' nicht durch konsistentes Ersetzen aus T entstanden und zudem wäre der Konzeptterm $\mathsf{Mann} \sqcap \mathsf{Frau}$ nicht inkonsistent, da das primitive Konzept $\mathsf{männlich}$ und die zugehörige Rolle $\mathsf{R}_{\mathsf{männlich}}$ unabhängig voneinander interpretiert werden können.

Im folgenden sei

> T eine Terminologie und T' die daraus durch konsistentes Ersetzen gewonnene Terminologie. Die Menge $S = S(T, T')$ enthalte die (konsistent) ersetzten primitiven Konzepte von T, d.h. ein primitives Konzept P ist genau dann in S enthalten, wenn *alle* Vorkommen von P und $\neg P$ in T ersetzt wurden. (6.1)

Als Folge der konsistenten Ersetzung enthält T' kein primitives Konzept P aus S und keine primitive Negation $\neg P$ mit $P \in S$.

Bevor wir zeigen, daß T und T' bzgl. Inkonsistenz und Subsumtion übereinstimmen, beweisen wir zunächst die folgende Hilfsaussage:

Lemma 6.1.
Es sei D ein Konzeptterm bestehend aus Konzepten und Rollen von T. Weiter sei D' der Konzeptterm, den man durch konsistentes Ersetzen bzgl. S (vgl. (6.1)) aus dem Konzeptterm D gewinnt, d.h. D' entsteht aus D, indem für jedes $P \in S$ *alle* Vorkommen von P und $\neg P$ in D durch entsprechende Zahlenrestriktionen ersetzt werden. Damit gelten die folgenden Aussagen:

1.) Zu jeder Interpretation I existiert eine Interpretation I', die sich von I nur durch die Interpretation der Rollen R_P, $P \in S$, unterscheidet, und für die $D^I = D'^{I'}$ gilt. Dabei hängt I' nur von S und I ab.

2.) Zu jeder Interpretation I' existiert eine Interpretation I, die sich von I' höchstens durch die Interpretation der primitiven Konzepte $P \in S$ unterscheidet, und für die $D'^{I'} = D^I$ gilt. Dabei ist I nur von S und I' abhängig.

Beweis: zu 1.) Wir definieren I' wie folgt: $dom(I') := dom(I)$; $C^{I'} := C^I$ für alle Konzepte C in T; $R^{I'} := R^I$ für alle Rollen R in T; $R_P^{I'} := \{(d, d); d \in dom(I) \setminus P^I\}$ für alle primitiven Konzepte $P \in S$ in T.

Offensichtlich ist I' nur von I und S abhängig. Wir zeigen $D'^{I'} = D^I$ durch Induktion über den Aufbau von D:

Ist D ein definiertes Konzept, ein primitives Konzept mit $D \notin S$, eine primitive Negation, die bzgl. S nicht ersetzt wird, oder eine Zahlenrestriktion, so gilt $D' = D$. Nach Definition von I' gilt damit $D'^{I'} = D^I$.

Ist D dagegen ein primitives Konzept P mit $P \in S$, so ist $D' = \exists^{\leq 0} R_P$. Es gilt: $D'^{I'} = (\exists^{\leq 0} R_P)^{I'} = \{d \in dom(I); |R_P^{I'}(d)| = 0\} \overset{\text{Def.}\, R_P^{I'}}{=} \{d \in dom(I); d \in P^I\} = P^I = D^I$.

Für $D = \neg P$ mit $P \in S$ ist $D' = \exists^{\geq 1} R_P$. Somit gilt $D'^{I'} = (\exists^{\geq 1} R_P)^{I'} = \{d \in dom(I); |R_P^{I'}(d)| \geq 1\} \overset{\text{Def.}\, R_P^{I'}}{=} \{d \in dom(I); d \in (\neg P)^I\} = (\neg P)^I = D^I$.

Ist $D = \forall R.C$, so ist $D' = \forall R.C'$, wobei man C' durch konsistentes Ersetzen bzgl. S aus C gewinnt. Nach Induktionsvoraussetzung gilt $C'^{I'} = C^I$. Damit ist $D'^{I'} = (\forall R.C')^{I'} = \{d \in dom(I); R^{I'}(d) \subseteq C'^{I'}\} \overset{\text{IV,Def.I}'}{=} \{d \in dom(I); R^I(d) \subseteq C^I\} = (\forall R.C)^I = D^I$.

Für $D = E \sqcap F$ ist $D' = E' \sqcap F'$. Dabei erhält man E' und F' durch konsistentes Ersetzen bzgl. S aus E bzw. F. Nach Induktionsvoraussetzung gilt $E'^{I'} = E^I$ und $F'^{I'} = F^I$. Damit ist $D'^{I'} = E'^{I'} \sqcap F'^{I'} = E^I \sqcap F^I = (E \sqcap F)^I = D^I$. [2]

zu 2.) Wir definieren I wie folgt: $C^I := C^{I'}$ für alle definierten Konzepte C in T; $P^I := P^{I'}$ für alle primitiven Konzepte $P \notin S$; $P^I := (\exists^{\leq 0} R_P)^{I'}$ für alle $P \in S$; $R^I := R^{I'}$ für alle Rollen R in T.

Offensichtlich ist I nur von I' und S abhängig. Wir zeigen nun $D^I = D'^{I'}$ durch Induktion über den Termaufbau von D':

Ist D' ein definiertes Konzept, ein primitives Konzept P bzw. eine primitive Negation $\neg P$ (also $P \notin S$) oder eine Zahlenrestriktion mit einer Rolle aus T, so ist $D = D'$. Nach Definition von I gilt deshalb $D^I = D'^{I'}$.

Für $D' = \exists^{\leq 0} R_P$ ist $D = P$, insbesondere $P \in S$. Nach Definition von I gilt $D^I = (\exists^{\leq 0} R_P)^{I'} = D'^{I'}$.

Ist $D' = \exists^{\geq 1} R_P$, so ist $D = \neg P$, insbesondere $P \in S$. Also gilt: $D^I = (\neg P)^I = dom(I) \setminus P^I \overset{\text{Def.I}}{=} dom(I) \setminus (\exists^{\leq 0} R_P)^{I'} = (\exists^{\geq 1} R_P)^{I'} = D'^{I'}$.

Für $D' = \forall R.C'$ ist $D = \forall R.C$. Dabei ist C' aus C durch konsistentes Ersetzen bzgl. S entstanden. Nach Induktionsvoraussetzung gilt $C^I = C'^{I'}$. Man zeigt wie in 1.): $D^I = D'^{I'}$.

Für $D' = E' \sqcap F'$ ist $D = E \sqcap F$, wobei E' und F' durch konsistentes Ersetzen bzgl. S aus E bzw. F entstanden sind. Nach Induktionsvoraussetzung gilt $E^I = E'^{I'}$ sowie $F^I = F'^{I'}$.[3] Analog zu 1.) ist damit $D^I = D'^{I'}$. $\square$

Damit zeigen wir nun

Satz 6.2.
Mit den Bezeichnungen aus (6.1) sowie den definierten Konzepten A und B gilt:

1.)
$$\begin{aligned} A \sqsubseteq_T B \quad &\text{gdw.} \quad A \sqsubseteq_{T'} B; \\ A \sqsubseteq_{gfp,T} B \quad &\text{gdw.} \quad A \sqsubseteq_{gfp,T'} B; \\ A \sqsubseteq_{lfp,T} B \quad &\text{gdw.} \quad A \sqsubseteq_{lfp,T'} B. \end{aligned}$$

2.) Für die gfp-, die deskriptive und die lfp-Semantik gilt: A ist inkonsistent bzgl. T gdw. A inkonsistent bzgl. T' ist.

Beweis: zu 1.) „$\Rightarrow$“: Es gelte $A \not\sqsubseteq_{T'} B$ ($A \not\sqsubseteq_{gfp,T'} B$ bzw. $A \not\sqsubseteq_{lfp,T'} B$). Somit existiert ein (gfp-, lfp-)Modell I' von T' und ein Individuum $d \in dom(I')$ mit $d \in A^{I'} \setminus B^{I'}$. Es sei $C = D$ das zu $C = D'$ in T' gehörende Axiom zu C in T. Nach Lemma 6.1, 2.) existiert zu I' eine Interpretation I, so daß $D^I = D'^{I'}$ gilt. Da I nur von I' und S abhängt und

[2] Es sei bemerkt, daß für 1.) die konsistente Ersetzung nicht notwendig ist.

[3] Ist $D' = P \sqcap \exists^{\leq 0} R_P$, d.h. D' ist nicht durch konsistentes Ersetzen aus $D = P \sqcap P$ entstanden, so gilt $((\exists^{\leq 0} R_P)^{I'} =) P^I \neq P^{I'}$, falls $P^{I'} \neq (\exists^{\leq 0} R_P)^{I'}$. Im Beweis würde also $E^I = E'^{I'}$ und $F^I = F'^{I'}$ nicht gelten.

außerdem I und I' auf den definierten Konzepten übereinstimmen gilt für alle Axiome $C = D'$ in T' und $C = D$ in T: $C^{I'} = C^I$ und $D'^{I'} = D^I$. Insbesondere ist I Modell von T. Wegen $d \in A^I \setminus B^I$ ist damit Aussage 1.) für die deskriptive Semantik gezeigt. Für die gfp- und die lfp-Semantik ist noch zu zeigen, daß I gfp-Modell bzw. lfp-Modell ist, falls I' gfp- bzw. lfp-Modell ist. Um dies nachzuweisen (vgl. Behauptung 2) werden für I' zunächst die Rollen R_P, $P \in S$, auf geeignete Weise interpretiert. Wir definieren dazu eine primitive Interpretation J'' wie folgt:

Es sei J' die zu I' gehörende primitive Interpretation. Auf den primitiven Konzepten und Rollen von T stimme J'' mit J' überein. Für alle $P \in S$ sei $R_P^{J''} := \{(d,d);\, d \in (\exists^{\geq 1} R_P)^{J'}\}$.

Für einen Konzeptterm D, der aus Konzepten und Rollen von T aufgebaut ist, sei D' der Konzeptterm, den man durch konsistentes Ersetzen bzgl. S aus D gewinnt. Weiter sei $\underline{A}$ ein Tupel bestehend aus (beliebigen) Extensionen der definierten Konzepte von T. Es sind somit $(J', \underline{A})$ und $(J'', \underline{A})$ Interpretationen. Damit gilt
Behauptung 1: $D'^{(J',\underline{A})} = D'^{(J'',\underline{A})}$.
Denn: Die Rolle R_P für $P \in S$ kommt in D' nur innerhalb von Konzepttermen der Form $\exists^{\leq 0} R_P$ und $\exists^{\geq 1} R_P$ vor. Nach Definition von J'' gilt: $(\exists^{\leq 0} R_P)^{J''} = (\exists^{\leq 0} R_P)^{J'}$ und $(\exists^{\geq 1} R_P)^{J''} = (\exists^{\geq 1} R_P)^{J'}$. Die Behauptung folgt nun leicht per Induktion über den Termaufbau von D'.

Nach Behautpung 1 gilt: $T'_{J'}(\underline{A}) = T'_{J''}(\underline{A})$. Für I' können wir also o.E. annehmen:

$$\text{Die Extensionen der Rollen } R_P,\ P \in S, \text{ bzgl. } J' \text{ sind wie für } J'' \text{ definiert.} \tag{6.2}$$

Es sei nun $\underline{A}$ das durch I definierte Tupel und J die zu I gehörende primitive Interpretation. Das Tupel $\underline{A}$ ist Fixpunkt der Abbildung T_J, da I Modell von T ist. Es gilt
Behauptung 2: Ist I' ein lfp-Modell von T', so ist I ein lfp-Modell von T.
Denn: Wir nehmen an, daß I kein lfp-Modell von T ist. Es existiert also zu $\underline{A}$ ein kleineres Tupel $\underline{A}'$ (komponentenweise Inklusionsordnung), so daß $\underline{A}'$ zusammen mit J Modell von T ist. Nach Lemma 6.1, 1.) existiert zu diesem Modell auch ein Modell I'' von T', in dem die definierten Konzepte durch $\underline{A}'$ definiert sind. Nach Wahl von I' (vgl. (6.2)) sowie nach Konstruktion der Interpretation I aus I' (Lemma 6.1, 2.)) und nach Konstruktion der Interpretation I'' aus I (Lemma 6.1, 1.)) stimmen die primitiven Interpretationen von I'' und I' überein. Nun ist aber das Tupel $\underline{A}'$ kleiner als $\underline{A}$, obwohl I' lfp-Modell von T' ist. Dies ist ein Widerspruch. Damit ist die Behauptung gezeigt.

Man zeigt analog eine entsprechende Aussage für die gfp-Semantik. Da I' und I auf den definierten Konzepten übereinstimmen, gilt $d \in A^I \setminus B^I$. Mit Behauptung 2 folgen damit die Aussagen in 1.) auch für die lfp- und analog für die gfp-Semantik.

„$\Leftarrow$": Es gelte $A \not\sqsubseteq_T B$ ($A \not\sqsubseteq_{gfp,T} B$ bzw. $A \not\sqsubseteq_{lfp,T} B$). Somit existiert ein (gfp- bzw. lfp-)Modell I von T und ein Individuum $d \in dom(I)$ mit $d \in A^I \setminus B^I$. Analog zu „$\Rightarrow$" folgt die Existenz eines Modells I' von T', welches auf den definierten Konzepten mit I übereinstimmt. Damit ist $A \not\sqsubseteq_{T'} B$ gezeigt. Ist I lfp- (gfp-)Modell von T, so zeigt man analog zur Behauptung 2, daß dann I' lfp- (gfp-)Modell von T' ist. Dabei kann hier — anders als für den Beweis der Richtung „$\Rightarrow$", in der I' angepaßt werden mußte (vgl. (6.2)) — die Interpretation I unverändert beibehalten werden.

zu 2.) Es gilt: A ist inkonsistent bzgl. einer Terminologie gdw. A von dem inkonsistenten Term $\exists^{\leq 0} R \sqcap \exists^{\geq 1} R$ (ein für alle drei Semantiken inkonsistenter Konzeptterm) bzgl. der Terminologie subsumiert wird. Damit folgt Aussage 2.) aus 1.). $\square$

Aus Satz 6.2 folgt

Korollar 6.3.
Jede $\mathcal{ALN}$-Terminologie T läßt sich — mit linearer Zeitkomplexität — durch eine $\mathcal{FLN}$-Terminologie T' mit den gleichen definierten Konzepten ersetzen, so daß für die definierten Konzepte die Aussagen zur Inkonsistenz und zur Subsumtion bzgl. T und T' übereinstimmen. □

Wir werden uns aufgrund von Korollar 6.3 bei der Charakterisierung der Semantiken, der Inkonsistenz und der Subsumtion sowie bei algorithmischen Betrachtungen auf $\mathcal{FLN}$ beschränken. Trotzdem werden wir an einigen Stellen auch auf die primitive Negation eingehen. Dabei lassen sich die Aussagen zur primitiven Negation jeweils durch die vorgestellte Reduktion von $\mathcal{ALN}$ auf $\mathcal{FLN}$ ableiten.

Neben primitiven Negationen möchten wir ebenfalls auf Minimum-Restriktionen der Form $\exists^{\leq 0}R$ verzichten. Dies ermöglicht es wiederum, die Mengen E_A und $E_{A,\omega}$ allein durch Ausschlußzustände zu charakterisieren und erlaubt somit, die in Kapitel 5 vorgestellten Algorithmen auch für $\mathcal{FLN}$ bzw. $\mathcal{ALN}$ zu verwenden. Man kann in folgendem Sinne o.E. auf derartige Minimum-Restriktionen verzichten.

Lemma 6.4.
Es sei D ein Konzeptterm in $\mathcal{FLN}$ ($\mathcal{ALN}$). Zu D existiert ein äquivalenter (vgl. Definition 3.4) Konzeptterm D' in $\mathcal{FLN}$ ($\mathcal{ALN}$), der keinen Term der Form $\exists^{\leq 0}R$ enthält. Dabei kann D' mit in der Größe von D linearem Zeitaufwand aus D konstruiert werden.

Beweis: Den Konzeptterm D' erhält man aus D, indem jeder Teilterm der Form $\exists^{\leq 0}R$ in D durch $\forall R.\bot$ ersetzt wird. Das Bottom-Symbol $\bot$ bezeichne dabei einen inkonsistenten Konzeptterm, z.B. $\exists^{\leq 1}R \sqcap \exists^{\geq 2}R$. Offensichtlich erhält man D' aus D mit linearem Zeitaufwand. Für eine Interpretation I gilt $(\forall R.\bot)^I = \{d \in dom(I); |R^I(d)| = 0\} = (\exists^{\leq 0}R)^I$. Mittels Induktion über den Termaufbau von D folgt damit leicht: $D^I = D'^I$. □

Dies führt uns zum

Satz 6.5.
Ist T eine Terminolgie in $\mathcal{FLN}$ ($\mathcal{ALN}$), so ist daraus — mit linearem Zeitaufwand — eine Terminologie T' in $\mathcal{FLN}$ ($\mathcal{ALN}$) mit denselben definierten und primitiven Konzepten sowie denselben Rollen konstruierbar, so daß für jede primitive Interpretation J und jedes Tupel $\underline{A}$ bestehend aus Extensionen der definierten Konzepte $T_J(\underline{A}) = T'_J(\underline{A})$ gilt und T' keine Terme der Form $\exists^{\leq 0}R$ enthält.

Beweis: Wie in Lemma 6.4 ersetzt man jeden Term der Form $\exists^{\leq 0}R$ in T durch $\forall R.\bot$ und erhalten dadurch die Terminologie T'. Es sei $A = D'$ das zu A gehörende Axiom in T' und $A = D$ das entsprechende Axiom in T. Nach Lemma 6.4 ist für eine Interpretation I, definiert durch die primitive Interpretation J und das Tupel $\underline{A}$, $D^I = D'^I$. Damit folgt $T_J(\underline{A}) = T'_J(\underline{A})$. □

Wir werden die Sprache $\mathcal{FLN}$ ($\mathcal{ALN}$), in der keine Terme der Form $\exists^{\leq 0}R$ erlaubt sind, mit $\mathcal{FLN}^r$ ($\mathcal{ALN}^r$) bezeichnen. An den entsprechenden Stellen wird auf die unterschiedliche Behandlung von $\mathcal{FLN}$ und $\mathcal{FLN}^r$ bzw. $\mathcal{ALN}$ und $\mathcal{ALN}^r$ hingewiesen.

Um die Terminologie aus Beispiel 3.3 in eine $\mathcal{ALN}^r$-Terminologie zu überführen, ersetzt man Axiom (3.3) durch:

$$\begin{aligned}\mathsf{Blätter} \quad = \quad & \mathsf{Baum} \sqcap \forall \mathsf{direkter\text{-}nachfolger}.(\exists^{\leq 1}\mathsf{direkter\text{-}nachfolger} \sqcap \\ & \exists^{\geq 2}\mathsf{direkter\text{-}nachfolger})\end{aligned}$$

Zahlenrestriktionen erlauben es auszudrücken, daß Individuen Nachfolger (z.B. eltern in Beispiel 3.2) besitzen müssen. Für alle drei eingeführten Semantiken wird deshalb folgender Begriff von Bedeutung sein:

Definition 6.6 (fordern).
Es sei T eine Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T. Weiter sei $W = R_1 \cdots R_n$ ein endliches Wort und $V = R_1 \cdots R_m$ Präfix von W, d.h $m \leq n$. Das Wort W wird ab V von A *gefordert*, falls für alle Zahlen i mit $m \leq i < n$ Zahlen $m_{i+1} \geq 1$ existieren, so daß $VR_{m+1} \cdots R_i \in L(A, \exists^{\geq m_{i+1}} R_{i+1})$ für alle $m \leq i < n$ gilt. Für das unendliche Wort $W \in \Sigma^\omega$, $W = R_1 R_2 R_3 \cdots$, und das endliche Präfix $V = R_1 \cdots R_m$ von W wird W ab V von A *gefordert*, falls jedes endliche Präfix von W, das V als Präfix hat, von A ab V gefordert wird, d.h.: Für alle $i \geq m$ existieren Zahlen $m_{i+1} \geq 1$, so daß $VR_{m+1} \cdots R_i \in L(A, \exists^{\geq m_{i+1}} R_{i+1})$ für alle $i \geq m$ gilt. Ist $V = \varepsilon$, so sagen wir statt „W wird von A ab ε gefordert" nur „W wird von A gefordert". ◇

Das folgende Lemma macht die Bedeutung von Definition 6.6 deutlich.

Lemma 6.7.
Zusätzlich zu den Bezeichnungen aus Definition 6.6 sei I ein Modell von T sowie d und e_m Individuen in $dom(I)$ mit $dV^I e_m$ und $d \in A^I$. Für den Fall, daß W ein endliches Wort ist, existiert ein $f \in dom(I)$ mit $dV^I e_m (R_{m+1} \cdots R_n)^I f$. Für den Fall, daß W ein unendliches Wort ist, existieren Individuen $e_{m+1}, e_{m+2}, e_{m+3}, \ldots \in dom(I)$ mit $dV^I e_m R^I_{m+1} e_{m+1} R^I_{m+2} e_{m+2} \cdots$.

Beweis: Wir zeigen die Aussage für den Fall $W \in \Sigma^\omega$; der Beweis zu $W \in \Sigma^*$ ist entsprechend zu führen. Die Existenz der Individuen e_i, $i \geq m$, mit $dV^I e_m (R_{m+1} \cdots R_i)^I e_i$ wird per Induktion über i ab m nachgewiesen:
Für $i = m$ gilt die Aussage nach Voraussetzung. Für $i \geq m$ ist $VR_{m+1} \cdots R_i$ Element der Sprache $L(A, \exists^{\geq m_{i+1}} R_{i+1})$. Nach Induktionsvoraussetzung gilt $dV^I e_m (R_{m+1} \cdots R_i)^I e_i$. Aus den Charakterisierungen der Semantiken (vgl. Satz 6.8, 6.27 bzw. 6.33)[4] folgt wegen $d \in A^I$ die Aussage $e_i \in (\exists^{\geq m_{i+1}} R_{i+1})^I$. Wegen $m_{i+1} \geq 1$ besitzt e_i somit einen R_{i+1}-Nachfolger e_{i+1}. □

In den folgenden Abschnitten werden wir uns nun die einzelnen Semantiken bzgl. der Sprache $\mathcal{FLN}$ (bzw. $\mathcal{FLN}^r$) und $\mathcal{ALN}$ (bzw. $\mathcal{ALN}^r$) ansehen. Wir beginnen mit der gfp-Semantik, die sich leichter als die anderen Semantiken durch Automaten charakterisieren läßt.

6.2 Charakterisierung der gfp-Semantik in $\mathcal{ALN}$

Zahlenrestriktionen verhalten sich bzgl. der Charakterisierung der gfp-Semantik durch endliche Automaten wie primitive Konzepte.

Satz 6.8 (Charakterisierung der gfp-Semantik bzgl. $\mathcal{FLN}$).
Es bezeichne T eine Terminologie in $\mathcal{FLN}$, $\mathcal{A}_T$ den zugehörigen Semi-Automaten, I ein gfp-Modell von T sowie A einen Konzeptnamen in T. Für jedes $d \in dom(I)$ gilt:
$d \in A^I$ gdw.

[4] Für den Beweis dieser Sätze wird Lemma 6.7 nicht benötigt.

(P1) für alle primitiven Konzepte P in T, alle Wörter $W \in L(A, P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in P^I$; und

(P2) für alle Maximum-Restriktionen $\exists^{\geq n} R$ in T, alle Wörter $W \in L(A, \exists^{\geq n} R)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\exists^{\geq n} R)^I$; und

(P3) für alle Minimum-Restriktionen $\exists^{\leq n} R$ in T, alle Wörter $W \in L(A, \exists^{\leq n} R)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\exists^{\leq n} R)^I$.

Beweis: analog zum Beweis von Proposition 19 in [Baa96]. Die Zahlenrestriktionen sind im Beweis wie primitive Konzepte zu behandeln. □

Bemerkung 6.9.
Die Reduktion von $\mathcal{ALN}$ auf $\mathcal{FLN}$ liefert für die Charakterisierung der gfp-Semantik, daß in Satz 6.8 bzgl. einer $\mathcal{ALN}$-Terminologie zusätzlich zu (P1) – (P3) die folgende Bedingung gefordert werden muß:
Für alle Terme der Form $\neg P$ in T, alle Wörter $W \in L(A, \neg P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\neg P)^I$. ◇

Mit Satz 6.8 ist es in Beispiel 3.2 für ein Individuum $d \in \mathsf{Mensch}^I$, I gfp-Modell von T, wegen $L(\mathsf{Mensch}, \exists^{\geq 2}\mathsf{eltern}) = \mathsf{eltern}^*$ und $L(\mathsf{Mensch}, \exists^{\leq 2}\mathsf{eltern}) = \mathsf{eltern}^*$ notwendig, daß alle Vorfahren von d genau zwei „eltern“ haben. Durch zyklische Definitionen werden also auch hier Aussagen über die reflexiv-transitive Hülle einer Relation formuliert.

6.2.1 Die gfp-Semantik in $\mathcal{ALN}$ — Inkonsistenz

Würde man in Beispiel 3.2 zusätzlich für die Urgroßeltern eines Menschen mindestens drei Elternteile fordern, d.h. die Konjunktion des Konzepttermes zu Mensch wird um einen Term $\forall \mathsf{eltern\,eltern\,eltern}.\exists^{\geq 3}\mathsf{eltern}$ ergänzt — sicherlich keine sinnvolle Definition —, so ist das Konzept Mensch aufgrund widersprüchlicher Zahlenrestriktionen inkonsistent; einerseits werden genau zwei Elternteile für die Urgroßeltern gefordert, andererseits jedoch mindestens drei. Widersprüchliche Zahlenrestriktionen führen also zu inkonsistenten Konzepten, falls die entsprechenden Rollenfüller (hier Urgroßeltern) gefordert werden.

Wir sprechen von *widersprüchlichen Zahlenrestriktionen* $\exists^{\geq l} R$ und $\exists^{\leq r} R$, falls $l > r$ gilt.

Bevor wir eine formale Charakterisierung der Inkonsistenz angeben, ist es für den Beweis wiederum nützlich, den Begriff des kanonischen Modells zu definieren. Da aufgrund der Maximum-Restriktionen auch Rollen-Nachfolger gefordert werden können, ist es anders als in $\mathcal{AL}_0$ nun nicht mehr ausreichend, für die Extensionen von Rollen leere Mengen zu wählen. Stattdessen bilden die Extensionen der Rollen der kanonischen primitiven Interpretation einen Baum von Individuen.

Definition 6.10 (kanonisches gfp-Modell bzgl. $\mathcal{FLN}$).
Es sei T eine $\mathcal{FLN}$-Terminologie, $\mathcal{A}_T = (\Sigma, Q, E)$ der zugehörige Semi-Automat sowie A ein Konzept in T. Die kanonische primitive Interpretation $J = J(A, d_0)$ zu A und Individuum d_0 sei wie folgt definiert:
$W_0, W_1, W_2, W_3, \ldots$ sei eine Aufzählung (o.E. unendliche; der endliche Fall kann entsprechend behandelt werden) der von A geforderten Wörter mit $W_i \neq W_j$ für alle $i \neq j$, so daß für alle $i, j \geq 0$ mit $i < j$ gilt: $|W_i| \leq |W_j|$. Wird ein Wort von A gefordert, so werden

auch alle Präfixe dieses Wortes von A gefordert. Somit ist $W_0 = \varepsilon$ und mit $|W_i| < |W_{i+1}|$ gilt $|W_{i+1}| = |W_i| + 1$. Mit Hilfe dieser Aufzählung definieren wir nun J induktiv:

$\boldsymbol{J}_0$: $dom(J_0) := \{d_0\}$; $R^{J_0} := \emptyset$ für alle Rollen R in T; die Extensionen der primitiven Konzepte werden erst für J definiert. Nach Definition besitzt J_0 nur endlich viele Individuen (Induktionsanfang).

$\boldsymbol{J}_{i+1}$: Zu W_{i+1} existiert genau ein $j < i+1$ und eine Rolle R aus T mit $W_{i+1} = W_jR$. Es sei $m \geq 1$ maximal mit der Eigenschaft $W_j \in L(A, \exists^{\geq m}R)$, d.h. es existiert kein $m' > m$ mit $W_j \in L(A, \exists^{\geq m'}R)$. Ein derartiges m existiert, da W_{i+1} von A gefordert wird und T nur endlich viele Zahlenrestriktionen enthält. Nach Induktionsvoraussetzung ist $dom(J_i)$ eine endliche Menge. Damit ist auch die Menge $K_{i+1} := \{d_1, \ldots, d_r\} := \{d \in dom(J_i); d_0 W_j^{J_i} d\}$ ($r \geq 0$ Größe von K_{i+1}) der W_j-Nachfolger von d_0 in J_i endlich. Es seien $e_1^1, \ldots, e_m^1, \ldots, e_1^r, \ldots, e_m^r$ $(r \cdot m)$ neue Individuen. Wir erweitern J_i zu J_{i+1} durch: $dom(J_{i+1}) := dom(J_i) \dot\cup \{e_1^1, \ldots, e_m^1, \ldots, e_1^r, \ldots, e_m^r\}$; $S^{J_{i+1}} := S^{J_i}$ für alle Rollen $S \neq R$ und $R^{J_{i+1}} := R^{J_i} \dot\cup \{(d_k, e_l^k); 1 \leq k \leq r,\ 1 \leq l \leq m\}$.

Offensichtlich ist auch $dom(J_{i+1})$ endlich (Induktionsschluß). Damit definieren wir $\boldsymbol{J}$ durch:

$dom(J) := \bigcup_{i \geq 0} dom(J_i)$; $S^J := \bigcup_{i \geq 0} S^{J_i}$ für alle Rollen S in T; für alle primitiven Konzepte P und Individuen $d \in dom(J)$ sei $d \in P^J$ gdw. ein Wort $W \in \Sigma^*$ existiert mit $W \in L(A, P)$ und $d_0 W^J d$.

Das *kanonische gfp-Modell* $I = I(A, d_0)$ zu A und d_0 sei das durch $J(A, d_0)$ und T definierte gfp-Modell. $\diamond$

In Definition 6.10 stellt, anschaulich, jedes J_i einen (endlichen) Baum mit Wurzel d_0 dar. Dabei ist $dom(J_i)$ die Knotenmenge des Baumes. Die Kanten werden durch die Extensionen der Rollen definiert. Der Baum J_0 besteht nur aus der Wurzel d_0. Der Baum J_i wird zum Baum J_{i+1} ergänzt, indem die Knoten von J_i, zu denen Pfade mit Label W_j (zu j vgl. Definition) existieren, jeweils m (vgl. Definition) neue R-Nachfolger erhalten. Die primitive Interpretation J ist „Limes" dieser Bäume. Die Knoten, die zur Extension (bzgl. J) eines primitiven Konzeptes P gehören, werden mit P markiert. Ist die Folge $W_0, W_1, W_2, \ldots$ unendlich, so enthält der Baum zu J unendliche Pfade (Lemma von König).

Um für das kanonische Modell $I = I(A, d_0)$ die Aussage $d_0 \in A^I$ folgern zu können, ist folgende Bedingung hinreichend[5]:

> Es existieren kein Wort $W \in \Sigma^*$ sowie keine widersprüchlichen Zahlenrestriktionen $\exists^{\geq l}R$ und $\exists^{\leq r}R$, $l > r$, so daß W von A gefordert wird und $W \in L(A, \exists^{\geq l}R) \cap L(A, \exists^{\leq r}R)$ gilt. (6.3)

Das folgende Lemma faßt Eigenschaften von J und I zusammen:

Lemma 6.11.
Mit den Bezeichnungen aus Definition 6.10 gilt:

1.) Die kanonische primitive Interpretation J ist ein Baum mit Wurzel d_0, d.h. zu jedem Individuum (Knoten) $e \in dom(J)$ existiert eindeutig ein endliches Wort W mit $d_0 W^J e$, und jeder Knoten $e \in dom(J)$, $e \neq d_0$, besitzt einen eindeutigen direkten Vorgänger, d.h. es existiert eine eindeutige Rolle $S \in \Sigma$ und ein eindeutiges Individuum $d \in dom(J)$ mit $dS^J e$; der Knoten d_0 besitzt keinen Vorgänger.

[5] und nach Satz 6.15 auch notwendig

Jeder Knoten e besitzt nur endlich viele direkte Nachfolger, d.h. J ist ein endlich-verzweigter Baum.
Jedes Wort W mit $d_0 W^J e$ wird von A gefordert.

2.) Für $V \in L(A, \exists^{\geq m} R)$, m maximal mit dieser Eigenschaft, d.h. es existiert kein $m' > m$ mit $V \in L(A, \exists^{\geq m'} R)$, sowie $d_0 V^J d$ für ein $d \in dom(J)$ gilt: $|R^J(d)| = m$.

3.) Wird $W \in \Sigma^*$ von A gefordert, so existiert (mindestens) ein Individuum $e \in dom(J)$ mit $d_0 W^J e$.

4.) Für ein endliches Wort V und ein Individuum d mit $d_0 V^J d$ gilt: $d \in P^J$ gdw. $V \in L(A, P)$.

5.) Ist Bedingung (6.3) erfüllt, so gelten bzgl. A, d_0, T und J die Eigenschaften (P1), (P2) und (P3) aus Satz 6.8. Für das gfp-Modell I folgt nach Satz 6.8 insbesondere $d_0 \in A^I$.

Beweis: zu 1.) Es sei $e \in dom(J)$. Nach Definition von J existiert eine Zahl i mit $e \in dom(J_i)$. Wir zeigen durch vollständige Induktion über i die eindeutige Existenz eines Wortes W mit $d_0 W^J e$ sowie die eindeutige Existenz eines Vorgängers zu $e \neq d_0$. Es sei $e \in dom(J_0)$. Damit ist $e = d_0$. Für $W = \varepsilon$ gilt somit $d_0 W^J e$. Nach Konstruktion der J_{i+1} ist d_0 kein Nachfolger eines Individuums. Somit ist W eindeutig. Es sei nun $e \in dom(J_{i+1})$. Ist $e \in dom(J_i)$, so existiert nach Induktionsvoraussetzung eindeutig ein Wort W mit $d_0 W^J e$. Ist dagegen $e \in dom(J_{i+1}) \setminus dom(J_i)$, so existiert eine eindeutige Zahl j mit $0 \leq j < i$ sowie ein Individuum $d \in dom(J_i)$ und eine Rolle R mit $W_{i+1} = W_j R$ und $d_0 W_j^{J_{i+1}} d R^{J_{i+1}} e$. Nach Induktionsvoraussetzung ist W_j eindeutig mit $d_0 W_j^J d$. Außerdem ist d in J_{i+1} nach Definition eindeutiger Vorgänger von e. Nach Konstruktion von J erhält e in J_k, $k > i+1$, keine weiteren Vorgänger. Damit ist das Wort $W = W_{i+1}$ durch die Beziehung $d_0 W^J e$ eindeutig bestimmt. Es sollte klar sein, daß d_0 Wurzel des Baumes zu J ist.

Wir zeigen nun, daß jedes Individuum $d \in dom(J)$ nur endlich viele direkte Nachfolger besitzt. Zu einem $d \in dom(J)$ existiert eindeutig ein $j \geq 0$ mit $d_0 W_j^J d$ (W_j von A gefordert). Nach Konstruktion von J werden zu d genau dann R-Nachfolger erzeugt, falls $W_j R$ von A gefordert wird, d.h. es existiert ein $i \geq 0$ mit $W_{i+1} = W_j R$. Nach Definition von J_{i+1} werden zu d endlich viele R-Nachfolger (genauer m R-Nachfolger, vgl. Definition) erzeugt. Außerdem werden nur in J_{i+1} R-Nachfolger zu d erzeugt, da nach obigem Beweis nur für das Wort W_j die Beziehung $d_0 W_j^J d$ gilt und jedes von A geforderte Wort in der Folge $W_0, W_1, W_2, \ldots$ genau einmal auftritt, also insbesondere W_{i+1} dort genau einmal auftritt. Demnach besitzt d auch in J nur endlich viele R-Nachfolger. Da die Anzahl der Rollen in der Terminologie T endlich ist, werden zu d insgesamt nur endlich viele Nachfolger erzeugt.

Es gelte $d_0 W^J e$. Für $e = d_0$ ist $W = \varepsilon$, und damit wird W von A gefordert. Für $e \neq d_0$ existiert eindeutig ein $i \geq 0$, so daß $e \in dom(J_{i+1}) \setminus dom(J_i)$ gilt. Nach Konstruktion gilt $d_0 W_{i+1}^{J_{i+1}} e$, also auch $d_0 W_{i+1}^J e$. Nach Definition der Folge $W_0, W_1, W_2, \ldots$ wird A von W_{i+1} gefordert. Wegen $d_0 W_{i+1}^J e$ und $d_0 W^J e$ ist nach obigem Beweis $W = W_{i+1}$, also wird W von A gefordert.

zu 2.) Wegen $d_0 V^J d$ wird nach 1.) das Wort V von A gefordert. Für $m \geq 1$ (m maximal) und $V \in L(A, \exists^{\geq m} R)$ wird auch das Wort VR von A gefordert. Es existieren also Zahlen j, i mit $0 \leq j < i+1$ sowie $W_j = V$ und $W_{i+1} = VR$. Nach Konstruktion werden demnach zu d in J_{i+1} genau m R-Nachfolger erzeugt. Da in keinem J_k, $k > i+1$, R-Nachfolger zu d erzeugt werden, gilt $|R^J(d)| = m$. Ist dagegen $m = 0$, so wird VR nicht von A gefordert; 1.) impliziert damit, daß d keine R-Nachfolger besitzt, da ansonsten VR von A gefordert werden müßte.

zu 3.) Wird $W \in \Sigma^*$ von A gefordert, so existiert (eindeutig) ein $i \geq 0$ mit $W = W_i$. Die Existenz eines Individuums $e \in dom(J)$ mit $d_0 W_i^J e$ zeigen wir durch Induktion über i:
Für $i = 0$ ist $W = \varepsilon$. Für $e = d_0$ gilt somit $d_0 W_i^J e$. Der Induktionsschritt folgt aus Aussage 2.): Zu W_{i+1} existiert eine Zahl j mit $j < i+1$ und eine Rolle R mit $W_{i+1} = W_j R$. Setze $V := W_j$. Nach Induktionsvoraussetzung existiert zu V ein Individuum d mit $d_0 V^J d$. Da W_{i+1} von A gefordert wird, existiert ein $m \geq 1$ mit $V \in L(A, \exists^{\geq m} R)$. Aussage 2.) liefert die Existenz eines Individuums $e \in dom(J)$ mit $d_0 W_j^J d R^J e$.

zu 4.) Wegen 1.) existiert genau ein $V \in \Sigma^*$ mit $d_0 V^J d$. Die Definition der Extensionen primitiver Konzepte liefert damit leicht die behauptete Aussage.

zu 5.) Es gelte Bedingung (6.3). Eigenschaft (P1) bzgl. A, d_0, T und J von Satz 6.8 folgt aus 4.).

Es sei $\exists^{\geq n} R$ eine Maximum-Restriktion in T, $W \in L(A, \exists^{\geq n} R)$ und $e \in dom(J)$ mit $(d_0, e) \in W^J$. Ist $n = 0$, so gilt $e \in (\exists^{\geq 0} R)^J$ wegen $(\exists^{\geq 0} R)^J = dom(J)$. Ist $n \geq 1$, so existiert auch eine maximale Zahl $m \geq 1$ mit $W \in L(A, \exists^{\geq m} R)$. Nach 2.) gilt $|R^J(e)| = m$, also folgt wegen $m \geq n$: $e \in (\exists^{\geq n} R)^J$. Dies zeigt die Gültigkeit von (P2).

Es sei nun $\exists^{\leq n} R$ eine Minimum-Restriktion in T sowie $W \in L(A, \exists^{\leq n} R)$ und $e \in dom(J)$ mit $(d_0, e) \in W^J$. Insbesondere wird nach 1.) das Wort W von A gefordert. Nach Voraussetzung existiert kein $m > n$ mit $W \in L(A, \exists^{\geq m} R)$ (keine widersprüchlichen Zahlenrestriktionen). Nach 2.) gilt somit $|R^J(e)| \leq n$, also $e \in (\exists^{\leq n} R)^J$. Damit gilt (P3).
□

Bemerkung 6.12.
Berücksichtigt man in Definition 6.10 neben widersprüchlichen Zahlenrestriktionen entsprechend auch Widersprüche, die durch P und $\neg P$ für ein primitives Konzept P verursacht werden, so zeigt man für eine $\mathcal{ALN}$-Terminologie ähnlich wie in Lemma 5.6, daß neben (P1) – (P3) aus Satz 6.8 die Bedingung aus Bemerkung 6.9 ebenfalls erfüllt ist. Somit gilt für eine $\mathcal{ALN}$-Terminologie und für das kanonische Modell $I = I(A, d_0)$ aus Definition 6.10 unter Berücksichtigung von P und $\neg P$ ebenfalls $d_0 \in A^I$. ◇

Für die algorithmische Charakterisierung der Inkonsistenz definieren wir

Definition 6.13 (Ausschlußzustand bzgl. der gfp-Semantik in $\mathcal{FLN}$).
Es bezeichne T eine Terminologie und $\mathcal{A}_T = (\Sigma, Q, E)$ den zugehörigen Semi-Automaten ohne Worttransitionen. Die Zustandsmenge $F_0 \subseteq Q$ heißt *Ausschlußzustand* bzgl. $\mathcal{A}_T$ (und bzgl. der gfp-Semantik in $\mathcal{FLN}$), falls eine Zahl $n \geq 0$, ein Wort $R_1 \cdots R_n \in \Sigma^*$ sowie für alle $1 \leq i \leq n$ Zahlen $m_i \geq 1$ und widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$, existieren, so daß für $F_i := next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq n$, gilt: $\exists^{\geq m_i} R_i \in F_{i-1}$ für alle $1 \leq i \leq n$ und $\exists^{\geq l} R$, $\exists^{\leq r} R \in F_n$. ◇

Für eine $\mathcal{ALN}$-Terminologie ist $F_0 \subseteq Q$ auch Ausschlußzustand, falls F_n ein primitives Konzept P und die primitive Negation $\neg P$ enthält.

Für die Entscheidung der Inkonsistenz und der Subsumtion von Konzepten werden wir ein Entscheidungsverfahren für die Menge der Ausschlußzustände benötigen.

Algorithmus 6.14.

Eingabe: Semi-Automat $\mathcal{A}_T = (\Sigma, Q, E)$ ohne Worttransitionen zu einer Terminologie T; $F_0 \subseteq Q$.

Ausgabe: Es existiert eine Berechnung mit Ausgabe „ja“ gdw. F_0 ein Ausschlußzustand ist.

$F := F_0$;
$z := 0$;
while $z < 2^{|Q|}$ do
  if „es existieren widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R, \exists^{\leq r} R \in F$, $l > r$“
    then Ausgabe „ja“;
  if „es existiert eine Zahl $m \geq 1$ und eine Maximum-Restriktion $\exists^{\geq m} R \in F$“
    then „Wähle (nicht-det.) $\exists^{\geq m} R \in F$ mit $m \geq 1$“
    else Ausgabe „nein“;
  $F := next_\varepsilon(F, R)$;
  $z := z + 1$
end;
Ausgabe „nein“. △

Die Korrektheit des Algorithmus folgt leicht aus der Definition des Ausschlußzustandes. Im Beweis der Vollständigkeit benutzt man ähnlich wie für die bisher angegeben Algorithmen ein „Pumping-Lemma-Argument“, um o.E. $n < 2^{|Q|}$ (vgl. Definition 6.13) annehmen zu können. Damit folgt die Existenz einer Berechnung mit Ausgabe „ja“ leicht. Da Algorithmus 6.14 ein NPSPACE-Algorithmus ist, zeigt dies, daß die Menge der Ausschlußzustände (bzgl. der gfp-Semantik in $\mathcal{FLN}$) zu einer Terminologie T mit einem PSPACE-Algorithmus entscheidbar ist.

Satz 6.15 (Charakterisierung: Inkonsistenz bzgl. gfp-Semantik in $\mathcal{FLN}$). Es sei T eine $\mathcal{FLN}$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat ohne Worttransitionen[6] und A ein Konzept in T. Folgende Aussagen sind äquivalent:

1.) A ist T-inkonsistent bzgl. der gfp-Semantik in $\mathcal{FLN}$.

2.) Es existiert ein von A gefordertes Wort $W \in \Sigma^*$, und es existieren widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$, mit $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$.

3.) ε-$closure(\{A\})$ ist ein Ausschlußzustand.

Beweis: Äquivalenz von 1.) und 2.):

„1. $\Leftarrow$ 2.“: Es gelte die rechte Seite der Äquivalenzaussage, und es sei I ein gfp-Modell von T mit $A^I \neq \emptyset$, d.h. es existiert ein Individuum $d \in A^I$. Gemäß Lemma 6.7 existiert ein Individuum $e \in dom(I)$ mit $dW^I e$. Nach (P2) und (P3) von Satz 6.8 muß dann $e \in (\exists^{\geq l} R)^I$ und $e \in (\exists^{\leq r} R)^I$ sein. Dies ist wegen $l > r$ ein Widerspruch.

„1. $\Rightarrow$ 2.“: Die rechte Seite der Äquivalenzaussage gelte nicht. Dann existiert nach Lemma 6.11, 5.) das kanonische gfp-Modell $I = I(A, d_0)$ mit $d_0 \in A^I$. Dies zeigt, daß A

[6] Die Betrachtung eines Semi-Automaten ohne Worttransitionen ist nur für die Charakterisierung durch Ausschlußzustände nötig (vgl. Lemma 2.5 und Beispiel 2.6). Die Äquivalenz von 1.) und 2.) gilt auch für beliebige Semi-Automaten $\mathcal{A}_T$

konsistent ist.
Äquivalenz von 2.) und 3.):

„2. $\Rightarrow$ 3.“: Es sei $W \in \Sigma^*$ ein von A gefordertes Wort $R_1 \cdots R_n$ mit $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$, $l > r$. Es seien $F_i := next_\varepsilon(A, R_1 \cdots R_i)$ für alle $0 \leq i \leq n$. Da W von A gefordert wird, existieren Maximum-Restriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, mit $R_1 \cdots R_i \in L(A, \exists^{\geq m_{i+1}} R_{i+1})$ für alle $0 \leq i < n$. Nach Lemma 2.5 gilt also $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$ für alle $0 \leq i < n$. Außerdem sind wegen $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$ die Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$ in F_n enthalten. Dies zeigt, daß $F_0 = \varepsilon\text{-}closure(\{A\})$ ein Ausschlußzustand ist.

„2. $\Leftarrow$ 3.“: Ist $F_0 := \varepsilon\text{-}closure(\{A\})$ ein Ausschlußzustand, so existiert ein Wort $W = R_1 \cdots R_n \in \Sigma^*$, und so existieren Maximum-Restriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, für alle $1 \leq i \leq n$, so daß für $F_i := next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq n$, gilt: $\exists^{\geq m_i} R_i \in F_{i-1}$ für alle $1 \leq i \leq n$. Außerdem enthält F_n widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$. Dies impliziert zusammen mit Lemma 2.5: $R_1 \cdots R_i \in L(A, \exists^{\geq m_{i+1}} R_{i+1})$ für alle $0 \leq i < n$, d.h. W wird von A gefordert, und $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$. □

Bemerkung 6.16.
Für $\mathcal{ALN}$-Terminologien muß man in 2.) neben widersprüchlichen Zahlenrestriktionen auch Widersprüche durch primitive Konzepte P, also $W \in L(A, P) \cap L(A, \neg P)$, berücksichtigen. Außerdem ist, wie auf Seite 68 dargestellt, der Begriff „Ausschlußzustand“ anzupassen. ◇

Satz 6.15 liefert die Existenz eines PSPACE-Entscheidungsalgorithmus für die Inkonsistenz, da $\varepsilon\text{-}closure(\{A\})$ mit polynomialem Aufwand berechnet werden kann und auf Ausschlußzustand mit einem PSPACE-Algorithmus getestet werden kann.

In Kapitel 7 werden wir mit Hilfe von Aussagen zu Schemata zusätzlich folgenden Satz zeigen:

Satz 6.17 (Härteresultat zur Inkonsistenz).
Das Inkonsistenzproblem für $\mathcal{ALN}$-($\mathcal{FLN}$-)Terminologien bzgl. der gfp- , lfp- und der deskriptiven Semantik ist NP-hart.

Beweis: vgl. Seite 103. □

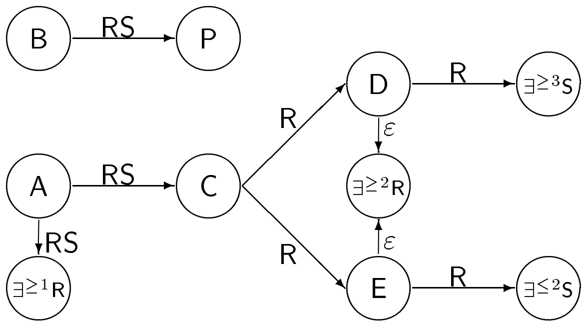
Für sogenannte schwach-azyklische (vgl. Definition 7.14) und azyklische Terminologien kann sogar die NP-Vollständigkeit der Inkonsistenz nachgewiesen werden (vgl. Abschnitt 7.2.2).

Auf Seite 65 haben wir das Konzept Mensch durch eine Bedingung an die Urgroßeltern ergänzt. Für dieses Konzept wird das Wort $\mathsf{eltern}\,\mathsf{eltern}\,\mathsf{eltern}$ von Mensch gefordert. Außerdem gilt $\mathsf{eltern}\,\mathsf{eltern}\,\mathsf{eltern} \in L(\mathsf{Mensch}, \exists^{\leq 2}\mathsf{eltern}) \cap L(\mathsf{Mensch}, \exists^{\geq 3}\mathsf{eltern})$. Damit ist gemäß Satz 6.15 das so definierte Konzept Mensch inkonsistent.

6.2.2 Die gfp-Semantik in $\mathcal{ALN}$ — Subsumtion

Wie im Kapitel 5 führt die Inkonsistenz (führen hier genauer widersprüchliche Zahlenrestriktionen) dazu, daß für die Charakterisierung der Subsumtion ($A \sqsubseteq_{gfp,T} B$) die Inklusionen der Sprachen $L(B, P) \subseteq L(A, P)$, $L(B, \exists^{\geq n} R) \subseteq L(A, \exists^{\geq n} R)$ sowie $L(B, \exists^{\leq n} R) \subseteq L(A, \exists^{\leq n} R)$ für primitive Konzepte P und Zahlenrestriktionen $\exists^{\geq n} R$, $\exists^{\leq n} R$ nur hinreichende Bedingungen darstellen. Die folgende Terminologie macht dies deutlich:

Beispiel 6.18.
Die Terminologie T sei durch den Semi-Automaten $\mathcal{A}_T$ wie folgt gegeben:



Obwohl $\mathsf{RS} \in L(\mathsf{B},\mathsf{P})$ und $\mathsf{RS} \notin L(\mathsf{A},\mathsf{P})$, also $L(\mathsf{B},\mathsf{P}) \not\subseteq L(\mathsf{A},\mathsf{P})$, gilt $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$. Denn: Es sei I ein gfp-Modell von T und d ein Individuum mit $d \in \mathsf{A}^I$. Wegen $\mathsf{RS} \in L(\mathsf{A}, \exists^{\geq 1}\mathsf{R})$ und $\mathsf{RSR} \in L(\mathsf{A}, \exists^{\geq 2}\mathsf{R})$ wird RSRR ab RS von A gefordert. Besitzt d einen RS-Nachfolger, so existiert nach Lemma 6.7 ein Individuum e mit $d(\mathsf{RSRR})^I e$. Wegen $\mathsf{RSRR} \in L(\mathsf{A}, \exists^{\geq 3}\mathsf{S}) \cap L(\mathsf{A}, \exists^{\leq 2}\mathsf{S})$ und Satz 6.8 muß $e \in (\exists^{\geq 3}\mathsf{S})^I$ und $e \in (\exists^{\leq 2}\mathsf{S})^I$ gelten; dies ist ein Widerspruch. Also besitzt d keine RS-Nachfolger. Aus Satz 6.8 folgt damit leicht $d \in \mathsf{B}^I$. ◇

Daß im Beispiel trotz $\{\mathsf{RS}\} = L(\mathsf{B},\mathsf{P}) \not\subseteq L(\mathsf{A},\mathsf{P})$ das Konzept B von A subsumiert wird, ist darauf zurückzuführen, daß RS das Konzept A „ausschließt".

Definition 6.19 (Ausschluß bzgl. der gfp-Semantik in $\mathcal{FLN}^r$).
Es sei T eine $\mathcal{FLN}^r$-Terminologie, $\mathcal{A}_T = (\Sigma, Q, E)$ der zugehörige Semi-Automat sowie A ein Konzept in T. Das Wort $W \in \Sigma^* \cup \Sigma^\omega$ *schließt A aus*, falls ein endliches Präfix $V \in \Sigma^*$ von W existiert und ein Wort $V' \in \Sigma^*$ sowie widersprüchliche Zahlenrestriktionen $\exists^{\geq l}R$, $\exists^{\leq r}R$, $l > r$, so daß $VV' \in L(A, \exists^{\geq l}R) \cap L(A, \exists^{\leq r}R)$ gilt und VV' ab V von A gefordert wird[7]. E_A bezeichne die Menge der endlichen Wörter, die A ausschließen. ◇

Aus Satz 6.15 folgt, daß für ein inkonsistentes Konzept A die Mengen E_A und Σ^* übereinstimmen, da dann bereits ε das Konzept A ausschließt.

Sind zusätzlich neben den Bezeichnungen aus Definition 6.19 ein gfp-Modell I von T sowie Individuen d und e mit $dV^I e$ gegeben, so existiert nach Lemma 6.7, da VV' ab V von A gefordert wird, ein Individuum f mit $dV^I eV'^I f$. Satz 6.8 impliziert also für $d \in A^I$: $f \in (\exists^{\geq l}R)^I$ und $f \in (\exists^{\leq r}R)^I$; dies ist ein Widerspruch. Es ist deshalb $d \notin A^I$. Damit gilt

Lemma 6.20.
Ist $W \in \Sigma^* \cup \Sigma^\omega$ ein A-ausschließendes Wort und das Wort V wie in Definition 6.19 gewählt mit $dV^I e$ für Individuen d und e sowie ein gfp-Modell I, so gilt: $d \notin A^I$. □

Es ist zu beachten, daß wir in Definition 6.19 von einer $\mathcal{FLN}^r$-Terminologie ausgehen. Dies hat folgenden Grund: Es sei $V \in \Sigma^*$ und $R \in \Sigma$ mit VR Präfix von W und $V \in L(A, \exists^{\leq 0}R)$; außerdem seien d, e Individuen mit $dW^I e$. Es existiert also ein f mit $dV^I f$. Für $d \in A^I$ folgt nach Satz 6.8 die Aussage $f \in (\exists^{\leq 0}R)^I$, ein Widerspruch dazu, daß f einen R-Nachfolger besitzt. Somit gilt $d \notin A^I$. Im Sinne von Lemma 6.20 schließt also W das Konzept A aus, d.h. mit $dW^I e$ kann d nicht zur Extension von A gehören. Bei Anwesenheit von Minimum-Restriktionen der Form $\exists^{\leq 0}R$ kann es also neben widersprüchlichen Zahlenrestriktionen

[7] Der Fall $W \in \Sigma^\omega$ wird wiederum erst für die deskriptive Semantik von Bedeutung sein.

weitere Gründe für den Ausschluß eines Konzeptes geben. Wir werden später sehen, daß dann eine Charakterisierung von E_A allein durch Ausschlußzustände nicht möglich ist. Da dadurch die Wiederverwendung der bisher betrachteten Entscheidungalgorithmen erschwert wird, betrachten wir (o.E., nach Satz 6.5) $\mathcal{FLN}^r$-Terminologien.

Es sei T die aus den Axiomen $\mathsf{A} = \forall \mathsf{R}.\exists^{\geq 3}\mathsf{R}$ und $\mathsf{B} = \forall \mathsf{R}.\exists^{\geq 2}\mathsf{R}$ bestehende Terminologie. Trotz $\mathsf{R} \in L(\mathsf{B}, \exists^{\geq 2}\mathsf{R}) \setminus (L(\mathsf{A}, \exists^{\geq 2}\mathsf{R}) \cup E_A)$ gilt $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$, denn für ein Individuum in A wird gefordert, daß alle R-Nachfolger mindestens drei R-Nachfolger besitzen; für Individuen in B werden zu R-Nachfolgern nur zwei R-Nachfolger gefordert. Dies läßt vermuten, daß nicht $L(B, \exists^{\geq l}R) \subseteq (L(A, \exists^{\geq l}R) \cup E_A)$, sondern $L(B, \exists^{\geq l}R) \subseteq (\bigcup_{r \geq l} L(A, \exists^{\geq r}R) \cup E_A)$ notwendig für $A \sqsubseteq_{gfp,T} B$ ist. Für Minimum-Restriktionen reicht eine derartige Bedingung jedoch auch nicht. Ist nämlich ein Wort $W \in L(B, \exists^{\leq l}R) \setminus (\bigcup_{r \leq l} L(A, \exists^{\leq r}R) \cup E_A)$, so sollte es möglich sein, ein gfp-Modell I mit $d \in A^I \setminus B^I$ für ein Individuum d zu definieren. Dazu sollte weiter ein Individuum $e \in dom(I)$ mit $dW^I e$ und $e \notin (\exists^{\leq l}R)^I$ existieren, so daß mit Satz 6.8 tatsächlich $d \notin B^I$ gefolgert werden kann. Dies erfordert, daß e mindestens $(l+1)$ R-Nachfolger besitzt. Wird nun aber WR von A ausgeschlossen, so kann es wegen $d \in A^I$ solche R-Nachfolger zu e nicht geben. Man wird also zusätzlich verlangen müssen, daß A nicht von WR ausgeschlossen wird. Beispiel:

Für die Axiome $\mathsf{A} = \forall \mathsf{RR}.\exists^{\leq 1}\mathsf{R} \sqcap \forall \mathsf{RR}.\exists^{\geq 2}\mathsf{R}$ und $\mathsf{B} = \forall \mathsf{R}.\exists^{\leq 1}\mathsf{R}$ ist $\mathsf{W}{=}\mathsf{R} \in L(\mathsf{B}, \exists^{\leq 1}\mathsf{R}) \setminus (\bigcup_{r \leq 1} L(\mathsf{A}, \exists^{\leq r}\mathsf{R}) \cup E_A)$. Für ein gfp-Modell I und ein Individuum d mit $d \in \mathsf{A}^I$ gilt, daß d keinen (WR=)RR-Nachfolger besitzen kann, da RR von A ausgeschlossen wird. Also kann es zu d auch keinen R-Nachfolger e geben, zu dem mindestens zwei R-Nachfolger existieren, der also nicht in $(\exists^{\leq 1}\mathsf{R})^I$ liegt. Tatsächlich gilt $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$.

Wir werden nun wiederum ein erweitertes kanonisches gfp-Modell I angeben, welches bei Verletzung der (noch einzuführenden) notwendigen Inklusionsbedingungen für die Subsumtion die Subsumtionsbeziehung $A \sqsubseteq_{gfp,T} B$ widerlegt, d.h. für das $d_0 \in A^I \setminus B^I$ gilt. Dazu wird das kanonische Modell zu A und d_0 wie folgt erweitert: Gilt eine Inklusion aufgrund eines Wortes W nicht, d.h. das Wort liegt in der linken aber nicht in der rechten Seite der Inklusionsbeziehung, so wird das kanonische Modell zu A und Individuum d_0, d.h. der Baum mit d_0 als Wurzel, so erweitert, daß ein Pfad von der Wurzel mit Label W in diesem Baum existiert. Widerlegt W die Inklusionsbeziehung einer Minimumrestriktion $\exists^{\leq l}R$, so muß der Pfad zu W zusätzlich um $(l+1)$ R-Nachfolger ergänzt werden, genauer um $(l+1)$ abzüglich der schon vorhandenen. Den so erhaltenen Baum gilt es nun so zu erweitern, daß die in (P1), (P2) und (P3) bzgl. A und d_0 gestellten Bedingungen erfüllt werden. Diese müssen aufgrund der Ergänzungen des kanonischen Baumes durch den Pfad mit Label W bzw. WR — anders als beim kanonischen Baum selbst — nicht mehr gelten.

Definition 6.21 (erweitertes kanonisches gfp-Modell bzgl. $\mathcal{FLN}^r$).
Es sei T eine $\mathcal{FLN}^r$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T, W ein Wort in Σ^*, $r \in \mathbb{N}$ und $R \in \Sigma$. Für $r = 0$ sei $J' = J(A, d_0, W)$ die *erweiterte kanonische primitive Interpretation*; für $r > 0$ bezeichne diese $J' = J(A, d_0, W, R, r)$. Es bezeichne $U_1, U_2, U_3, \ldots$ eine Aufzählung aller Wörter aus Σ^* mit $U_i \neq U_j$ für $i \neq j$ und $|U_i| \leq |U_j|$ für alle $i < j$. Es sei $J = J(A, d_0)$ die kanonische primitive Interpretation (vgl. Definition 6.10) und $I = I(A, d_0)$ das zugehörige gfp-Modell von T. Wir definieren J' induktiv wie folgt:

$\boldsymbol{J}_0$: Wird W von A gefordert, so existiert nach Lemma 6.11, 3.) ein Individuum $d_1 \in dom(J)$ mit $d_0 W^I d_1$. Für $k := r - |R^J(d_1)|$ (falls ≥ 0, sonst $k := 0$) seien $f_1, \ldots, f_k$ neue,

paarweise verschiedene Individuen. Einer einheitlichen Bezeichnung wegen (vgl. dazu den Fall, daß W nicht von A gefordert wird) sei zudem $U := W$. Wir definieren:
$dom(J_0) := dom(J) \dot\cup \{f_1, \ldots, f_k\}$; $S^{J_0} := S^J$ für alle Rollen $S \neq R$; $R^{J_0} := R^J \dot\cup \{(d_1, f_i);\ 1 \leq i \leq k\}$.

Wird W nicht von A gefordert, so existiert ein Präfix U von W maximaler Länge, welches von A gefordert wird. Es existiert demnach ein $d_1 \in dom(J)$ mit $d_0 U^J d_1$. Da W nicht von A gefordert wird, existiert weiterhin ein $V \in \Sigma^+$, $V = R_1 \cdots R_n$, mit $W = U R_1 \cdots R_n$. Es sei $k := r$, und $d_2, d_3, \ldots, d_{n+1}, f_1, \ldots, f_k$ seien neue, paarweise verschiedene Individuen. Wir definieren:
$dom(J_0) := dom(J) \dot\cup \{d_2, \ldots, d_{n+1}, f_1, \ldots, f_k\}$; $S^{J_0} := S^J \dot\cup \{(d_i, d_{i+1});\ 1 \leq i \leq n,\ S = R_i\} \dot\cup \{(d_{n+1}, f_i);\ 1 \leq i \leq k,\ S = R\}$ für alle Rollen S in T.
Die Extensionen der primitiven Konzepte werden erst für J' definiert. Nach Lemma 6.11, 1.) und der Konstruktion von J_0 besitzt jedes Individuum in $dom(J_0)$ nur endlich viele direkte Nachfolger. Damit gilt $|V'^{J_0}(d_0)| < \infty$ für alle $V' \in \Sigma^*$ (Induktionsanfang).

$\boldsymbol{J_{i+1}}$: Für das Wort U_{i+1} der oben definierten Aufzählung gilt nach Induktionsvoraussetzung: $|U_{i+1}^{J_i}(d_0)| < \infty$. Damit ist auch die folgende Menge endlich: $M_{i+1} \subseteq dom(J_i) \times \Sigma \times (\mathbb{N} \setminus \{0\})$ mit $(g, S, m) \in M_{i+1}$ gdw. $d_0 U_{i+1}^{J_i} g$, $U_{i+1} \in L(A, \exists^{\geq m} S)$, $m \geq 1$, m maximal mit dieser Eigenschaft (, d.h. es existiert kein $m' > m$ mit $U_{i+1} \in L(A, \exists^{\geq m'} S)$) und $|S^{J_i}(g)| < m$. Die Menge M_{i+1} enthalte genau die Tripel $(g_1, S_1, m_1), \ldots, (g_{k_{i+1}}, S_{k_{i+1}}, m_{k_{i+1}})$; $k_{i+1} \in \mathbb{N}$ sei dabei die Größe von M_{i+1}. Damit enthält M_{i+1} ein Tripel (g, S, m), falls im Baum zu J_i, um die Bedingung (P2) in Satz 6.8 bzgl. d_0 und A zu erfüllen, zu g noch zusätzlich $(m - |S^{J_i}(g)|)$ S-Nachfolger ergänzt werden müssen. Dazu sei $l_j := |S_j^{J_i}(g_j)|$ (nach Voraussetzung ist $l_j < m_j$) für alle $1 \leq j \leq k_{i+1}$ und weiter $D_{i+1} := \{g_1^1, \ldots, g_{m_1 - l_1}^1, \ldots, g_1^{k_{i+1}}, \ldots, g_{m_{k_{i+1}} - l_{k_{i+1}}}^{k_{i+1}}\}$ eine Menge neuer, paarweise verschiedener Individuen. Wir definieren:
$dom(J_{i+1}) := dom(J_i) \dot\cup D_{i+1}$; $S^{J_{i+1}} := S^{J_i} \dot\cup \bigcup_{1 \leq j \leq k_{i+1}, S = S_j} \{(g_j, g_1^j), \ldots, (g_j, g_{m_j - l_j}^j)\}$ für alle Rollen S in T.
Da nur zu endlich vielen Individuen, nämlich zu $g_1, \ldots, g_{k_{i+1}}$, nur jeweils endlich viele direkte Nachfolger zusätzlich definiert werden, gilt auch für J_{i+1}: $|V'^{J_{i+1}}(d_0)| < \infty$ für alle $V' \in \Sigma^*$ (Induktionsschritt).
Damit wird die erweiterte kanonische primitive Interpretation $\boldsymbol{J'}$ definiert durch:
$dom(J') := \bigcup_{i \in \mathbb{N}} dom(J_i)$; $S^{J'} := \bigcup_{i \in \mathbb{N}} S^{J_i}$ für alle Rollen S in T; für alle primitiven Konzepte P und Individuen d sei $d \in P^{J'}$ gdw. ein $V' \in L(A, P)$ existiert mit $d_0 V'^{J'} d$.
Die *erweiterten kanonischen gfp-Modelle* $I' = I(A, d_0, W)$ bzw. $I' = I(A, d_0, W, R, r)$ sind nun die zu $J' = J(A, d_0, W)$ bzw. $J' = J(A, d_0, W, R, r)$ gehörenden gfp-Modelle von T. $\diamond$

Die erweiterte kanonische primitive Interpretation J' ist in Abbildung 6.1 skizziert (vgl. dazu auch Lemma 6.22, 4.) und 5.)). Sie stellt einen Baum mit Wurzel d_0 dar, der durch schrittweises Erweitern (jeweils an den Endpunkten der Pfade mit Label U_{i+1}, $i \geq 0$, nicht notwendig Blätter des Baumes J_i) aus J_0 entsteht.

Um nun für das erweiterte kanonische Modell $I' = I(A, d_0, W)$ $(r = 0)$ bzw. $I' = I(A, d_0, W, R, r)$ $(r > 0)$ die Aussage $d_0 \in A^{I'}$ beweisen zu können, ist die folgende Bedingung hinreichend[8]:

Das Konzept A ist konsistent und W schließe A nicht aus; für $r > 0$ schließe zudem das Wort WR das Konzept A nicht aus. Für alle $l < r$ gelte $W \notin L(A, \exists^{\leq l} R)$. (6.4)

[8]und, wie man mit Satz 6.15 und Lemma 6.20 leicht zeigt, auch notwendig

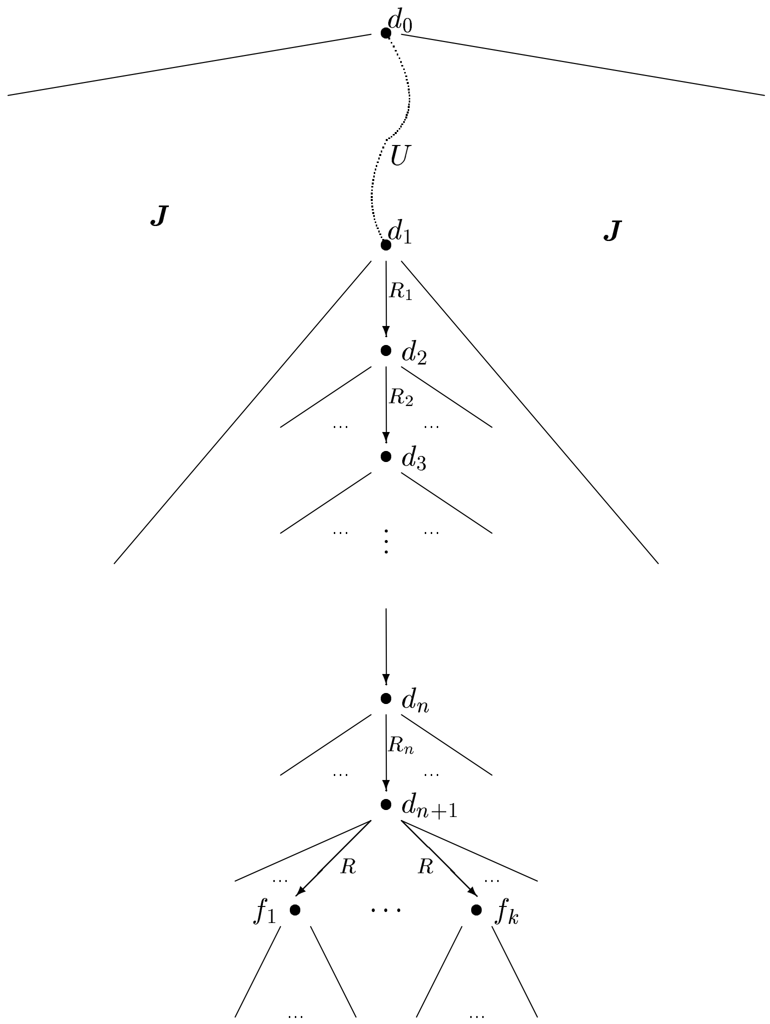


Abbildung 6.1: erweiterte kanonische primitive Interpretation J'

Wir fassen Eigenschaften von J' und I' in folgendem Lemma zusammen:

Lemma 6.22.
Für Definition 6.21 gelten die folgenden Aussagen:

1.) J' ist ein Baum mit Wurzel d_0, d.h. zu jedem $e \in dom(J')$ existiert eindeutig ein (Label) $V' \in \Sigma^*$ mit $d_0 V'^{J'} e$ und zu jedem (Knoten) $e \in dom(J')$, $e \neq d_0$, existiert eindeutig ein Vorgänger, d.h. eine eindeutige Rolle $S \in \Sigma$ und ein eindeutiges Individuum $d \in dom(J')$ mit $dS^{J'}e$; der Knoten d_0 besitzt keinen Vorgänger.

2.) Zu jedem $d \in dom(J')$ werden höchstens für *ein* $i \geq 0$ in J_{i+1} direkte Nachfolger erzeugt. Genauer werden sogar nur endlich viele direkte Nachfolger erzeugt. Das Individuum d besitzt damit nur endlich viele direkte Nachfolger. Der Baum J' ist also endlich-verzweigt.

3.) Ist $r = 0$ und wird W von A gefordert, so ist $S^{J'} = S^J$ für alle Rollen S, $P^{J'} = P^J$ für alle primitiven Konzepte P sowie $dom(J') = dom(J)$, d.h. J' stimmt mit J überein.

4.) Alle Elemente $d \in dom(J') \setminus dom(J)$ sind (direkte oder indirekte) Nachfolger von $d_1 \in dom(J)$. Neben den direkten Nachfolgern aus $dom(J)$ besitzt d_1 lediglich direkte Nachfolger aus $dom(J_0) \setminus dom(J)$, d.h. d_2, falls W nicht von A gefordert wird oder $f_1, \ldots, f_k$, falls W von A gefordert wird.

5.) Für alle Individuen $d, e \in dom(J)$ und $V' \in \Sigma^*$ gilt: $dV'^{J'}e$ gdw. $dV'^{J}e$; mit $d_0V'^{J'}d$ wird V' von A gefordert.

6.) Wird $V' \in \Sigma^*$ von A gefordert, so existiert (mindestens) ein $d \in dom(J)$ mit $d_0V'^{J}d$; nach 5.) also auch $d_0V'^{J'}d$.

7.) Für $V' \in \Sigma^*$ und $d \in dom(J')$ mit $d_0V'^{J'}d$ gilt: $d \in P^{J'}$ gdw. $V' \in L(A, P)$.

8.) Es sei $V' \in L(A, \exists^{\geq m}S)$, m maximal mit dieser Eigenschaft, sowie $d \in dom(J') \setminus \{d_1, \ldots, d_{n+1}\}$ mit $d_0V'^{J'}d$. Dann gilt $|S^{J'}(d)| = m$. Für $r = 0$ gilt dies auch für $d = d_{n+1}$.

9.) Es existiert ein $d \in dom(J')$ mit $d_0W^{J'}d$ und $|R^{J'}(d)| \geq r$.
Ist Bedingung (6.4) erfüllt, so gelten außerdem die Eigenschaften (P1), (P2) sowie (P3) von Satz 6.8 bzgl. A, d_0, T und J'. Da I' gfp-Modell zu J' und T ist, gilt nach Satz 6.8 damit $d_0 \in A^{I'}$.

Beweis: Aussage 1.) zeigt man leicht per Induktion über die induktive Definition von J': Mit Lemma 6.11, 1.) und der Definition von J_0 folgt direkt, daß J_0 ein Baum mit Wurzel d_0 ist. Nach Konstruktion von J_{i+1} wird der Baum J_i an den Endpunkten (g_j) der Pfade mit Label U_{i+1} durch endlich viele (S_j-)Nachfolger $(g_1^j, \ldots, g_{m_j-l_j}^j)$ erweitert. Es sollte klar sein, daß dadurch die Baumeigenschaft erhalten bleibt.

Aussage 2.) folgt aus 1.), da zu $d \in dom(J')$ eindeutig ein $i \geq 0$ existiert mit $d_0U_{i+1}^{J'}d$. Nach Konstruktion von J' werden damit nur in J_{i+1} (endlich viele) direkte Nachfolger zu d erzeugt. Also besitzt d auch in J' nur endlich viele direkte Nachfolger.

Zu 3.): Ist $r = 0$ und wird W von A gefordert, so werden in J_0 nach Konstruktion keine neuen Individuen generiert. Da für J die Bedingung (P2) von Satz 6.8 bereits erfüllt ist, gilt für alle $i \geq 0$: $M_{i+1} = \emptyset$. Damit bleibt der Baum zu J unverändert. Die Extensionen der primitiven Konzepte werden für J' genau wie für J definiert. Insgesamt bleibt somit die primitive Interpretation J erhalten.

Zu 4.): (P2) von Satz 6.8 ist bzgl. A, d_0 und J erfüllt. Damit werden nach Konstruktion von J' für Elemente aus $dom(J) \setminus \{d_1\}$ in J' keine Nachfolger erzeugt; alle Elemente aus $dom(J') \setminus dom(J)$ sind deshalb (direkte oder indirekte) Nachfolger von d_1. Zu d_1 werden in J_{i+1}, $i \geq 0$, keine direkten Nachfolger erzeugt: ansonsten würden eine Rolle S und eine Zahl m existieren mit $U_{i+1} \in L(A, \exists^{\geq m}S)$, m maximal mit dieser Eigenschaft, sowie $d_0U_{i+1}^{J_i}d_1$ und $|S^{J_i}(d_1)| < m$. Nach 1.) gilt außerdem für U wegen $d_0U^{J_i}d_1$ die Gleichheit $U_{i+1} = U$. Nach Definition von U gilt $d_0U^{J}d_1$. Aus $|S^{J_i}(d_1)| < m$ folgt $|S^{J}(d_1)| < m$. Zusammen mit $U \in L(A, \exists^{\geq m}S)$ ist dies ein Widerspruch zur Gültigkeit von (P2) bzgl. A, d_0 und J. Somit liegen alle direkten Nachfolger von d_1 in $dom(J_0)$.

Aussage 5.) folgt aus der Tatsache, daß (der Baum) J' eine Erweiterung von J ist, und aus Lemma 6.11, 1.). Aussage 6.) folgt aus 5.) und Lemma 6.11, 3.). Die Definitionen der Extensionen der primitiven Konzepte liefern zusammen mit 1.) die Gültigkeit von Aussage 7.).

Es gelten nun die Voraussetzungen von Aussage 8.). Für $d \in dom(J)$ gilt nach 5.): $d_0 V'^J d$. Lemma 6.11, 2.) liefert $|S^J(d)| = m$. Nach 4.) werden zu Elementen d aus $dom(J)$ in keinem J_{i+1}, $i \geq 0$, Nachfolger erzeugt; es ist nur möglich, daß in J_0 zu $d = d_1$ direkte R-Nachfolger bzw. R_1-Nachfolger generiert werden. Damit gilt Aussage 8.) für Individuen $d \in dom(J) \setminus \{d_1\}$. Es sei nun $d \in dom(J') \setminus (dom(J) \cup \{d_2, \ldots, d_{n+1}\})$. Damit ist $d \in \{f_1, \ldots, f_k\}$ oder ein in einem J_{j+1}, $j \geq 0$, erzeugtes Individuum. Somit besitzt d in J_0 bzw. J_{j+1} noch keine Nachfolger. Es existiert ein $i \geq 0$ mit $V' = U_{i+1}$. Damit in J_{i+1} S-Nachfolger zu d erzeugt werden, muß $d \in J_i$ gelten. Dies ist für $d \in J_0$ der Fall. Wurde d in J_{j+1} erzeugt, so existiert in J_j ein e, so daß für $V' = V''S'$, $S' \in \Sigma$, die Aussage $d_0 V''^{J_j} e$ gilt und d ein S'-Nachfolger von e ist. Es ist $V'' = U_{j+1}$, also $j+1 < i+1$, da die Aufzählung $U_1, U_2, \ldots$ von Σ^* nach der Länge der Wörter aufsteigend angeordnet ist. Wegen $j+1 \leq i$ ist $d \in J_i$. Nach Konstruktion werden damit in J_{i+1} genau m S-Nachfolger zu d erzeugt, also $|S^{J_{i+1}}(d)| = m$. Nach 2.) werden sonst keine Nachfolger zu d generiert, was $|S^{J'}(d)| = m$ impliziert. Für $r = 0$ gilt, daß auch d_{n+1} in J_0 keine Nachfolger besitzt. Das obige Argument kann damit auch auf $d = d_{n+1}$ angewendet werden. Dies schließt den Beweis zu Aussage 8.) ab.

Zu 9.): Aus der Definition von J_0 folgt direkt die Existenz eines Individuums $d \in dom(J')$ mit $d_0 W^{J'} d$ und $|R^{J'}(d)| \geq r$.

Es sei Bedingung (6.4) erfüllt. Es sind die Eigenschaften (P1), (P2) und (P3) von Satz 6.8 bzgl. A, d_0, T und J' zu zeigen.

Dazu sei P ein primitives Konzept in T, V' ein Wort mit $V' \in L(A, P)$ und $g \in dom(J')$ mit $(d_0, g) \in V'^{J'}$. Damit gilt $g \in P^{J'}$ nach Aussage 7.). Dies zeigt die Gültigkeit von (P1).

Sei nun $\exists^{\geq l} S$ eine Maximum-Restriktion in T, V' ein Wort mit $V' \in L(A, \exists^{\geq l} S)$ und $g \in dom(J')$ mit $(d_0, g) \in V'^{J'}$. Für $l = 0$ folgt wegen $(\exists^{\geq l} S)^{J'} = dom(J')$ sofort $g \in (\exists^{\geq l} S)^{J'}$. Sei nun $l \geq 1$. Es existiert ein $i \geq 0$ mit $V' = U_{i+1}$. Es sei m ($m \geq l$) maximal mit $V' \in L(A, \exists^{\geq m} S)$. Wie im Beweis zu 8.) zeigt man $g \in dom(J_i)$. Gilt $|S^{J_i}(g)| < m$, so ist $(g, S, m) \in M_{i+1}$, und nach Definition von J_{i+1} gilt damit $|S^{J_{i+1}}(g)| \geq m$. Ist $|S^{J_i}(g)| \geq m$, so gilt, da J_{i+1} eine Erweiterung von J_i ist, auch $|S^{J_{i+1}}(g)| \geq m$. Nach Definition von J' folgt also in beiden Fällen $|S^{J'}(g)| \geq m \geq l$, also $g \in (\exists^{\geq l} S)^{J'}$. Damit gilt (P2).

Sei $\exists^{\leq l} S$ eine Minimum-Restriktion in T — wegen T in $\mathcal{FLN}^r$ ist $l > 0$ —, V' ein Wort mit $V' \in L(A, \exists^{\leq l} S)$ sowie $g \in dom(J')$ mit $(d_0, g) \in V'^{J'}$.

Für $g \in dom(J) \setminus \{d_1\}$ gilt wegen 5.): $(d_0, g) \in V'^J$. Da A konsistent ist, folgt mit Lemma 6.11, 5.) die Aussage $g \in (\exists^{\leq l} S)^J$. Nach 4.) besitzt d_1 als einziges Individuum in $dom(J)$ Nachfolger in $dom(J') \setminus dom(J)$. Somit gilt $g \in (\exists^{\leq l} S)^{J'}$, insbesondere $g \in (\exists^{\leq l} S)^{J_0}$.

Es sei nun $g \in (dom(J_0) \setminus dom(J)) \cup \{d_1\}$. Es werde W zunächst von A gefordert, d.h. $g = d_1$ oder $g \in \{f_1, \ldots, f_k\}$. Ist $g = d_1$, so ist $V' = W$ nach 1.). Für $S = R$ gilt nach Voraussetzung $W \notin L(A, \exists^{\leq l'} R)$ für $l' < r$ und damit $l \geq r$. Für $|R^{J_0}(d_1)| = r$ ist somit $d_1 \in (\exists^{\leq l} S)^{J_0}$. Für $|R^{J_0}(d_1)| > r$ ist $R^J(d_1) = R^{J_0}(d_1)$ nach Konstruktion von J_0. Wegen $V' = W$ gilt $d_0 V'^J d_1$. Außerdem ist nach Lemma 6.11, 5.) die Bedingung (P3) bzgl. A, d_0 und J erfüllt. Somit gilt $d_1 \in (\exists^{\leq l} R)^J$; wegen $R^J(d_1) = R^{J_0}(d_1)$ und $S = R$ also auch $g \in (\exists^{\leq l} S)^{J_0}$. Für $S \neq R$ ist $S^J(d_1) = S^{J_0}(d_1)$, was wegen Lemma 6.11, 5.) ebenfalls $d_1 \in (\exists^{\leq l} S)^{J_0}$ liefert. Da die $f_1, \ldots, f_k$ keine Nachfolger in J_0 besitzen, gilt mit $g = f_i$, $1 \leq i \leq k$: $g \in (\exists^{\leq l} S)^{J_0}$.

Es werde nun W nicht von A gefordert, d.h. es gilt $g \in \{d_1, \ldots, d_{n+1}\}$ oder $g \in \{f_1, \ldots, f_k\}$. Da ein d_j, $2 \leq j \leq n$, genau einen Nachfolger in J_0 besitzt, nämlich den R_{j+1}-Nachfolger d_{j+1}, gilt für $g = d_j$, $2 \leq j \leq n$, wegen $l > 0$ auch $g \in (\exists^{\leq l} S)^{J_0}$.

Für $g = d_1$ und $S \neq R_1$ ist $S^J(d_1) = S^{J_0}(d_1)$, woraus mit Lemma 6.11, 5.) wiederum $g \in (\exists^{\leq l} S)^{J_0}$ folgt. Für $g = d_1$ und $S = R_1$ wird das Wort UR_1 nicht von A gefordert, d.h. nach Lemma 6.11, 1.) hat g keinen R_1-Nachfolger in J. In J_0 besitzt g somit genau einen R_1-Nachfolger, nämlich d_2, was wegen $l \geq 1$ wiederum $g \in (\exists^{\leq l} S)^{J_0}$ liefert. Für $g = d_{n+1}$ ist $V' = W$. Nach Konstruktion von J_0, und da W nicht von A gefordert wird, besitzt g in J_0 genau r R-Nachfolger und sonst keine Nachfolger: Für $S = R$ ist wegen $W \notin L(A, \exists^{\leq l'} R)$ für alle $l' < r$, $l \geq r$, und also $g \in (\exists^{\leq l} S)^{J_0}$. Für $S \neq R$ ist $|S^{J_0}(g)| = 0$, also $g \in (\exists^{\leq l} S)^{J_0}$. Die Individuen $f_1, \dots, f_k$ besitzen in J_0 keine Nachfolger, womit für $g = f_i$, $1 \leq i \leq k$, ebenfalls $g \in (\exists^{\leq l} S)^{J_0}$ gilt.

Damit gilt $g \in (\exists^{\leq l} S)^{J_0}$ für alle $g \in dom(J_0)$ mit $d_0 V'^{J'} g$ und $V' \in L(A, \exists^{\leq l} S)$ (Induktionsanfang).

Induktionsvoraussetzung: für alle $g \in dom(J_i)$, $d_0 V'^{J'} g$ sowie $V' \in L(A, \exists^{\leq l} S)$ gilt $g \in (\exists^{\leq l} S)^{J_i}$. Wir zeigen nun im Induktionsschluß:

$$\text{Für alle } g \in dom(J_{i+1}),\ d_0 V'^{J'} g \text{ sowie } V' \in L(A, \exists^{\leq l} S) \text{ gilt } g \in (\exists^{\leq l} S)^{J_{i+1}}. \tag{6.5}$$

Bevor wir (6.5) zeigen, weisen wir zunächst nach, daß damit die Gültigkeit von (P3) folgt: Ist $g \in dom(J')$, $d_0 V'^{J'} g$ sowie $V' \in L(A, \exists^{\leq l} S)$, so existiert ein $j \geq 0$ mit $g \in dom(J_j)$. Für höchstens ein $i \geq 0$ (nach 2.)) werden zu g in J_{i+1}, $j < i + 1$ (Beweis von 8.)), Nachfolger zu g generiert. Nach (6.5) ist $g \in (\exists^{\leq l} S)^{J_{i+1}}$. Damit folgt $g \in (\exists^{\leq l} S)^{J'}$ nach 2.).

Beweis von (6.5): Es sei $g \in dom(J_{i+1})$, $d_0 V'^{J'} g$ und $V' \in L(A, \exists^{\leq l} S)$. Ist $g \in dom(J_{i+1}) \setminus dom(J_i)$, so wurde g in J_{i+1} neu erzeugt und besitzt nach Konstruktion von J_{i+1} keine Nachfolger. Dies liefert $g \in (\exists^{\leq l} S)^{J_{i+1}}$.

Ist $g \in dom(J_i)$, und es existiert kein m mit $(g, S, m) \in M_{i+1}$, so werden zu g keine S-Nachfolger in J_{i+1} erzeugt. Nach Induktionsvoraussetzung gilt $g \in (\exists^{\leq l} S)^{J_i}$, also auch $g \in (\exists^{\leq l} S)^{J_{i+1}}$.

Ist jedoch $(g, S, m) \in M_{i+1}$, d.h. es gilt $d_0 U_{i+1}^{J_i} g$ sowie $U_{i+1} \in L(A, \exists^{\geq m} S)$, $m \geq 1$ maximal mit dieser Eigenschaft, und $|S^{J_i}(g)| < m$, dann werden zu g genau $m - |S^{J_i}(g)|$ neue Nachfolger erzeugt. Somit gilt $|S^{J_{i+1}}(g)| = m$. Ist $g \notin (\exists^{\leq l} S)^{J_{i+1}}$, dann gilt $|S^{J_{i+1}}(g)| > l$, also $m > l$. Wegen $d_0 V'^{J'} g$ und $d_0 U_{i+1}^{J'} g$ folgt nach 1.): $V' = U_{i+1}$. Also gilt $U_{i+1} \in L(A, \exists^{\geq m} S) \cap L(A, \exists^{\leq l} S)$ für $m > l$. Wird U_{i+1} von A gefordert, so muß nach 6.) das Individuum g aus $dom(J)$ sein. Nach 5.) gilt $d_0 U_{i+1}^{J} g$. Da (P2) und (P3) von Satz 6.8 für A und d_0 nach 6.11, 5.) gelten, wäre $g \in (\exists^{\geq m} S)^J \cap (\exists^{\leq l} S)^J = \emptyset$; dies ist aber ein Widerspruch. Also wird U_{i+1} nicht von A gefordert und somit ist $g \in dom(J') \setminus dom(J)$. Nach 4.) sind alle Elemente aus $dom(J') \setminus dom(J)$ (direkte oder indirekte) Nachfolger von d_1. Demnach muß U_{i+1} von der Form UXY sein, $X \in \Sigma^+$ (nach 4.)), $Y \in \Sigma^*$, so daß UX maximales Präfix von WR ($r > 0$) bzw. W (r=0) ist. Nach Konstruktion von J' wird UXY ab UX von A gefordert, da zu einem Endpunkt eines Pfades mit Label U_j, $j \geq 0$, nur dann S'-Nachfolger erzeugt werden, falls $U_j \in L(A, \exists^{\geq m'} S')$ für $m' \geq 1$ gilt. Zusammen mit $UXY \in L(A, \exists^{\geq m} S) \cap L(A, \exists^{\leq l} S)$ und $m > l$ schließt dann WR ($r > 0$) bzw. W ($r = 0$) das Konzept A aus, im Widerspruch zur Voraussetzung. Dies zeigt, daß $g \in (\exists^{\leq l} S)^{J_{i+1}}$ gilt. Damit ist (6.5) gezeigt. □

Es ist zu beachten, daß wir in Lemma 6.22, 9.) (Beweis zu (P3)) benutzt haben, daß T eine $\mathcal{FLN}^r$-Terminologie ist. Wir werden deshalb in der folgenden Charakterisierung der Subsumtion auch von einer $\mathcal{FLN}^r$-Terminologie ausgehen.

Satz 6.23 (Charakterisierung: Subsumtion bzgl. gfp-Semantik in $\mathcal{FLN}^r$). Es bezeichne T eine $\mathcal{FLN}^r$-Terminologie und $\mathcal{A}_T$ den zugehörigen Semi-Automaten, A und B bezeichnen Konzepte in T. Es gilt $A \sqsubseteq_{gfp,T} B$ gdw.

1.) $L(B,P) \subseteq L(A,P) \cup E_A$ für alle primitiven Konzepte P in T; und

2.) $L(B,\exists^{\geq l}R) \subseteq (\bigcup_{r\geq l} L(A,\exists^{\geq r}R) \cup E_A)$ für alle Maximum-Restriktionen der Form $\exists^{\geq l}R$ in T mit $l>0$; und

3.) $L(B,\exists^{\leq l}R)\cdot R \subseteq ((\bigcup_{r\leq l} L(A,\exists^{\leq r}R))\cdot R \cup E_A)$ für alle Minimum-Restriktionen der Form $\exists^{\leq l}R$ in T.

Beweis: „$\Rightarrow$“: Wir nehmen an, daß eine der Inklusionen nicht gilt und zeigen $A \not\sqsubseteq_{gfp,T} B$.

(1) Annahme: $L(B,P) \not\subseteq L(A,P) \cup E_A$ für ein primitives Konzept P in T.
Es existiert demnach ein Wort $W \in L(B,P) \setminus (L(A,P) \cup E_A)$. Wäre A inkonsistent, so wäre $E_A = \Sigma^*$, im Widerspruch zu $W \notin E_A$. Das Konzept A ist also konsistent. Wegen $W \notin E_A$ wird A außerdem nicht von W ausgeschlossen. Somit existiert nach Lemma 6.22, 9.) das erweiterte kanonische Modell $I' = I(A,d_0,W)$ zu A, einem Individuum d_0 und W mit $d_0 \in A^{I'}$. Nach Lemma 6.22, 9.) existiert außerdem ein $d \in dom(I')$ mit $d_0 W^{I'} d$. Lemma 6.22, 7.) liefert mit $W \notin L(A,P)$ auch $d \notin P^{I'}$. Satz 6.8 impliziert zusammen mit $W \in L(B,P)$, $d_0 W^{I'} d$ und $d \notin P^{I'}$: $d_0 \notin B^{I'}$. Dies zeigt $A \not\sqsubseteq_{gfp,T} B$.

(2) Annahme: $L(B,\exists^{\geq l}R) \not\subseteq (\bigcup_{r\geq l} L(A,\exists^{\geq r}R) \cup E_A)$ für eine Maximum-Restriktion $\exists^{\geq l}R$ in T mit $l>0$. Es existiert also ein $W\in\Sigma^*$ mit $W \in L(B,\exists^{\geq l}R)\setminus(\bigcup_{r\geq l} L(A,\exists^{\geq r}R) \cup E_A)$. Wie in (1) ist wegen $W \notin E_A$ das Konzept A konsistent und wird von W nicht ausgeschlossen. Damit existiert das erweiterte kanonische Modell $I' = I(A,d_0,W)$ zu A, einem Individuum d_0 und W mit $d_0 \in A^{I'}$ (Lemma 6.22, 9.)). Außerdem existiert ein $d \in dom(I')$ mit $d_0 W^{J'} d$. Nach Lemma 6.22, 8.) besitzt wegen $W \notin \bigcup_{r\geq l} L(A,\exists^{\geq r}R)$ das Individuum d weniger als l R-Nachfolger ($l>0$) und damit ist $d \notin (\exists^{\geq l}R)^{I'}$. Zusammen mit Satz 6.8 liefert dies bzgl. B und d_0: $d_0 \notin B^{I'}$. Dies zeigt wiederum $A \not\sqsubseteq_{gfp,T} B$.

(3) Annahme: $L(B,\exists^{\leq l}R)\cdot R \not\subseteq ((\bigcup_{r\leq l} L(A,\exists^{\leq r}R))\cdot R \cup E_A)$ für eine Minimum-Restriktion $\exists^{\leq l}R$ in T.
Es existiert also ein $W \in \Sigma^*$ mit $WR \in L(B,\exists^{\leq l}R)\cdot R \setminus ((\bigcup_{r\leq l} L(A,\exists^{\leq r}R))\cdot R \cup E_A)$. Wie in (1) ist wegen $WR \notin E_A$ das Konzept A konsistent und wird von WR nicht ausgeschlossen. Zusammen mit $W \notin \bigcup_{r\leq l} L(A,\exists^{\leq r}R)$ existiert das erweiterte kanonische Modell $I' = I(A,d_0,W,R,l+1)$ mit $d_0 \in A^{I'}$ sowie $|R^{I'}(d)| \geq l+1$ für ein $d \in dom(I')$ mit $d_0 W^{I'} d$ (Lemma 6.22, 9.)). Wegen $W \in L(B,\exists^{\leq l}R)$ und (P3) von Satz 6.8 bzgl. B und d_0 folgt damit $d_0 \notin B^{I'}$; insgesamt also $A \not\sqsubseteq_{gfp,T} B$.

„$\Leftarrow$“: Es gelte die rechte Seite der Äquivalenz. Annahme: $A \not\sqsubseteq_{gfp,T} B$.
Es existiert also ein gfp-Modell I zu T und ein Individuum $d_0 \in dom(I)$ mit $d_0 \in A^I \setminus B^I$. Wegen $d_0 \notin B^I$ gilt mindestens eine der Bedingungen (P1), (P2) und (P3) von Satz 6.8 zu B und d_0 nicht.

(4) Gilt (P1) nicht, so existieren ein primitives Konzept P, ein $W \in L(B,P)$ und ein Individuum $e \in dom(I)$ mit $d_0 W^I e$ und $e \notin P^I$. Wegen $L(B,P) \subseteq L(A,P) \cup E_A$ gilt $W \in L(A,P)$ oder $W \in E_A$. Im Fall $W \in L(A,P)$ folgt mit (P1) für A und d_0 sofort $d_0 \notin P^I$, im Widerspruch zur Annahme. Im Fall $W \in E_A$ folgt $d_0 \notin A^I$ mit Lemma 6.20 und $d_0 W^I e$; ebenfalls im Widerspruch zur Annahme.

(5) Gilt (P2) nicht, so existiert eine Maximum-Restriktion $\exists^{\geq l}R$ in T, ein Wort $W \in L(B,\exists^{\geq l}R)$ sowie ein $e \in dom(I)$ mit $d_0 W^I e$ und $e \notin (\exists^{\geq l}R)^I$; insbesondere ist damit $l>0$.

Wegen $L(B, \exists^{\geq l}R) \subseteq (\bigcup_{r \geq l} L(A, \exists^{\geq r}R) \cup E_A)$ ist $W \in \bigcup_{r \geq l} L(A, \exists^{\geq r}R)$ oder $W \in E_A$. Es sei $r \in \mathbb{N}$ mit $r \geq l$ und $W \in L(A, \exists^{\geq r}R)$. Wegen $|R^I(e)| < l \leq r$ ist $e \notin (\exists^{\geq r}R)^I$, woraus aufgrund von d_0W^Ie und (P2) bzgl. A und d_0 auch $d_0 \notin A^I$ folgt, im Widerspruch zur Annahme. Für $W \in E_A$ folgt wie in (4): $d_0 \notin A^I$; ebenfalls im Widerspruch zur Annahme.

(6) Gilt Bedingung (P3) nicht, so existiert eine Minimum-Restriktion $\exists^{\leq l}R$ in T, ein $W \in L(B, \exists^{\leq l}R)$ sowie ein Individuum $e \in dom(I)$ mit d_0W^Ie und $e \notin (\exists^{\leq l}R)^I$. Wegen $L(B, \exists^{\leq l}R) \cdot R \subseteq ((\bigcup_{r \leq l} L(A, \exists^{\leq r}R)) \cdot R \cup E_A)$ ist $W \in L(A, \exists^{\leq r}R)$ für ein $r \leq l$ oder $WR \in E_A$. Sei zunächst $W \in L(A, \exists^{\leq r}R)$. Wegen $|R^I(e)| > l \geq r$ ist $e \notin (\exists^{\leq r}R)^I$, was zusammen mit d_0W^Ie und (P3) bzgl. A und d_0 auch $d_0 \notin A^I$ impliziert. Wie in (4) liefert $WR \in E_A$ mit d_0W^Ie und $|R^I(e)| > l$ ebenfalls $d_0 \notin A^I$, im Widerspruch zur Annahme. □

Bemerkung 6.24.
Für eine $\mathcal{ALN}^r$-Terminologie ist in Satz 6.23 die folgende Bedingung hinzuzufügen:

$$L(B, \neg P) \subseteq L(A, \neg P) \cup E_A \text{ für alle Terme der Form } \neg P \text{ in } T.$$

Zusätzlich ist die Definition von E_A anzupassen, d.h. neben widersprüchlichen Zahlenrestriktionen sind auch Paare P, $\neg P$ für primitive Konzepte P zu berücksichtigen. Im Beweis betrachtet man nun wie im letzten Kapitel zwei „Versionen“ (Min und Max) des erweiterten kanonischen Modells. ◇

In Beispiel 6.18 ist $E_A = \mathsf{RS}\Sigma^*$, $L(\mathsf{B}, \mathsf{P}) = \{\mathsf{RS}\}$ und $L(\mathsf{B}, \exists^{\geq 1}\mathsf{R}) = L(\mathsf{B}, \exists^{\geq 2}\mathsf{R}) = L(\mathsf{B}, \exists^{\geq 3}\mathsf{S}) = L(\mathsf{B}, \exists^{\leq 2}\mathsf{S}) = \emptyset$. Damit sieht man leicht, daß die Bedingungen in Satz 6.23 bzgl. A und B erfüllt sind und somit $\mathsf{A} \sqsubseteq_{gfp,T} \mathsf{B}$ gilt.

Mit Satz 6.23 folgt in Beispiel 3.3 die intuitiv erwartete Subsumtionsbeziehung $\mathsf{Binärbaum} \sqsubseteq_{gfp,T} \mathsf{Ternärbaum}$. Denn da die Terminologie keine Maximum-Restriktion enthält, ist $E_{\mathsf{Binärbaum}} = E_{\mathsf{Ternärbaum}} = \emptyset$. Zudem sind die Sprachen $L(\mathsf{Binärbaum}, \mathsf{Baum})$, $L(\mathsf{Ternärbaum}, \mathsf{Baum})$, $L(\mathsf{Binärbaum}, \exists^{\leq 2}\mathsf{direkter\text{-}nachfolger})$ sowie die Sprache $L(\mathsf{Ternärbaum}, \exists^{\leq 3}\mathsf{direkter\text{-}nachfolger})$ gleich $\mathsf{direkter\text{-}nachfolger}^*$. Damit sind die in Satz 6.23 geforderten Inklusionen für $\mathsf{Binärbaum} \sqsubseteq_{gfp,T} \mathsf{Ternärbaum}$ erfüllt. Die Subsumtionsbeziehung $\mathsf{Ternärbaum} \sqsubseteq_{gfp,T} \mathsf{Binärbaum}$ gilt dagegen erwartungsgemäß nicht. Wird in Beispiel 3.2 ein Axiom zum Konzept Esel ergänzt, welches analog zum Konzept Mensch definiert ist, so sind die für Mensch und Esel in Satz 6.23 relevanten Sprachen identisch. Die Konzepte Mensch und Esel sind deshalb äquivalent. Dies entspricht nicht der erwarteten Subsumtionsbeziehung, was auf die unzureichende Beschreibung der Konzepte Mensch und Esel zurückzuführen ist oder darauf, daß die gfp-Semantik hier nicht angemessen ist.

Satz 6.23 macht die Entscheidung der Subsumtion durch Testen der Bedingungen 1.), 2.) und 3.) möglich. Dazu werden wir zunächst die Menge E_A — wie in Kapitel 5 — mit Hilfe von Ausschlußzuständen charakterisieren.

Lemma 6.25.
Für die $\mathcal{FLN}^r$-Terminologie T, den Semi-Automaten $\mathcal{A}_T$ ohne Worttransitionen und das Konzept A in T gilt:

$$E_A = \{W \in \Sigma^*; \text{ mit } W \text{ erreicht man von } A \text{ aus einen Ausschlußzustand}\}.$$

Dabei ist „einen Auschlußzustand erreichen“ wie in Definition 5.14 definiert und „Ausschlußzustand“ bezieht sich auf Definiton 6.13.

Beweis: „$\subseteq$“: Sei $W \in E_A$, d.h. es existiert ein Präfix $V \in \Sigma^*$ von W sowie ein Wort $V' \in \Sigma^*$, $V' = R_1 \cdots R_n$, $n \geq 0$, und es existieren widersprüchliche Zahlenrestriktionen $\exists^{\geq l}R$, $\exists^{\leq r}R$, $l > r$, so daß $VV' \in L(A, \exists^{\geq l}R) \cap L(A, \exists^{\leq r}R)$. Außerdem wird das Wort VV' ab V von A gefordert, d.h. es existieren Maximum-Restriktionen $\exists^{\geq m_{i+1}}R_{i+1}$ mit $VR_1 \cdots R_i \in L(A, \exists^{\geq m_{i+1}}R_{i+1}), m_i \geq 1$, für alle $0 \leq i < n$. Für $F_i := next_\varepsilon(A, VR_1 \cdots R_i)$, $0 \leq i \leq n$, gilt nach Lemma 2.5 somit: $\exists^{\geq l}R, \exists^{\leq r}R \in F_n$ und $\exists^{\geq m_i}R_{i+1} \in F_i$ für alle $0 \leq i < n$. Dies zeigt, daß F_0 ein Ausschlußzustand ist, der mit W von A aus erreicht wird.
„$\supseteq$“: Es sei $W \in \Sigma^*$ ein Wort, mit dem man von A aus einen Ausschlußzustand erreicht, d.h. es existiert ein Präfix V von W, so daß $F_0 := next_\varepsilon(A, V)$ ein Ausschlußzustand ist. Nach Definition des Ausschlußzustandes existiert demnach ein $V' = R_1 \cdots R_n \in \Sigma^*$, $n \geq 0$, so daß für $F_i := next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq n$, Zahlenrestriktionen $\exists^{\geq m_i}R_i$ mit $\exists^{\geq m_i}R_i \in F_{i-1}, m_i \geq 1$, für alle $1 \leq i \leq n$ existieren. Außerdem existieren widersprüchliche Zahlenrestriktionen $\exists^{\geq l}R$, $\exists^{\leq r}R$, $l > r$, mit $\exists^{\geq l}R$, $\exists^{\leq r}R \in F_n$. Damit impliziert Lemma 2.5 die Aussagen: $VV' \in L(A, \exists^{\geq l}R) \cap L(A, \exists^{\leq r}R)$ und $VR_1 \cdots R_i \in L(A, \exists^{\geq m_{i+1}}R_{i+1})$, $m_i \geq 1$, für alle $0 \leq i < n$, d.h. VV' wird ab V von A gefordert. Dies liefert $W \in E_A$. $\square$

Wie auf Seite 71 bereits dargestellt, muß für $\mathcal{FLN}$-Terminologien Definition 6.19 ergänzt werden. Ein Wort W schließt ein Konzept A auch dann aus, wenn W ein Präfix VR, $V \in \Sigma^*$, $R \in \Sigma$, enthält mit $V \in L(A, \exists^{\leq 0}R)$. Dieser Fall kann nicht allein durch das Erreichen eines Ausschlußzustandes charakterisiert werden. Im Zustand $next_\varepsilon(A, V)$, der $\exists^{\leq 0}R$ enthalte, hängt es nämlich vom nächsten Zeichen S ($W = VSV'$) in W ab, ob W das Konzept A ausschließt; für $S = R$ wird A von W ausgeschlossen, für $S \neq R$ impliziert $\exists^{\leq 0}R \in next_\varepsilon(A, V)$ jedoch keinen Ausschluß. Ein derartiger Test ist leicht durchführbar. Um jedoch auf umfangreichere Anpassungen der bisher betrachteten Entscheidungsalgorithmen verzichten zu können, betrachten wir $\mathcal{FLN}^r$-Terminologien, die nur kleine Modifikationen der Algorithmen erfordern.

Aufgrund von Lemma 6.25 ist es möglich, Algorithmus 5.16 für die Entscheidung von $L(B, P) \subseteq L(A, P) \cup E_A$ heranzuziehen. Dazu ist lediglich die Verwendung von Definition 6.13 statt der in Abschnitt 5.2 dem Ausschlußzustand zugrundeliegenden Definition nötig. Da das Problem $T_2 \notin \{F \subseteq Q;\ F \text{ Ausschlußzustand}\}$ im Algorithmus 5.16 mit einem PSPACE-Algorithmus entscheidbar ist (vgl. Algorithmus 6.14), ist Algorithmus 5.16 weiterhin ein NPSPACE-Algorithmus.

Algorithmus 5.16 (bzgl. Definition 6.13) kann auch zur Entscheidung von $L(B, \exists^{\geq l}R) \subseteq (\bigcup_{r \geq l} L(A, \exists^{\geq r}R) \cup E_A)$ für eine Maximum-Restriktion $\exists^{\geq l}R$ in T, $l \geq 1$, herangezogen werden. Aufgrund der Vereinigung der regulären Sprachen $L(A, \exists^{\geq l}R)$, $r \geq l$, muß dieser allerdings ein wenig modifiziert werden. Es sei dazu die Menge $Z := \{\exists^{\geq r}R;\ r \geq l$ und $\exists^{\geq r}R$ Maximum-Restriktion in $T\}$. In Algorithmus 5.16 wird der Ausdruck „$(P \notin T_1$ or $P \in T_2)$“ in der while-Bedingung durch „$(\exists^{\geq l}R \notin T_1$ or $T_2 \cap Z \neq \emptyset)$“ ersetzt. In der letzten if-Bedingung wird „$(P \in T_1$ and $P \notin T_2)$“ durch „$(\exists^{\geq l}R \in T_1$ and $T_2 \cap Z = \emptyset)$“ substituiert. Man zeigt nun analog zu Abschnitt 5.2.2, daß für den so modifizierten Algorithmus 5.16 bei Eingabe von $\mathcal{A}_T$ (ohne Worttransitionen) und Konzepten A, B eine Berechnung mit Ausgabe „ja“ existiert gdw. $L(B, \exists^{\geq l}R) \not\subseteq (\bigcup_{r \geq l} L(A, \exists^{\geq r}R) \cup E_A)$. Offensichtlich benötigt dieser Algorithmus ebenfalls nur polynomialen Platz.

Nun geben wir einen (nicht-deterministischen) Algorithmus zur Entscheidung von $L(B, \exists^{\leq l}R) \cdot R \subseteq ((\bigcup_{r \leq l} L(A, \exists^{\leq r}R)) \cdot R \cup E_A)$ für eine Minimum-Restriktion $\exists^{\leq l}R$ in T an. Der Algorithmus wählt, falls ein solches existiert, nicht-deterministisch ein Wort, welches die Inklusion widerlegt. Die Inklusion besagt, daß für jedes $W \in L(B, \exists^{\leq l}R)$ gilt: $W \in \bigcup_{r \leq l} L(A, \exists^{\leq r}R)$ oder $WR \in E_A$. Der Algorithmus muß also nicht-deterministisch ein Wort W mit $W \in L(B, \exists^{\leq l}R)$ sowie $W \notin \bigcup_{r \leq l} L(A, \exists^{\leq r}R)$ und $WR \notin E_A$ wählen. Müßte statt $WR \notin E_A$ nur $W \notin E_A$ gelten, so könnte der oben zu den Maximum-Restriktionen vorgestellte Algorithmus verwendet werden. Es ist dann nur $\exists^{\geq l}R$ durch $\exists^{\leq l}R$ zu ersetzen und $Z := \{\exists^{\leq r}R;\ r \leq l$ und $\exists^{\leq r}R$ eine Zahlenrestriktion in $T\}$ (*) zu setzen. Nun gilt $WR \notin E_A$ gdw. sowohl $W \notin E_A$ ist als auch für $T_2' := next_\varepsilon(A, W)$ gilt, daß $next_\varepsilon(T_2', R)$ kein Ausschlußzustand ist. Damit ist der Algorithmus zu den Maximum-Restriktionen wie folgt zu modifizieren, um ihn auf Minimum-Restriktionen anwenden zu können: Der Algorithmus erhält statt $\exists^{\geq l}R$ die Minimum-Restriktion $\exists^{\leq l}R$ als Eingabe; Z setzt man gemäß (*); die letzte if-Bedingung wird ersetzt durch „$T_2 \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ and $next_\varepsilon(T_2, R) \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ and $\exists^{\leq l}R \in T_1$ and $T_2 \cap Z = \emptyset$“. Die Bedingung $T_2 \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ sichert $W \notin E_A$, und $next_\varepsilon(T_2, R) \notin \{F \subseteq Q;$ F Ausschlußzustand$\}$ sichert $WR \notin E_A$. Ähnlich wie in Abschnitt 5.2.2 zeigt man, daß dieser Algorithmus das Gewünschte leistet, d.h. bei Eingabe von $\mathcal{A}_T$ (ohne Worttransitionen) und Konzepten A, B existiert genau dann eine Berechnung mit Ausgabe „ja“, wenn $L(B, \exists^{\leq l}R) \cdot R \not\subseteq ((\bigcup_{r \leq l} L(A, \exists^{\leq r}R)) \cdot R \cup E_A)$ gilt. Auch dieser Algorithmus ist ein NPSPACE-Algorithmus.

Zusammen mit der Reduktion von $\mathcal{ALN}$ auf $\mathcal{FLN}^r$ (linearer Aufwand, vgl. Korollar 6.3 und Satz 6.5) und der PSPACE-Vollständigkeit des Subsumtionsproblems bzgl. der gfp-Semantik in $\mathcal{FL}_0$ ([Baa96], Korollar 21) erhalten wir

Korollar 6.26.
Das Subsumtionsproblem bzgl. der gfp-Semantik in $\mathcal{ALN}$ ($\mathcal{FLN}$) ist PSPACE-vollständig. □

6.3 Charakterisierung der deskriptiven Semantik in $\mathcal{ALN}$

Wie bei der gfp-Semantik bereiten für die Charakterisierung der deskriptiven Semantik Zahlenrestriktionen keine Probleme. Auch für $\mathcal{FLN}$ ($\mathcal{ALN}$) spielt, wie für $\mathcal{AL}_0$, bei der Charakterisierung der Extension eines Konzeptes der Bezug dieses Konzeptes zu definierten Konzepten eine Rolle.

Satz 6.27 (Charakterisierung der deskriptiven Semantik bzgl. $\mathcal{FLN}$).
Es sei T eine $\mathcal{FLN}$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat, J eine primitive Interpretation sowie $\underline{A}$ ein Tupel mit $T_J(\underline{A}) \subseteq \underline{A}$; weiter bezeichne I das durch J und $\underline{A}$-$gfp(T_J)$ definierte Modell (vgl. Satz 2.4). Für jedes Konzept A und alle Individuen $d \in dom(I)$ gilt: $d \in A^I$ gdw.

(P1) für alle primitiven Konzepte P in T, alle Wörter $W \in L(A, P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in P^I$; und

(P2) für alle Maximum-Restriktionen $\exists^{\geq n}R$ in T, alle Wörter $W \in L(A, \exists^{\geq n}R)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\exists^{\geq n}R)^I$; und

(P3) für alle Minimum-Restriktionen $\exists^{\leq n}R$ in T, alle Wörter $W \in L(A, \exists^{\leq n}R)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\exists^{\leq n}R)^I$; und

(P4) für alle definierten Konzepte B, alle Wörter $W \in L(A, B)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\underline{A})_j$ $(j = index(B))$.

Beweis: analog zum Beweis von Proposition 28 in [Baa96]. Die Zahlenrestriktionen sind im Beweis wie primitive Konzepte zu behandeln. □

Bemerkung 6.28.
Dieser Satz kann genau wie der Satz zur Charakterisierung der gfp-Semantik (Bemerkung 6.9) auf die Sprache $\mathcal{ALN}$ verallgemeinert werden, indem die primitive Negation $\neg P$ analog zu (P1), (P2) bzw. (P3) behandelt wird. ◇

6.3.1 Die deskriptive Semantik in $\mathcal{ALN}$ — Inkonsistenz

Wegen Lemma 5.20 können wir die Charakterisierung der Inkonsistenz bzgl. der gfp-Semantik in $\mathcal{FLN}$ (bzw. $\mathcal{ALN}$) auch für die deskriptive Semantik übernehmen (vgl. Satz 6.15). Insbesondere werden die Begriffe „kanonisches Modell“ (Definition 6.10) und „Ausschlußzustand“ (Definition 6.13) genauso wie für die gfp-Semantik definiert. Satz 6.15 liefert damit, wie bei der gfp-Semantik, daß die Inkonsistenz (bzgl. deskriptiver Semantik) eines Konzeptes mit einem PSPACE-Algorithmus entscheidbar ist. Außerdem gilt nach Satz 6.17 die NP-Härte der Inkonsistenz und für schwach-azyklische sowie azyklische $\mathcal{ALN}$- ($\mathcal{FLN}$-)Terminologien die NP-Vollständigkeit der Inkonsistenz (vgl. Korollar 7.16).

6.3.2 Die deskriptive Semantik in $\mathcal{ALN}$ — Subsumtion

Sind A und B zwei Konzepte in einer Terminologie T, so folgt aus $A \sqsubseteq_T B$ auch die Subsumtionsbeziehung $A \sqsubseteq_{gfp,T} B$. Dies legt nahe, die für die gfp-Semantik in Satz 6.23 formulierten Bedingungen zur Subsumtion für die deskriptive Semantik zu übernehmen. Insbesondere übertragen wir auch die Definitionen von „Ausschluß“ (Definition 6.19) und E_A (Definition 6.19) auf die deskriptive Semantik.

Bemerkung 6.29.
Lemma 6.20 gilt auch für die deskriptive Semantik, da die Aussage ebenso für beliebige Modelle bewiesen werden kann, nicht nur für gfp-Modelle. ◇

Beispiel 5.22 motivierte die Berücksichtigung von Ausschlußwörtern in (5.5), 2.). Werden P und $\neg P$ in diesem Beispiel durch widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$, ersetzt, so legt das Beispiel eine entsprechende Änderung auch für $\mathcal{FLN}$ nahe.

Im Beweis zur Charakterisierung von $A \sqsubseteq_T B$ werden wir ein Modell benötigen, welches die Subsumtionsbeziehung widerlegt. Dazu definieren wir, wie für die gfp-Semantik (vgl. Definition 6.21), eine erweiterte kanonische primitive Interpretation. In Definition 6.21 haben wir die zu einem Konzept A, einem Individuum d_0 und einem endlichen Wort W erweiterte kanonische primitive Interpretation $J' = J(A, d_0, W)$ betrachtet. Aufgrund von Bedingung (P4) aus Satz 6.27 kann es nun — wie auch für $\mathcal{AL}_0$ — nötig sein, zusätzlich ω-Wörter W zu berücksichtigen.

Definition 6.30 (erweiterte kanonische primitive Interpretation).
Es sei T eine $\mathcal{FLN}^r$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T und W ein Wort in $\Sigma^* \cup \Sigma^\omega$. Für $W \in \Sigma^*$ ist die *erweiterte kanonische primitive Interpretation* $J' = J(A, d_0, W)$ zu A, einem Individuum d_0 und W wie in Definition 6.21

erklärt. Für $W = R_1' R_2' R_3' \cdots \in \Sigma^\omega$ definieren wir J' wie folgt:
Wird W von A gefordert, so sei J' die kanonische primitive Interpretation $J(A, d_0)$ (vgl. Definition 6.10). Wird W nicht von A gefordert, so existiert ein endliches Präfix U von W maximaler Länge ($W = U R_1 R_2 R_3 \cdots$), welches von A gefordert wird. Nach Lemma 6.11, 3.) existiert ein Individuum d_1 mit $d_0 U^J d_1$. Es seien $d_2, d_3, \ldots$ neue Individuen. Wir definieren J_0 durch:
$dom(J_0) := dom(J) \dot\cup \{d_2, d_3, d_4, \ldots\}$; $S^{J_0} := S^J \dot\cup \{(d_i, d_{i+1}); i \geq 1, S = R_i\}$ für alle Rollen S in T. Wie in Definition 6.21 gilt $|V'^{J_0}(d_0)| < \infty$ für alle $V' \in \Sigma^*$.

Wir können somit J' wie in Definition 6.21 induktiv durch $J_1, J_2, \ldots$ definieren[9]. $\diamond$

Um für die erweiterte kanonische primitive Interpretation J' auf die Eigenschaften (P1), (P2), (P3) und (P4) von Satz 6.27 bzgl. A, d_0 T und J' schließen zu können, ist die folgende Bedingung hinreichend[10]:

Das Konzept A ist konsistent und wird nicht von dem (endlichen oder unendlichen) Wort W ausgeschlossen (Definition 6.19). (6.6)

Für die erweiterte kanonische primitive Interpretation J' bzgl. eines endlichen Wortes W gelten die in Lemma 6.22 aufgeführten Eigenschaften, da J' wie in Definition 6.21 definiert ist. Es ist dabei zu beachten, daß wir noch kein Modell zu J' definiert haben. Dies wird — wie in Abschnitt 5.3.2 — erst im Beweis zur Charakterisierung der Subsumtion formuliert. Die Aussage $d_0 \in A^{I'}$ aus Lemma 6.22, 9.) kann an dieser Stelle deshalb noch nicht getroffen werden. Für $W \in \Sigma^\omega$ zeigt man die Aussagen des Lemmas analog zum Fall $W \in \Sigma^*$, wobei die Eigenschaften 8.) und 9.) wie folgt formuliert werden:

8.) Es sei $V' \in L(A, \exists^{\geq m} S)$, m maximal mit dieser Eigenschaft, sowie $d \in dom(J') \setminus \{d_1, d_2, d_3, \ldots\}$ mit $d_0 V'^{J'} d$. Dann gilt $|S^{J'}(d)| = m$.

9.) Es existieren Individuen $d_1, d_2, d_3, \ldots \in dom(J')$ mit $d_0 U^{J'} d_1 R_1^{J'} d_2 R_2^{J'} d_3 \cdots$
Ist Bedingung (6.6) erfüllt, so gelten die Eigenschaften (P1), (P2) und (P3) von Satz 6.27 (oder auch Satz 6.8) bzgl. A, d_0, T und J'.

Damit können wir den folgenden Satz zeigen:

Satz 6.31 (Charakterisierung: Subsumtion bzgl. deskr. Semantik in $\mathcal{FLN}$). Es bezeichne T eine $\mathcal{FLN}^r$-Terminologie und $\mathcal{A}_T$ den zugehörigen Semi-Automaten, A und B bezeichnen Konzepte in T. Es gilt $A \sqsubseteq_T B$ gdw.

1.) $L(B, P) \subseteq L(A, P) \cup E_A$ für alle primitiven Konzepte P in T; und

2.) $L(B, \exists^{\geq l} R) \subseteq (\bigcup_{r \geq l} L(A, \exists^{\geq r} R) \cup E_A)$ für alle Maximum-Restriktionen der Form $\exists^{\geq l} R$ in T mit $l > 0$; und

3.) $L(B, \exists^{\leq l} R) \cdot R \subseteq ((\bigcup_{r \leq l} L(A, \exists^{\leq r} R)) \cdot R \cup E_A)$ für alle Minimum-Restriktionen der Form $\exists^{\leq l} R$ in T; und

4.) für alle definierten Konzepte C und alle unendlichen Pfade der Form $B, U_0, C, U_1, C, U_2, C, \ldots$, existiert ein $k \geq 0$, so daß $U_0 \cdots U_k \in L(A, C) \cup E_A$.

[9] Die primitive Interpretation $J' = J(A, d_0, W, R, r)$ für $R \in \Sigma$ und $r > 0$ wird nicht benötigt (vgl. Beweis zu Satz 6.31 („$\Rightarrow$“)).

[10] und notwendig (nach Satz 6.15 und Bemerkung 6.29)

Beweis: „$\Leftarrow$“: Es gelte die rechte Seite der Behauptung. Zusätzlich sei I ein Modell von T, das durch die primitive Interpretation J und den Fixpunkt $\underline{A}$ von T_J definiert ist. Offensichtlich gilt $T_J(\underline{A}) \subseteq \underline{A}$ und $\underline{A} = \underline{A}\text{-}gfp(T_J)$. Es sei $d \in dom(I)$ mit $d \notin B^I$. Zu zeigen ist $d \notin A^I$.

Wegen $d \notin B^I$ gilt nach Satz 6.27, daß mindestens eine der Bedingungen (P1), (P2), (P3) oder (P4) nicht gilt. Die Fälle, daß (P1), (P2) oder (P3) nicht gelten, zeigt man unter Verwendung von Satz 6.27 (statt Satz 6.8) analog zum Beweis von Satz 6.23 („$\Leftarrow$“). Der Beweis zu (P4) kann unter Verwendung von Satz 6.27 (statt Satz 5.19) und Bemerkung 6.29 (statt Lemma 5.21) analog zum Beweis von Satz 5.25 („$\Leftarrow$“) geführt werden.

„$\Rightarrow$“: Es gelte $A \sqsubseteq_T B$; insbesondere gilt damit $A \sqsubseteq_{gfp,T} B$. Da E_A für die gfp-Semantik und die deskriptive Semantik gleich definiert sind, stimmen die Aussagen 1.), 2.) und 3.) von Satz 6.23 und Satz 6.31 überein. Somit ist nur noch 4.) zu zeigen. Wie im Beweis zu Satz 5.25 („$\Rightarrow$“) nimmen wir dazu an, daß 4.) nicht gilt und konstruieren ein Modell, welches im Widerspruch zu $A \sqsubseteq_T B$ steht. Unter Verwendung des Satzes 6.27 (statt Satz 5.19) und Lemma 6.22 bzw. den Aussagen 8.) und 9.) von Seite 83 (statt Lemma 5.24) ist dies analog zum Beweis von Satz 5.25 („$\Rightarrow$“) möglich. Insbesondere wird nur die in Definition 6.30 eingeführte erweiterte kanonische primitive Interpretation $J' = J(A, d_0, W)$ benötigt, nicht jedoch die Variante $J' = J(A, d_0, W, R, r)$ für $R \in \Sigma$ und $r > 0$, die aus diesem Grund auch nicht eingeführt wurde. □

Satz 6.31 kann analog zu Bemerkung 6.24 auf $\mathcal{ALN}$-Terminologien verallgemeinert werden.

Auf Seite 79 haben wir festgestellt, daß die Konzepte Mensch und Esel bzgl. der gfp-Semantik äquivalent sind. Satz 6.31 liefert, daß diese Konzepte bzgl. der deskriptiven Semantik unvergleichbar sind. Denn: Der Semi-Automat enthält die unendlichen Pfade $\mathsf{Mensch}, \mathsf{eltern}, \mathsf{Mensch}, \mathsf{eltern}, \mathsf{Mensch}, \ldots$ bzw. $\mathsf{Esel}, \mathsf{eltern}, \mathsf{Esel}, \mathsf{eltern}, \mathsf{Esel}, \ldots$. Wegen $E_{\mathsf{Mensch}} = E_{\mathsf{Esel}} = \emptyset$ und $L(\mathsf{Mensch}, \mathsf{Esel}) = L(\mathsf{Esel}, \mathsf{Mensch}) = \emptyset$ ist für kein $k \geq 0$ $\mathsf{eltern}^k \in L(\mathsf{Mensch}, \mathsf{Esel}) \cup E_{\mathsf{Mensch}}$ bzw. $\mathsf{eltern}^k \in L(\mathsf{Esel}, \mathsf{Mensch}) \cup E_{\mathsf{Esel}}$. Obwohl die deskriptive Semantik also in diesem Beispiel die erwarteten Subsumtionsbeziehungen liefert, ist dies für Beispiel 3.3 — im Gegensatz zur gfp-Semantik (vgl. Seite 79) — nicht der Fall. Es ist $E_{\mathsf{Binärbaum}} = E_{\mathsf{Ternärbaum}} = \emptyset$; außerdem $L(\mathsf{Binärbaum}, \mathsf{Ternärbaum}) = L(\mathsf{Ternärbaum}, \mathsf{Binärbaum}) = \emptyset$. Zu den unendlichen Pfaden Binärbaum, direkter-nachfolger, Binärbaum, direkter-nachfolger,... und Ternärbaum, direkter-nachfolger, Ternärbaum, direkter-nachfolger,... existiert kein $k \geq 0$, so daß $\mathsf{direkter\text{-}nachfolger}^k$ Element von $L(\mathsf{Binärbaum}, \mathsf{Ternärbaum}) \cup E_{\mathsf{Binärbaum}}$ bzw. $L(\mathsf{Ternärbaum}, \mathsf{Binärbaum}) \cup E_{\mathsf{Ternärbaum}}$ ist. Nach Satz 6.31 sind die Konzepte Binärbaum und Ternärbaum somit unvergleichbar, obwohl man intuitiv $\mathsf{Binärbaum} \sqsubseteq_T \mathsf{Ternärbaum}$ erwarten würde.

Da die Bedingungen 1.), 2.) und 3.) von Satz 6.31 dieselben sind wie diejenigen von Satz 6.23, können diese — wie in Abschnitt 6.2.2 gezeigt — durch einen PSPACE-Algorithmus entschieden werden. Bis auf die Definition von E_A stimmen Bedingung 4.) von Satz 6.31 und 3.) von Satz 5.25 überein. Da die Menge E_A auch für $\mathcal{FLN}^r$ allein durch das Erreichen von Ausschlußzuständen charakterisiert wurde, kann Algorithmus 5.26 ebenfalls für die Entscheidung von Bedingung 4.) verwendet werden. Die Bedingung $T_2 \notin L = \{F \subseteq Q;$ F Ausschlußzustand$\}$ ist dann bzgl. Definition 6.13 zu interpretieren. Da diese Bedingung durch einen PSPACE-Algorithmus entscheidbar ist, ist Algorithmus 5.26 damit ein NPSPACE-Algorithmus.

Zusammen mit der Reduktion von $\mathcal{ALN}$ auf $\mathcal{FLN}^r$ (linearer Zeitaufwand, vgl. Korollar 6.3 und Satz 6.5) erhalten wir

Korollar 6.32.
Das Subsumtionsproblem bzgl. der deskriptiven Semantik in $\mathcal{ALN}$ ist in PSPACE enthalten. □

Wie für die Sprachen $\mathcal{FL}_0$ und $\mathcal{AL}_0$ ist auch für $\mathcal{ALN}$ die PSPACE-Härte der Subsumtion bzgl. der deskriptiven Semantik ein offenes Problem.

6.4 Charakterisierung der lfp-Semantik in $\mathcal{ALN}$

Anders als für $\mathcal{FL}_0$ oder $\mathcal{AL}_0$ können in $\mathcal{FLN}$ unendliche Ketten *gefordert* werden. Dies wird für die Charakterisierung der Inkonsistenz sowie der Subsumtion von Bedeutung sein. Bei der Charakterisierung der lfp-Semantik verhalten sich Zahlenrestriktionen jedoch wie primitive Konzepte, so daß sich die Charakterisierung der lfp-Semantik leicht von $\mathcal{FL}_0$ ($\mathcal{AL}_0$) auf $\mathcal{FLN}$ bzw. $\mathcal{ALN}$ verallgemeinern läßt.

Satz 6.33 (Charakterisierung der lfp-Semantik bzgl. $\mathcal{FLN}$).
Es sei T eine $\mathcal{FLN}$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat, I ein lfp-Modell von T sowie A ein Konzept in T. Dann gilt für alle Individuen $d \in dom(I)$: $d \in A^I$ gdw.

(P1) für alle primitiven Konzepte P in T, alle Wörter $W \in L(A, P)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in P^I$; und

(P2) für alle Maximum-Restriktionen $\exists^{\geq n} R$ in T, alle Wörter $W \in L(A, \exists^{\geq n} R)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\exists^{\geq n} R)^I$; und

(P3) für alle Minimum-Restriktionen $\exists^{\leq n} R$ in T, alle Wörter $W \in L(A, \exists^{\leq n} R)$ und alle Individuen $e \in dom(I)$ mit $(d, e) \in W^I$ gilt $e \in (\exists^{\leq n} R)^I$; und

(P4) für alle unendlichen Pfade der Form $A, W_1, C_1, W_2, C_2, \ldots$ und alle Individuen $d_1, d_2, d_3, \ldots \in dom(I)$ existiert ein $n \geq 1$ mit $(d_{n-1}, d_n) \notin W_n^I$. („Unendliche Ketten sind verboten.“)

Beweis: analog zum Beweis von Proposition 22 in [Baa96]. Die Zahlenrestriktionen sind im Beweis wie primitive Konzepte zu behandeln. □

Bemerkung 6.34.
Dieser Satz kann genau wie der Satz zur Charakterisierung der gfp-Semantik (Bemerkung 6.9) auf die Sprache $\mathcal{ALN}$ verallgemeinert werden, indem die primitive Negation $\neg P$ analog zu (P1), (P2) bzw. (P3) behandelt wird. ◇

In Beispiel 3.2 gilt $\mathsf{eltern}^n \in L(\mathsf{Mensch}, \exists^{\geq 2}\mathsf{eltern})$ für alle $n \geq 0$, d.h. das ω-Wort eltern eltern eltern... wird vom Konzept Mensch gefordert. Für ein lfp-Modell I von T existieren somit nach Lemma 6.7 für ein Individuum $d_0 \in \mathsf{Mensch}^I$ Individuen $d_1, d_2, d_3, \ldots$ mit $d_0 \mathsf{eltern}^I d_1 \mathsf{eltern}^I d_2 \ldots$ Außerdem gilt $\mathsf{eltern}^\omega \in U(\mathsf{Mensch})$. Damit wird für das Individuum d_0 eine unendliche Kette gefordert, die nach (P4) verboten ist. Somit ist das Konzept Mensch bzgl. der lfp-Semantik inkonsistent (vgl. auch Seite 89). Die lfp-Semantik scheint also für dieses Beispiel ungeeignet zu sein. Mit Satz 6.33 läßt sich auch begründen, daß in Beispiel 4.3 die Extension von Binärbaum bzgl. der lfp-Semantik nur die Individuen a, b, c und d enthält; von den anderen Individuen gehen nämlich unendliche Ketten aus, die nach Satz 6.33 nicht erlaubt sind.

6.4.1 Die lfp-Semantik in $\mathcal{ALN}$ — Inkonsistenz

Unendliche Ketten können wie angedeutet der Grund für inkonsistente Konzepte sein. Anders als für $\mathcal{AL}_0$ können solche unendlichen Ketten in $\mathcal{FLN}$ durch Maximum-Restriktionen gefordert werden.

Wir zeigen den Satz zur Charakterisierung der Inkonsistenz mit Hilfe von

Definition 6.35 (kanonisches lfp-Modell bzgl. $\mathcal{FLN}$).
Es sei T eine $\mathcal{FLN}$-Terminologie, $\mathcal{A}_T = (\Sigma, Q, E)$ der zugehörige Semi-Automat sowie A ein Konzept in T. Die *kanonische primitive Interpretation* $J = J(A, d_0)$ zu A und Individuum d_0 sei wie für die gfp-Semantik (Definition 6.10) definiert. Das *kanonische lfp-Modell* $I = I(A, d_0)$ sei das durch J definierte lfp-Modell von T. $\diamond$

Um für das kanonische lfp-Modell I die Aussage $d_0 \in A^I$ beweisen zu können, ist Bedingung (6.3) zu erweitern:

Es exisitiere kein Wort $W \in \Sigma^*$, und es existieren keine widersprüchlichen Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$, so daß W von A gefordert wird und $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$ gilt; außerdem existiere kein von A gefordertes Wort $W \in \Sigma^* \cup \Sigma^\omega$ mit $W \in U(A)$. (6.7)

Da J wie in Definition 6.10 definiert ist, gelten auch die in Lemma 6.11, 1.) – 4.) formulierten Eigenschaften von J. Wir zeigen zusätzlich

Lemma 6.36.
Mit den Bezeichnungen aus Definition 6.35 und Bedingung (6.7) gelten (P1), (P2), (P3) und (P4) von Satz 6.33 bzgl. A, d_0 und J. Für das kanonische lfp-Modell I folgt also insbesondere $d_0 \in A^I$.

Beweis: Nach Lemma 6.11, 5.) gelten die Bedingungen (P1), (P2) und (P3) aus Satz 6.8 (, also auch die entsprechenden Bedingungen von Satz 6.33), da mit den Bedingungen aus (6.7) diejenigen aus (6.3) erfüllt sind.

Um $d_0 \in A^I$ zu zeigen, genügt nach Satz 6.33 somit der Nachweis von (P4) bzgl. A, d_0 und J. Gilt (P4) nicht, so existiert ein unendlicher Pfad der Form $A, W_1, C_1, W_2, C_2, \ldots$ in $\mathcal{A}_T$, und so existieren Individuen $d_1, d_2, d_3, \ldots$ mit $d_{i-1} W_i^J d_i$ für alle $i \geq 1$. Für $W = W_1 W_2 W_3 \cdots$ ist somit $W \in U(A)$, und wegen $d_0 (W_1 W_2 \cdots W_i)^J d_i$, $i \geq 0$, gilt nach Lemma 6.11, 1.), daß jedes endliche Präfix von W von A gefordert wird. Damit wird W von A gefordert. Zusammen mit $W \in U(A)$ liefert dies einen Widerspruch zur Bedingung (6.7). Damit gilt (P4), und also $d_0 \in A^I$. $\square$

Für die algorithmische Behandlung der Inkonsistenz und der Subsumtion wird sich wiederum der Begriff des Ausschlußzustandes als nützlich erweisen.

Definition 6.37 (Ausschlußzustand bzgl. der lfp-Semantik in $\mathcal{FLN}$).
Es bezeichne T eine $\mathcal{FLN}$-Terminologie und $\mathcal{A}_T = (\Sigma, Q, E)$ den zugehörigen Semi-Automaten ohne Worttransitionen. Eine Zustandsmenge $F_0 \subseteq Q$ heißt *Ausschlußzustand* bzgl. $\mathcal{A}_T$ (und bzgl. der lfp-Semantik in $\mathcal{FLN}$), falls gilt:

1.) es existiert ein ω-Wort $R_1 R_2 R_3 \cdots \in \Sigma^\omega$, und es existieren Zustandsmengen $F_1, F_2, F_3, \ldots \subseteq Q$ sowie Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, für alle $i \geq 1$ mit $F_i = next_\varepsilon(F_{i-1}, R_i)$, $i \geq 1$, und $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$, $i \geq 0$; oder

2.) es existiert eine Zahl $n \geq 0$ sowie ein Wort $R_1 \cdots R_n \in \Sigma^*$, und es existieren Zustandsmengen $F_1, \ldots, F_n \subseteq Q$ und Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, für alle $1 \leq i \leq n$ mit $F_i = next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq n$, und $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$ für alle $0 \leq i < n$. Zu F_n existieren außerdem widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$, mit $\exists^{\geq l} R$, $\exists^{\leq r} R \in F_n$, oder es existiert ein definiertes Konzept C, das auf einem ε-Zyklus liegt, mit $C \in F_n$.

$\diamond$

Für Terminologien aus $\mathcal{ALN}$ betrachtet man in Definition 6.37, 2.) neben den widersprüchlichen Zahlenrestriktionen bzw. neben C auch P und $\neg P$ für ein primitives Konzept P.

Die Entscheidungsalgorithmen für Inkonsistenz und Subsumtion benötigen einen (PSPACE-)Entscheidungsalgorithmus für Ausschlußzustände. Wir geben dazu einen NPSPACE-Algorithmus an.

Algorithmus 6.38.

Eingabe: Semi-Automat $\mathcal{A}_T = (\Sigma, Q, E)$ ohne Worttransitionen zu einer Terminologie T; $F_0 \subseteq Q$.

Ausgabe: Es existiert eine Berechnung mit Ausgabe „ja" gdw. F_0 ein Ausschlußzustand ist.

Es bezeichne M die Menge der definierten Konzepte, die auf einem ε-Zyklus liegen.

    $F := F_0$;
    $z := 0$;
    while $z < 2^{|Q|}$ do
(1)   if $F \cap M \neq \emptyset$ or „es existieren widersprüchliche Zahlenrestriktionen
            $\exists^{\geq l} R, \exists^{\leq r} R \in F$, $l > r$" then Ausgabe „ja";
        if „es existieren $m \geq 1$ und Maximum-Restriktion $\exists^{\geq m} R \in F$"
            then „Wähle (nicht-det.) ein $\exists^{\geq m} R \in F$ mit $m \geq 1$"
            else Ausgabe „nein";
        $F := next_\varepsilon(F, R)$;
        $z := z + 1$
    end;
(2) Ausgabe „ja". $\triangle$

Korrektheit: i.) Der Algorithmus gebe in (1) „ja" aus. Es existiert demnach ein endliches Wort $W = R_1 \cdots R_m \in \Sigma^*$ mit $m < 2^{|Q|}$, so daß für die Zustandsmengen $F_i = next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq m$, Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, $1 \leq i \leq m$, existieren mit $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$ für alle $0 \leq i < m$. Außerdem existieren widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$, $\exists^{\leq r} R$, $l > r$, oder ein definiertes Konzept C, das auf einem ε-Zyklus liegt, mit $\exists^{\geq l} R, \exists^{\leq r} R \in F_m$ bzw. $C \in F_m$. Dies zeigt, daß F_0 ein Ausschlußzustand ist.

ii.) Es werde in (2) „ja" ausgegeben. Demnach existiert ein Wort $W = R_1 \cdots R_m \in \Sigma^*$ mit $m = 2^{|Q|}$, so daß für die Zustandsmengen $F_i = next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq m$, Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, $1 \leq i \leq m$, existieren mit $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$ für alle $0 \leq i < m$. Wegen $m = 2^{|Q|}$ existieren Zahlen p und q mit $0 \leq p < q \leq 2^{|Q|}$ und $F_p = F_q$. Damit werden für $\alpha = R_1 \cdots R_p (R_{p+1} \cdots R_q)^\omega$ die Zustandsmengen $F_0, \ldots, F_p, F_{p+1}, \ldots, F_q$, $F_{p+1}, \ldots, F_q, F_{p+1}, \ldots$ durchlaufen. Dies zeigt, daß F_0 ein Ausschlußzustand ist.

Vollständigkeit: Es sei F_0 ein Ausschlußzustand. Damit gilt einer der in iii.), iv.) und v.) betrachteten Fälle:

iii.) Es existiert eine Zahl $m \geq 0$ sowie ein endliches Wort $W = R_1 \cdots R_m \in \Sigma^*$, und es existieren Zustandsmengen $F_1, \ldots, F_m \subseteq Q$ und Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, für alle $1 \leq i \leq m$ mit $F_i = next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq m$, und $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$ für alle $0 \leq i < m$. Zu F_m existiere ein definiertes Konzept C, das auf einem ε-Zyklus liegt, mit $C \in F_m$. Ist $m \geq 2^{|Q|}$, dann existieren Zahlen p und q mit $0 \leq p < q \leq 2^{|Q|}$ und $F_p = F_q$. Für das Wort $W' = R_1 \cdots R_p R_{q+1} \cdots R_m$ werden die Zustandsmengen $F_0, \ldots, F_p, F_{q+1}, \ldots, F_m$ durchlaufen. Es existiert somit neben W, $|W| \geq 2^{|Q|}$, auch ein kürzeres Wort W' mit entsprechenden Eigenschaften, so daß wir o.E. $|W| < 2^{|Q|}$ annehmen können. Wählt der Algorithmus im i-ten Schleifendurchlauf, $1 \leq i \leq m$, jeweils die Maximum-Restriktion $\exists^{\geq m_i} R_i$, so ist nach dem m-ten Schleifendurchlauf $z = m < 2^{|Q|}$ und $F = F_m$, also $C \in F \cap M$. Damit wird im $(m+1)$-ten Schleifendurchlauf in (1) „ja“ ausgegeben. Oder:

iv.) Anders als in iii.) enthalte nun die Menge F_m statt des definierten Konzeptes C widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$. Damit zeigt man, ähnlich wie in iii.), die Existenz einer Berechnung mit Ausgabe „ja“. Oder:

v.) Es existiert ein ω-Wort $R_1 R_2 R_3 \cdots \in \Sigma^\omega$, und es existieren Zustandsmengen $F_1, F_2, F_3, \ldots \subseteq Q$ sowie Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, für alle $i \geq 1$ mit $F_i = next_\varepsilon(F_{i-1}, R_i)$, $i \geq 1$, und $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$, $i \geq 0$. Der Algorithmus wähle im i-ten Schleifendurchlauf, $1 \leq i \leq 2^{|Q|}$, die Maximum-Restriktion $\exists^{\geq m_i} R_i$. Ist dann $z = 2^{|Q|}$, so wird in (2) „ja“ ausgegeben.

Komplexität: Man sieht leicht, daß Algorithmus 6.38 stets terminiert und nur polynomialen Platz benötigt.

Satz 6.39 (Charakterisierung: Inkonsistenz bzgl. lfp-Semantik in $\mathcal{FLN}$). Es sei T eine $\mathcal{FLN}$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat ohne Worttransitionen[11] und A ein Konzept in T. Folgende Aussagen sind äquivalent:

1.) A ist T-inkonsistent bzgl. der lfp-Semantik in $\mathcal{FLN}$.

2.) i.) Es existiert ein (endliches oder unendliches) von A gefordertes Wort W mit $W \in U(A)$; oder

ii.) es existieren widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$, und es existiert ein von A gefordertes Wort $W \in \Sigma^*$ mit $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$.

3.) ε-$closure(\{A\})$ ist ein Ausschlußzustand.

Beweis: Äquivalenz der Aussagen 1.) und 2.):

„1. $\Rightarrow$ 2.“: Gilt die rechte Seite der Äquivalenzaussage nicht, so existiert nach Lemma 6.36 das kanonische lfp-Modell $I = I(A, d_0)$ zu A und einem Individuum d_0 mit $d_0 \in A^I$. Dies zeigt die Konsistenz von A.

„1. $\Leftarrow$ 2.“: Annahme: Es existiert ein lfp-Modell I von T sowie ein Individuum $d_0 \in dom(I)$ mit $d_0 \in A^I$. Außerdem gelte einer der beiden folgenden Fälle:
i.) Es existiert ein von A gefordertes Wort $W \in U(A)$. Ist W endlich, so existiert nach Lemma 6.7 ein Individuum e mit $d_0 W^I e$. Wegen $W \in U(A)$ existiert ein definiertes

[11] Die Betrachtung eines Semi-Automaten ohne Worttransitionen ist nur für die Charakterisierung durch Ausschlußzustände nötig (vgl. Lemma 2.5 und Beispiel 2.6). Die Äquivalenz von 1.) und 2.) gilt auch für beliebige Semi-Automaten $\mathcal{A}_T$.

Konzept C, so daß $A, W, C, \varepsilon, C, \varepsilon, C, \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$ ist; außerdem gilt $d_0 W^I e \varepsilon^I e \varepsilon^I e \cdots$. Satz 6.33, (P4) liefert damit $d_0 \notin A^I$, im Widerspruch zur Annahme. Ist W das unendliche Wort $R_1 R_2 R_3 \cdots$, so existieren nach Lemma 6.7 Individuen $d_1, d_2, d_3, \ldots$ mit $d_0 R_1^I d_2 R_2^I d_2 \cdots$. Zusammen mit $W \in U(A)$ folgt auch hier aus Satz 6.33, (P4): $d_0 \notin A^I$, ebenfalls im Widerspruch zur Annahme.
ii.) Es existiert ein von A gefordertes Wort $W \in \Sigma^*$, und es existieren widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$, $\exists^{\leq r} R$, $l > r$ mit $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$. Aus Lemma 6.7 folgt damit die Existenz eines Individuums e mit $d_0 W^I e$. Wegen $d_0 \in A^I$ folgt aus Satz 6.33, (P2) bzw. (P3) weiter: $e \in (\exists^{\geq l} R)^I$ und $e \in (\exists^{\geq l} R)^I$, was zusammen mit $l > r$ einen Widerspruch darstellt.

Äquivalenz der Aussagen 2.) und 3.):

„2. $\Rightarrow$ 3.“: Es sei $W = R_1 \cdots R_m$ ein endliches Wort in $U(A)$, welches von A gefordert werde. Somit ist W Label eines unendlichen Pfades $A, R_1, C_1, R_2, C_2, \ldots, R_m, C_m, \varepsilon, C, \varepsilon, C, \ldots$ ($\mathcal{A}_T$ Semi-Automat ohne Worttransitionen) für ein definiertes Konzept C, das auf einem ε-Zyklus liegt. Weiter existieren für $F_i := next_\varepsilon(A, R_1 \cdots R_i)$, $0 \leq i \leq m$, Maximum-Restriktionen $\exists^{\geq m_{i+1}} R_{i+1}$ mit $m_{i+1} \geq 1$, $0 \leq i < m$, so daß gilt: $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$ für alle $0 \leq i < m$. Schließlich ist nach Lemma 2.5 das Konzept C Element von F_m. Dies zeigt, daß $F_0 = \varepsilon\text{-}closure(\{A\})$ ein Ausschlußzustand ist.

Es sei nun $W = R_1 R_2 R_3 \cdots$ ein von A gefordertes unendliches Wort in $U(A)$. Für $F_i := next_\varepsilon(A, R_1 \cdots R_i)$, $i \geq 0$, gilt, da W von A gefordert wird, daß die Mengen F_i für alle $i \geq 0$ Zahlenrestriktionen $\exists^{\geq m_{i+1}} R_{i+1}$ mit $m_{i+1} \geq 1$ enthalten. Damit ist $F_0 = \varepsilon\text{-}closure(\{A\})$ ein Ausschlußzustand.

Schließlich sei $W = R_1 \cdots R_m \in \Sigma^*$ ein von A gefordertes Wort mit $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$ für widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$, $\exists^{\leq r} R$, $l > r$. Es seien $F_i := next_\varepsilon(A, R_1 \cdots R_i)$ für alle $0 \leq i \leq m$. Da W von A gefordert wird, existieren Maximum-Restriktionen $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$ mit $m_{i+1} \geq 1$ für alle $0 \leq i < m$; außerdem enthält F_m die Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$. Damit ist auch in diesem Fall $F_0 = \varepsilon\text{-}closure(\{A\})$ ein Ausschlußzustand.

„2. $\Leftarrow$ 3.“: Ist $F_0 := \varepsilon\text{-}closure(\{A\})$ ein Ausschlußzustand, so trifft einer der drei Fälle zu:
i.) Es existiert ein ω-Wort $W = R_1 R_2 R_3 \cdots \in \Sigma^\omega$, und es existieren Zustandsmengen $F_1, F_2, F_3, \ldots \subseteq Q$ sowie Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, $i \geq 1$, mit $F_i = next_\varepsilon(F_{i-1}, R_i)$, $i \geq 1$, und $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$, $i \geq 0$. Wegen $F_i \neq \emptyset$ für alle $i \geq 0$ ist nach Lemma 2.10 das Wort $W \in U(A)$. Lemma 2.5 impliziert $R_1 \cdots R_i \in L(A, \exists^{\geq m_{i+1}} R_{i+1})$, $i \geq 0$. Also wird W von A gefordert.
ii.) Es existiert ein endliches Wort $W = R_1 \cdots R_m \in \Sigma^*$, und es existieren Zustandsmengen $F_1, \ldots, F_m \subseteq Q$ sowie Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, $1 \leq i \leq m$, mit $F_i = next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq m$, und $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$ für alle $0 \leq i < m$. Existiert ein definiertes Konzept C, das auf einem ε-Zyklus liegt, mit $C \in F_m$, so existiert ein unendlicher Pfad der Form $A, R_1, C_1, R_2, C_2, \ldots, R_m, C_m, \varepsilon, C, \varepsilon, C, \ldots$ in $\mathcal{A}_T$, also $W \in U(A)$. Wegen $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$, $m_{i+1} \geq 1$ für alle $0 \leq i < m$ wird wie in i.) W von A gefordert.
iii.) Im Unterschied zu ii.) enthalte nun F_m statt des Konzeptes C widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$, $\exists^{\leq r} R$, $l > r$. Wie in ii.) wird W von A gefordert. Weiterhin gilt gemäß Lemma 2.5: $W \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$. □

Satz 6.39, i.) beschreibt den Fall, daß für die Elemente in der Extension von A unendliche Ketten gefordert werden. Das Konzept Mensch aus Beispiel 3.2 ist also aufgrund von Satz

6.39, i.) inkonsistent, da $\mathsf{eltern}^\omega \in U(\mathsf{Mensch})$ gilt und eltern^ω von Mensch gefordert wird.

Betrachtet man in Satz 6.39, ii.) neben widersprüchlichen Zahlenrestriktionen auch P und $\neg P$ für ein primitives Konzept P, so gilt der Satz (bei entsprechender Anpassung der Definition des Ausschlußzustandes) auch für $\mathcal{ALN}$-Terminologien.

Zusammen mit Algorithmus 6.38 zeigt Satz 6.39, daß das Inkonsistenzproblem für $\mathcal{FLN}$ ($\mathcal{ALN}$) bzgl. der lfp-Semantik ein PSPACE-Problem ist. Außerdem gilt nach Satz 6.17 die NP-Härte der Inkonsistenz und für schwach-azyklische sowie azyklische $\mathcal{ALN}$- (bzw. $\mathcal{FLN}$-)Terminologien die NP-Vollständigkeit der Inkonsistenz (vgl. Korollar 7.16).

6.4.2 Die lfp-Semantik in $\mathcal{ALN}$ — Subsumtion

Satz 6.33 impliziert, daß für $A \sqsubseteq_T B$ die folgende Bedingung hinreichend ist: $L(B,P) \subseteq L(A,P)$ für alle primitiven Konzepte P, $L(B, \exists^{\geq n} R) \subseteq L(A, \exists^{\geq n} R)$ für alle Maximum-Restriktionen $\exists^{\geq n} R$, $L(B, \exists^{\leq n} R) \subseteq L(A, \exists^{\leq n} R)$ für alle Minimum-Restriktionen $\exists^{\leq n} R$ und $U(B) \subseteq U(A)$. Die Beispiele 5.33 und 6.18 machen jedoch deutlich, daß diese Bedingung nicht notwendig für die Subsumtion ist. Es ist vielmehr nötig, A-ausschließende Wörter zu berücksichtigen, die wegen Satz 6.33, (P4) auch unendlich sein können. Die Charakterisierung der Inkonsistenz (Satz 6.39) gibt einen Hinweis darauf, wie der Begriff „Ausschluß“ bzgl. der lfp-Semantik in $\mathcal{FLN}^r$ zu definieren ist.

Definition 6.40 (Ausschluß bzgl. der lfp-Semantik in $\mathcal{FLN}^r$).
Es sei T eine $\mathcal{FLN}^r$-Terminologie[12], $\mathcal{A}_T = (\Sigma, Q, E)$ der zugehörige Semi-Automat sowie A ein Konzept in T. Das endliche Wort $W \in \Sigma^*$ *schließt A aus*, falls eine der folgenden Bedingungen erfüllt ist:

1.) Es existiert ein (endliches oder unendliches) Wort $\alpha \in U(A)$ sowie ein Wort $V \in \Sigma^*$, das Präfix von W und α ist, so daß α ab V von A gefordert wird.

2.) Es existiert ein Präfix $V \in \Sigma^*$ von W sowie ein Wort $V' \in \Sigma^*$, und es existieren widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$ und $\exists^{\leq r} R$, $l > r$, so daß $VV' \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$ gilt und VV' ab V von A gefordert wird.

Das ω-Wort $W \in \Sigma^\omega$ *schließt A aus*, falls ein endliches Präfix von W existiert, welches A ausschließt, oder $W \in U(A)$ gilt.

Weiter seien die Mengen $E_A := \{W \in \Sigma^*; W \text{ schließt } A \text{ aus}\}$, $E_{A,\omega} := \{W \in \Sigma^* \cup \Sigma^\omega;$ W schließt A aus$\}$ sowie $E^f_{A,\omega} := \{W \in \Sigma^* \cup \Sigma^\omega;$ ein endliches Präfix von W schließt A aus$\}$ definiert. ◇

Aus dem gleichen Grund wie bei der gfp-Semantik wird „Ausschluß“ nur für $\mathcal{FLN}^r$-Terminologien definiert. Für $\mathcal{FLN}$-Terminologien müßten wir die Definition ergänzen (vgl. Seite 71).

Für ein inkonsistentes Konzept A gilt: $E_A = \Sigma^*$ und $E^f_{A,\omega} = E_{A,\omega} = \Sigma^* \cup \Sigma^\omega$, da nach Satz 6.39 bereits das leere Wort ε das Konzept A ausschließt. Gilt umgekehrt $E_A = \Sigma^*$ ($E^f_{A,\omega} = \Sigma^* \cup \Sigma^\omega$ oder $E_{A,\omega} = \Sigma^* \cup \Sigma^\omega$), so wird A von ε ausgeschlossen. Satz 6.39 und Definition 6.40 liefern damit die Inkonsistenz von A.

Die Bedeutung von Definition 6.40 zeigt

[12] Durch Berücksichtigung von P und $\neg P$ neben widersprüchlichen Zahlenrestriktionen kann die Definition auf $\mathcal{ALN}^r$-Terminologien verallgemeinert werden.

Lemma 6.41.
Mit den Bezeichnungen und Voraussetzungen aus Definition 6.40 gilt für ein lfp-Modell I zur Terminologie T:

1.) Ist $W \in \Sigma^* \cup \Sigma^\omega$ sowie $V \in \Sigma^*$ ein Präfix von W mit den Eigenschaften 1.) oder 2.) aus Definition 6.40, und sind $d, e \in dom(I)$ Individuen mit $dV^I e$, so gilt $d \notin A^I$.[13]

2.) Für ein ω-Wort $W \in U(A)$, $W = R_1 R_2 R_3 \cdots$, sowie Individuen $d_0, d_1, d_2, \ldots$ mit $d_0 R_1^I d_1 R_2^I d_2 \cdots$ ist $d_0 \notin A^I$.

Beweis: zu 1.) Es sei zunächst α (vgl. Definition 6.40, 1.)) ein endliches Wort, d.h. $\alpha = VV'$ für ein $V' \in \Sigma^*$. Da α ab V von A gefordert wird, liefert Lemma 6.7 die Existenz eines Individuums f mit $dV^I eV'^I f$. Wegen $\alpha \in U(A)$ existiert ein definiertes Konzept C, das auf einem ε-Zyklus liegt, so daß $A, \alpha, C, \varepsilon, C, \varepsilon, \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$ ist. Zusammen mit $dV^I eV'^I f \varepsilon^I f \varepsilon^I f \cdots$ und (P4) von Satz 6.33 gilt damit $d \notin A^I$.

Es sei nun α das ω-Wort $VR_1R_2R_3 \cdots \in U(A)$. Da α ab V von A gefordert wird, folgt aus Lemma 6.7 die Existenz von Individuen $d_1, d_2, d_3, \ldots$ mit $dV^I eR_1^I d_1 R_2^I d_2 R_3^I d_3 \cdots$. Zusammen mit $\alpha \in U(A)$ liefert Satz 6.33, (P4) ebenfalls $d \notin A^I$.

Schließt W das Konzept A aufgrund von 2.) in Definition 6.40 aus, so existiert nach Lemma 6.7, da VV' ab V von A gefordert wird, ein Individuum f mit $dV^I eV'^I f$. Aus $VV' \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$ und (P2), (P3) von Satz 6.33 folgt damit für $d \in A^I$: $f \in (\exists^{\geq l} R)^I$ und $f \in (\exists^{\leq r} R)^I$, was zusammen mit $l > r$ einen Widerspruch darstellt. Es ist also $d \notin A^I$.

zu 2.) Mit Satz 6.33, (P4) folgt sofort $d_0 \notin A^I$. □

Für den Beweis der Charakterisierung der Subsumtion bzgl. der lfp-Semantik benötigen wir

Definition 6.42 (erweitertes kanonisches lfp-Modell bzgl. $\mathcal{FLN}^r$).
Es sei T eine $\mathcal{FLN}^r$-Terminologie, $\mathcal{A}_T$ der zugehörige Semi-Automat sowie A ein Konzept in T, W ein Wort in $\Sigma^* \cup \Sigma^\omega$, $r \in \mathbb{N}$ und $R \in \Sigma$. Für $W \in \Sigma^*$ definieren wir die *erweiterte kanonische primitive Interpretation* $J' = J(A, d_0, W)$ $(r = 0)$ bzw. $J' = J(A, d_0, W, R, r)$ $(r > 0)$ wie für die gfp-Semantik in Definition 6.21. Für $W \in \Sigma^\omega$ sei $J' = J(A, d_0, W)$ wie für die deskriptive Semantik in Definition 6.30 definiert. Die *erweiterten kanonischen lfp-Modelle* $I' = I(A, d_0, W)$ bzw. $I' = I(A, d_0, W, R, r)$ sind die zu $J' = J(A, d_0, W)$ bzw. $J' = J(A, d_0, W, R, r)$ gehörenden lfp-Modelle von T. ◇

Um $d_0 \in A^{I'}$, d.h. die Gültigkeit von (P1) – (P4) von Satz 6.33 bzgl. A, d_0, T und J' folgern zu können, benötigen wir die Bedingung:

> Das Konzept A ist konsistent und wird von W nicht ausgeschlossen (Definition 6.40); für $r > 0$ — in diesem Fall sei W ein endliches Wort — werde A zudem nicht von WR ausgeschlossen. Für alle $l < r$ gelte $W \notin L(A, \exists^{\leq l} R)$. (6.8)

Es gelten die Aussagen von Lemma 6.22 bzw. die Aussagen 8.) und 9.) von Seite 83 zur erweiterten primitiven kanonischen Interpretation nach Definition auch für die primitive Interpretation J' aus Defintion 6.42. Schließt ein endliches Wort $W \in \Sigma^*$ das Konzept

[13] Wird statt $dV^I e$ die Beziehung $dW^I f$ für ein Individuum f vorausgesetzt, so folgt, da V Präfix von W ist, die Existenz eines Individuums e mit $dV^I e$. Nach Aussage 1.) gilt somit ebenfalls $d \notin A^I$.

A bzgl. Definition 6.19, so schließt es A auch bzgl. Definition 6.40 aus. Somit folgen aus Bedingung (6.8) die Bedingungen (6.4) und (6.6). Damit genügt aber nach Lemma 6.22, 9.) bzw. Aussage 9.) von Seite 83 für den Beweis der Aussage $d_0 \in A^{I'}$ der Nachweis der Gültigkeit von Satz 6.33, (P4) bzgl. A, d_0, T und J'. Diesen führen wir in

Lemma 6.43.
Mit den Bezeichnungen aus Definition 6.42 und Bedingung (6.8) gilt: $d_0 \in A^{I'}$.

Beweis: Nach obigen Ausführungen ist nur noch (P4) für A, d_0, T und J' zu zeigen. Wir nehmen dazu an, daß (P4) nicht gilt. Es existiert also ein unendlicher Pfad der Form $A, V_1, C_1, V_2, C_2, \ldots$ in $\mathcal{A}_T$, und es existieren Individuen $e_1, e_2, e_3, \ldots \in dom(J')$ mit $(e_{i-1}, e_i) \in V_i^{J'}$ für alle $i \geq 1$, $e_0 := d_0$. Es sei $I = I(A, d_0)$ das kanonische Modell zu A und d_0 sowie $J = J(A, d_0)$ die zugehörige primitive Interpretation. Wären $e_1, e_2, e_3, \ldots \in dom(I)$, so wäre (P4) von Satz 6.33 bzgl. A, d_0 und J verletzt, da nach Lemma 6.22, 5.) gilt: $(e_{i-1}, e_i) \in V_i^J$. Nach Lemma 6.36 gilt aber $d_0 \in A^I$, womit nach Satz 6.33 die Gültigkeit von Satz 6.33, (P4) bzgl. A, d_0 und J folgt; dies ist ein Widerspruch.

Es existiert demnach ein $i \geq 1$ mit $e_i \in dom(J') \setminus dom(J)$. Nach Lemma 6.22, 4.) sind alle Elemente aus $dom(J') \setminus dom(J)$ (direkte oder indirekte) Nachfolger von d_1 (vgl. Definition 6.21, $d_0 U^J d_1$). Zusammen mit $e_i \in dom(J') \setminus dom(J)$ gilt deshalb für $V = V_1 V_2 V_3 \cdots$ die Aussage: $V = W$ (für $r = 0$), $V = WR$ (für $r > 0$) oder $V = UXY$, $X \in \Sigma^+$, $Y \in \Sigma^* \cup \Sigma^\omega$, UX maximales Präfix von W (für $r = 0$) bzw. von WR (für $r > 0$)(vgl. Beweis zu Lemma 6.22, 9.)). Die Fälle $V = W$ bzw. $V = WR$ können wegen $V \in U(A)$ direkt ausgeschlossen werden, da sonst A von W bzw. WR ausgeschlossen würde, im Widerspruch zur Voraussetzung. Es ist also $V \neq W$ bzw. $V \neq WR$ und $V = UXY$. Nach Definition von J' wird V ab UX von A gefordert. Wegen $V \in U(A)$ und UX Präfix von W bzw. WR wird damit A von W bzw. WR ausgeschlossen, im Widerspruch zur Voraussetzung. Damit muß (P4) gelten, und es folgt $d_0 \in A^{I'}$. □

Nun können wir folgenden Satz zeigen:

Satz 6.44 (Charakterisierung: Subsumtion bzgl. lfp-Semantik in $\mathcal{FLN}^r$).
Es bezeichne T eine $\mathcal{FLN}^r$-Terminologie und $\mathcal{A}_T$ den zugehörigen Semi-Automaten, A und B bezeichnen Konzepte in T. Es gilt $A \sqsubseteq_{lfp,T} B$ gdw.

1.) $L(B, P) \subseteq L(A, P) \cup E_A$ für alle primitiven Konzepte P in T; und

2.) $L(B, \exists^{\geq l} R) \subseteq (\bigcup_{r \geq l} L(A, \exists^{\geq r} R) \cup E_A)$ für alle Maximum-Restriktionen der Form $\exists^{\geq l} R$ in T mit $l > 0$; und

3.) $L(B, \exists^{\leq l} R) \cdot R \subseteq ((\bigcup_{r \leq l} L(A, \exists^{\leq r} R)) \cdot R \cup E_A)$ für alle Minimum-Restriktionen der Form $\exists^{\leq l} R$ in T; und

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

Beweis: „$\Rightarrow$“: Wir nehmen an, daß jeweils eine der Bedingungen 1.) – 4.) nicht gilt und zeigen jeweils $A \not\sqsubseteq_{lfp,T} B$.

Die Fälle 1.), 2.) und 3.) behandelt man unter Verwendung von Lemma 6.43 und Satz 6.33 wie im Beweis zur Charakterisierung der Subsumtion bzgl. der gfp-Semantik (Satz 6.23).

Den Fall, daß 4.) nicht gilt, behandelt man wie im Beweis von Satz 5.38. Statt Lemma 5.24 zu verwenden, wird Lemma 6.22 bzw. die Aussagen 8.) und 9.) von Seite 83 sowie Lemma 6.43 benutzt. Außerdem wird Satz 6.33 statt Satz 5.29 verwendet.

„$\Leftarrow$“: Es gelte die rechte Seite der Äquivalenzaussage. Wir nehmen $A \not\sqsubseteq_{lfp,T} B$ an. Damit existiert ein lfp-Modell I zu T und ein Individuum $d_0 \in dom(I)$ mit $d_0 \in A^I \setminus B^I$. Wegen $d_0 \notin B^I$ gelten also (P1), (P2), (P3) oder (P4) von Satz 6.33 bzgl. B, d_0, T und I nicht.

Die Fälle (P1), (P2) und (P3) behandelt man mit Hilfe von Satz 6.33 und Lemma 6.41 wie im Beweis von Satz 6.23 („$\Leftarrow$“).

Den Fall, daß (P4) nicht gilt, behandelt man mit Hilfe von Satz 6.33 und Lemma 6.41 wie im Beweis zu Satz 5.38. □

Zur Verallgemeinerung von Satz 6.44 auf $\mathcal{ALN}^r$-Terminologien vgl. Bemerkung 6.24.

Die Einschränkung in Definition 6.40 auf die Sprache $\mathcal{FLN}^r$ ($\mathcal{ALN}^r$) erlaubt die Charakterisierung der Mengen E_A und $E^f_{A,\omega}$ allein durch Ausschlußzustände.

Lemma 6.45.
Für eine $\mathcal{FLN}^r$-Terminologie T, den Semi-Automaten $\mathcal{A}_T$ ohne Worttransitionen und ein Konzept A in T gilt:

1.) $E_A = \{W \in \Sigma^*;$ mit W erreicht man von A aus einen Ausschlußzustand$\}$;

2.) $E^f_{A,\omega} = \{W \in \Sigma^* \cup \Sigma^\omega;$ mit W erreicht man von A aus einen Ausschlußzustand$\}$.

Dabei ist „einen Auschlußzustand erreichen“ wie in Definition 5.14 definiert; „Ausschlußzustand“ bezieht sich auf Definiton 6.37.

Beweis: zu 1.) „$\subseteq$“: Sei $W \in E_A$, d.h.:
a) Es existiert ein endliches oder unendliches Wort $\alpha \in U(A)$ sowie ein endliches Wort V, das Präfix von α und W ist, so daß α ab V von A gefordert wird. Ist α endlich, so sei $\alpha = VR_1 \cdots R_m \in \Sigma^*$. Es sei weiter $F_i := next_\varepsilon(A, VR_1 \cdots R_i)$ für alle $0 \le i \le m$. Wegen $\alpha \in U(A)$ und α endlich existiert ein definiertes Konzept C, das auf einem ε-Zyklus liegt, so daß $A, \alpha, C, \varepsilon, C, \varepsilon, C, \ldots$ ein unendlicher Pfad in $\mathcal{A}_T$ ist. Damit ist $C \in F_m$ nach Lemma 2.5. Da α ab V von A gefordert wird, existieren Zahlenrestriktionen $\exists^{\ge m_i} R_i$, $m_i \ge 1$, für alle $1 \le i \le m$ mit $\exists^{\ge m_{i+1}} R_{i+1} \in F_i$ für alle $0 \le i < m$. Dies zeigt, daß F_0 ein Ausschlußzustand ist, der mit W von A aus erreichbar ist. Ist α nun das unendliche Wort $VR_1R_2R_3 \cdots$, so gilt für $F_i := next_\varepsilon(A, VR_1 \cdots R_i)$ für alle $i \ge 0$, daß Zahlenrestriktionen $\exists^{\ge m_i} R_i$, $m_i \ge 1$, für alle $i \ge 1$ existieren mit $\exists^{\ge m_{i+1}} R_{i+1} \in F_i$ für alle $i \ge 0$. Daraus folgt wiederum, daß F_0 ein Ausschlußzustand ist, der mit W von A aus erreichbar ist. Oder:
b) Es existiert ein Präfix $V \in \Sigma^*$ von W sowie ein Wort $V' \in \Sigma^*$, und es existieren widersprüchliche Zahlenrestriktionen $\exists^{\ge l} R$ und $\exists^{\le r} R$, $l > r$, so daß $VV' \in L(A, \exists^{\ge l} R) \cap L(A, \exists^{\le r} R)$ und VV' ab V von A gefordert wird. Man zeigt nun ähnlich zu a) (Fall: α endlich), daß F_0 Ausschlußzustand ist.

„$\supseteq$“: Es sei $W \in \Sigma^*$ ein Wort, mit dem man von A aus einen Ausschlußzustand erreicht, d.h. es existiert ein Präfix $V \in \Sigma^*$ von W, so daß $F_0 := next_\varepsilon(A, V)$ ein Ausschlußzustand ist. Für F_0 gilt also:
a) Es existiert ein ω-Wort $V' = R_1R_2R_3 \cdots \in \Sigma^\omega$, und es existieren Zustandsmengen $F_1, F_2, F_3, \ldots \subseteq Q$ sowie Zahlenrestriktionen $\exists^{\ge m_i} R_i$, $m_i \ge 1$, für alle $i \ge 1$ mit $F_i = next_\varepsilon(F_{i-1}, R_i)$, $i \ge 1$, und $\exists^{\ge m_{i+1}} R_{i+1} \in F_i$, $i \ge 0$. Damit wird VV' ab V von A gefordert,

und wegen $F_i \neq \emptyset$ für alle $i \geq 0$ gilt nach Lemma 2.10: $VV' \in U(A)$. Also schließt W das Konzept A aus. Oder:
b) Es existiert ein endliches Wort $V' = R_1 \cdots R_n \in \Sigma^*$, und es existieren Zustandsmengen $F_1, \ldots F_n \subseteq Q$ sowie Zahlenrestriktionen $\exists^{\geq m_i} R_i$, $m_i \geq 1$, für alle $1 \leq i \leq n$ mit $F_i = next_\varepsilon(F_{i-1}, R_i)$, $1 \leq i \leq n$, und $\exists^{\geq m_{i+1}} R_{i+1} \in F_i$,$0 \leq i < n$. Zu F_n existieren außerdem widersprüchliche Zahlenrestriktionen $\exists^{\geq l} R$, $\exists^{\leq r} R$, $l > r$, mit $\exists^{\geq l} R, \exists^{\leq r} R \in F_n$, oder es existiert ein definiertes Konzept C, das auf einem ε-Zyklus liegt, mit $C \in F_n$. Damit wird VV' ab V von A gefordert, und es ist $VV' \in L(A, \exists^{\geq l} R) \cap L(A, \exists^{\leq r} R)$ bzw. $\alpha = VV' \in U(A)$. Also wird A von W ausgeschlossen.

zu 2.) Wir haben an keiner Stelle im Beweis zu 1.) benutzt, daß E_A nur endliche Wörter enthält. Der Beweis zu $E^f_{A,\omega}$ kann deshalb analog geführt werden. □

Diese Charakterisierung der Menge E_A ermöglicht die Entscheidung der Bedingungen 1.), 2.) und 3.) von Satz 6.44 wie für die gfp-Semantik mit Algorithmus 5.16 bzw. den modifizierten Versionen (vgl. Seite 80). Dabei ist die Bedingung $T_2 \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ bzgl. Definition 6.37 zu interpretieren. Algorithmus 6.38 zeigt, daß diese Bedingung mit einem PSPACE-Algorithmus entscheidbar ist. Somit sind die Bedingungen 1.), 2.) und 3.) von Satz 6.44 durch einen PSPACE-Algorithmus entscheidbar. Die Charakterisierung von $E^f_{A,\omega}$ erlaubt es, Bedingung 4.) durch Algorithmus 5.41 zu entscheiden, wobei wiederum für den Ausschlußzustand Definition 6.37 zu wählen ist. Da $T_2 \notin \{F \subseteq Q;\ F$ Ausschlußzustand$\}$ mit einem PSPACE-Algorithmus getestet werden kann, ist Algorithmus 5.41 ein NPSPACE-Algorithmus.

Zusammen mit der Reduktion von $\mathcal{ALN}$ auf $\mathcal{FLN}^r$ (linearer Aufwand, vgl. Korollar 6.3 und Satz 6.5) und der PSPACE-Vollständigkeit des Subsumtionsproblems bzgl. der lfp-Semantik in $\mathcal{FL}_0$ ([Baa96], Korollar 27) erhalten wir

Korollar 6.46.
Das Subsumtionsproblem bzgl. der lfp-Semantik in $\mathcal{ALN}$ ($\mathcal{FLN}$) ist PSPACE-vollständig. □

Kapitel 7

$\mathcal{ALN}$-Schemata als spezielle $\mathcal{ALN}$-Terminologien

Gemäß [BDNS97] erfüllt eine Terminologie zwei Aufgaben: Zum einen werden „primitiv definierte" Konzepte und Rollen eingeführt und zum anderen werden mit Hilfe dieser Konzepte und Rollen definierte Konzepte eingeführt. Angelehnt an (objekt-orientierte) Datenbanken wird eine Terminologie deshalb in Schemadefinition und Viewdefinition (Sichten) eingeteilt.

Die Schemadefinition erlaubt für die primitiv definierten Konzepte und Rollen die Spezifizierung einfacher notwendiger Eigenschaften und Beziehungen[1], z.B. die Existenz von Rollenfüllern und Subsumtionsbeziehungen für Konzepte sowie Bereichsbeschränkungen für Rollen. Hier geht es nur darum, die möglichen Modelle der Terminologie einzuschränken. Aus diesem Grund wird die Bedeutung eines Schemas auch bei Vorhandensein zyklischer Definitionen durch die deskriptive Semantik definiert.

Analog zu Datenbankensichten dient die Viewdefinition der Einführung von Konzepten mit Hilfe der Konzepte und Rollen des Schemas. Hier werden Konzepte — oft mit einer im Vergleich zur Schemasprache ausdrucksstärkeren Sprache — durch Angabe von hinreichenden *und* notwendigen Bedingungen definiert. Modelle des Schemas werden dabei eindeutig durch Fixpunktsemantiken auf die Viewdefinition erweitert.

In diesem Kapitel werden wir uns mit Schemata beschäftigen. Diese bestehen aus einer endlichen Menge von Konzeptinklusionen und Rolleninklusionen. Der Unterschied zu Terminologien ist also der, daß statt Axiome (Konzeptdefinitionen) für Konzepte nur notwendige Bedingungen in Form von Konzeptinklusionen eingeführt werden sowie zusätzlich Bereichsbeschränkungen für Rollen durch Rolleninklusionen angegeben werden können.

Für den Wissensingenieur sind Gültigkeit von Schemata und Subsumtion bzgl. Schemata Standard-Inferenzdienste. In [BDNS97] wurden dazu spezielle (N)PSPACE-Entscheidungsalgorithmen angegeben. Wir werden in diesem Kapitel sehen, wie Gültigkeit und Subsumtion bzgl. Schemata auf Konsistenz und Subsumtion bzgl. Terminologien reduziert werden können. Dadurch erhalten wir zusammen mit den bisherigen Ergebnissen dieser Arbeit ebenfalls die erwähnten PSPACE-Resultate. Dabei läßt die hier betrachtete Schemasprache im Gegensatz zur Sprache in [BDNS97] beliebige Zahlenrestriktionen zu.

[1] Damit unterscheiden sich primitiv definierte Konzepte von primitiven Konzepten, da die Interpretation primitiver Konzepte keinen Einschränkungen unterliegt, die Interpretation primitiv definierter Konzepte jedoch bereits bestimmte Eigenschaften besitzen muß, um die in der Schemadefinition formulierten (notwendigen) Bedingungen zu erfüllen.

Für „schwach-azyklische“ $\mathcal{ALN}$-Schemata und -Terminologien erhalten wir zusätzlich, daß Konsistenz und Gültigkeit co-NP-Probleme sind. Die Reduktion liefert außerdem, unter Verwendung von Resultaten aus [BDNS97], co-NP-Härte für Konsistenz und Gültigkeit bzgl. $\mathcal{ALN}$-Schemata und -Terminologien; insgesamt folgt für „schwach-azyklische“ $\mathcal{ALN}$-Schemata und -Terminologien also die co-NP-Vollständigkeit der Konsistenz und der Gültigkeit.

Im nächsten Abschnitt werden wir eine $\mathcal{ALN}$-basierte Schemasprache einführen. Es zeigt sich, daß diese mit der Schemasprache aus [BDNS97] bis auf die Einbindung beliebiger Zahlenrestriktionen übereinstimmt. Die angesprochene Reduktion wird im Abschnitt 7.2 behandelt.

7.1 $\mathcal{ALN}$-Schemata

$\mathcal{ALN}$-Schemata erlauben Konzeptinklusionen zwischen $\mathcal{ALN}$-Konzepttermen sowie einfache Rolleninklusionen. Wir werden sehen, daß Konzeptinklusionen mit „flachen“ Konzepttermen für die Inferenzprobleme Konsistenz, (lokale) Gültigkeit und Subsumtion ausreichend sind.

Definition 7.1 ($\mathcal{ALN}$-, $\mathcal{SLN}$- und $\mathcal{SL}_{dis}$-Schemata).
Syntax:
Ein *$\mathcal{ALN}$-Schema S* besteht aus einer endlichen Menge von *(S-)Inklusionsaxiomen.* Inklusionsaxiome können *(S-)Konzeptinklusionen* oder *(S-)Rolleninklusionen* sein.

Eine Konzeptinklusion hat die Form $A \sqsubseteq D$ mit einem Konzeptnamen $A \in N_C$ (zu N_C vgl. Definition 3.1) und einem $\mathcal{ALN}$-Konzeptterm D. Existiert zu einem Konzeptnamen A eine S-Konzeptinklusion mit A als linker Seite, so heißt A *definiertes (S-)Konzept.* Für S wird gefordert, daß alle Konzepte aus N_C, die in S vorkommen, definierte Konzepte sind. Die Konzepte aus N_P (zu N_P vgl. Definition 3.1), die in S vorkommen, heißen *primitive (S-)Konzepte.*

Rolleninklusionen haben die Form $R \sqsubseteq A \times B$ mit einem Rollennamen R und (primitiven oder definierten) Konzepten A und B. Wie bei Konzepten spricht man von *primitiven* und *definierten (S-)Rollen.*

Anders als bei Terminologien sind zu einem Konzeptnamen bzw. Rollennamen mehrere Inklusionsaxiome erlaubt.

$\mathcal{SLN}$-Schemata enthalten wie $\mathcal{ALN}$-Schemata Rolleninklusionen. Als Konzeptterme auf der rechten Seite von Konzeptinklusionen sind anders als bei $\mathcal{ALN}$-Schemata nur Zahlenrestriktionen, primitive Negation, (primitive und definierte) Konzepte sowie Werterestriktionen der Form $\forall R.B$, R Rollenname und B (primitives oder definiertes) Konzept, zugelassen.

$\mathcal{SL}_{dis}$-Schemata sind wie $\mathcal{SLN}$-Schemata definiert, nur daß diese ausschließlich Zahlenrestriktionen der Form $\exists^{\geq 1} R$ und $\exists^{\leq 1} R$ enthalten dürfen. Neben der primitiven Negation sind jedoch außerdem Konzeptterme der Form $\neg B$ für ein *definiertes* Konzept B zugelassen.

Semantik:
Eine *Interpretation I* zu einem $\mathcal{ALN}$- ($\mathcal{SLN}$- , $\mathcal{SL}_{dis}$-)Schema ist wie in Definition 3.4 definiert. Sie *erfüllt* die Konzeptinklusion $A \sqsubseteq D$, falls $A^I \subseteq D^I$ gilt. Entsprechend ist die Rolleninklusion $R \sqsubseteq A \times B$ *erfüllt*, falls $R^I \subseteq A^I \times B^I$ gilt. Eine Interpretation ist *Modell*

von S (S-Modell) genau dann, wenn sie alle S-Inklusionsaxiome erfüllt. Die Semantik von S ist durch die deskriptive Semantik gegeben, d.h. durch die Menge aller S-Modelle.

Ein Konzept A heißt bzgl. S *konsistent* (S-konsistent), falls ein S-Modell I mit $A^I \neq \emptyset$ existiert. Wir nennen S *gültig*, falls ein S-Modell I existiert, so daß *für alle* Konzepte A in S gilt: $A^I \neq \emptyset$. Dagegen heißt S *lokal-gültig*, falls für jedes Konzept A in S ein S-Modell I_A existiert mit $A^{I_A} \neq \emptyset$. Das Konzept A wird von B bzgl. S subsumiert (S-subsumiert, i.Z. $A \sqsubseteq_S B$) genau dann, wenn $A^I \subseteq B^I$ für alle S-Modelle I gilt. $\diamond$

Wir betrachten zunächst ein Beispiel für ein $\mathcal{ALN}$-Schema, das einen Ausschnitt aus einer Wissensbank zur Personalverwaltung darstellt:

Beispiel 7.2.
S:[2]

$$\begin{array}{rcl}
\mathsf{Angestellter} & \sqsubseteq & \exists^{\geq 1}\mathsf{gehalt} \\
\mathsf{Angestellter} & \sqsubseteq & \exists^{\leq 1}\mathsf{gehalt} \\
\mathsf{Angestellter} & \sqsubseteq & \exists^{\geq 1}\mathsf{chef} \\
\mathsf{Manager} & \sqsubseteq & \mathsf{Angestellter} \\
\mathsf{Manager} & \sqsubseteq & \forall \mathsf{gehalt}.\mathsf{HohesGehalt} \\
\mathsf{HohesGehalt} & \sqsubseteq & \mathsf{Gehalt} \\
\mathsf{gehalt} & \sqsubseteq & \mathsf{Angestellter} \times \mathsf{Gehalt} \\
\mathsf{chef} & \sqsubseteq & \mathsf{Angestellter} \times \mathsf{Manager}
\end{array}$$

$\diamond$

Um die Gültigkeit eines Schemas zu entscheiden, reicht die Entscheidung der Konsistenz der einzelnen Konzepte des Schemas. Wir zeigen dazu:

Satz 7.3.
Sind I_1 und I_2 (bel.) Modelle des $\mathcal{ALN}$-Schemas S mit disjunkten Domänen, so existiert ein S-Modell I, so daß für alle $\mathcal{ALN}$-Konzeptterme D gilt: $D^I = D^{I_1} \dot\cup D^{I_2}$.

Beweis: Wir definieren I als Vereinigung von I_1 und I_2, d.h.: $C^I := C^{I_1} \cup C^{I_2}$ für alle Konzepte C in S; $R^I := R^{I_1} \cup R^{I_2}$ für alle Rollen R in S. Ist D ein $\mathcal{ALN}$-Konzeptterm, so zeigt man leicht per Induktion über den Aufbau von Konzepttermen, daß (*) $D^I = D^{I_1} \dot\cup D^{I_2}$ gilt.

Es bleibt zu zeigen, daß I S-Modell ist. Ist $C \sqsubseteq D$ eine S-Konzeptinklusion, so gilt $C^I \stackrel{(*)}{=} C^{I_1} \cup C^{I_2} \stackrel{I_1, I_2\ S\text{-Modelle}}{\subseteq} D^{I_1} \cup D^{I_2} \stackrel{(*)}{=} D^I$. Somit erfüllt I alle S-Konzeptinklusionen. Für eine S-Rolleninklusion $R \sqsubseteq A_1 \times A_2$ ist $R^I = R^{I_1} \dot\cup R^{I_2} \stackrel{I_1, I_2\ S\text{-Modelle}}{\subseteq} {A_1}^{I_1} \times {A_2}^{I_1} \cup {A_1}^{I_2} \times {A_2}^{I_2} \subseteq (A_1^{I_1} \cup A_1^{I_2}) \times (A_2^{I_1} \cup A_2^{I_2}) = {A_1}^I \times {A_2}^I$. Also erfüllt I auch die S-Rolleninklusionen. Dies zeigt, daß I S-Modell ist. $\square$

Dies führt zu folgendem Korollar, welches die Reduktion der Gültigkeit eines Schemas auf die Konsistenz von Konzepten einer Terminologie ermöglichen wird (Abschnitt 7.2).

[2] angelehnt an ein Beispiel aus [BDNS97]

Korollar 7.4.
Für ein $\mathcal{ALN}$-Schema S gilt: S ist gültig gdw. S lokal-gültig ist.

Beweis: „$\Rightarrow$“: trivial.
„$\Leftarrow$“: Es sei S lokal-gültig und $A_1, \ldots, A_n$ seien alle Konzepte in S. Zu jedem Konzept A_i, $1 \leq i \leq n$, in S existiert somit ein S-Modell I_{A_i} mit $A_i^{I_{A_i}} \neq \emptyset$. Wir können o.E. annehmen, daß die Domänen dieser Modelle paarweise disjunkt sind. Per Induktion über n und mit Satz 7.3 zeigt man nun leicht die Existenz eines S-Modells I mit $A_i^I \neq \emptyset$ für alle $1 \leq i \leq n$. Damit ist die Gültigkeit von S gezeigt. □

Wir zeigen im folgenden, daß $\mathcal{ALN}$-Schemata bis auf die Verwendung (beliebiger) Zahlenrestriktionen in einem nun zu spezifizierendem Sinn mit den in [BDNS97] definierten $\mathcal{SL}_{dis}$-Schemata übereinstimmen.

Ein Schema S' verhält sich bzgl. Konsistenz, (lokaler) Gültigkeit und Subsumtion wie ein Schema S, falls folgende Bedingungen erfüllt sind:

1.) Für alle S-Konzepte A gilt: A ist S-konsistent gdw. A ist S'-konsistent; und

2.) S ist lokal-gültig gdw. zu jedem Konzept A aus S ein S'-Modell I_A existiert mit $A^{I_A} \neq \emptyset$; und

3.) S ist gültig gdw. ein S'-Modell I existiert mit $A^I \neq \emptyset$ für alle Konzepte A aus S; und

4.) für alle S-Konzepte A, B gilt: $A \sqsubseteq_S B$ gdw. $A \sqsubseteq_{S'} B$.

Offensichtlich folgt aus 1.) die Eigenschaft 2.) und daraus zusammen mit Satz 7.3 die Eigenschaft 3.).

Wir reduzieren nun $\mathcal{ALN}$-Schemata schrittweise auf $\mathcal{SLN}$-Schemata, so daß die Eigenschaften 1.) – 4.) jeweils erhalten bleiben.

Gemäß Seite 28 kann jeder $\mathcal{ALN}$-Konzeptterm äquivalent als Konjunktion von Termen der Form $\forall W.D$ geschrieben werden, wobei W ein endliches Wort über der Menge der Rollennamen ist (für $W = \varepsilon$ bezeichnet $\forall W.D$ den Term D) und D ein Konzeptname, eine primitive Negation oder eine Zahlenrestriktion. Sind $D_1, \ldots, D_n$ $\mathcal{ALN}$-Konzeptterme, so erfüllt eine Interpretation die Konzeptinklusion $A \sqsubseteq D_1 \sqcap \cdots \sqcap D_n$ gdw. sie die Inklusionen $A \sqsubseteq D_1, \ldots, A \sqsubseteq D_n$ erfüllt. Damit kann (mit linearem Aufwand) jedes $\mathcal{ALN}$-Schema S in ein äquivalentes Schema S' überführt werden, in dem jede rechte Seite einer Konzeptinklusion die genannte Form $\forall W.D$ aufweist. Da S' äquivalent zu S ist, d.h. S und S' besitzen dieselben Modelle, gelten insbesondere die Eigenschaften 1.) – 4.).

Mit Hilfe des folgenden Lemmas reduzieren wir S' nun weiter, so daß die Konzeptinklusionen von der Form sind: $A \sqsubseteq \forall R.B$ oder $A \sqsubseteq D$ für A und B Konzeptnamen, R Rollenname sowie D Konzeptname, primitive Negation oder Zahlenrestriktion. Wir reduzieren S' also zu einem $\mathcal{SLN}$-Schema.

Lemma 7.5.
Es sei $A \sqsubseteq \forall RW.D$ eine Konzeptinklusion mit A, R, W und D wie oben definiert ($W = \varepsilon$ ist dabei zugelassen). Außerdem sei A_1 ein neuer Konzeptname. Dann gilt:

1.) Erfüllt eine Interpretation I die Konzeptinklusion $A \sqsubseteq \forall RW.D$, so erfüllt sie mit $A_1^I := (\forall W.D)^I$ auch die Konzeptinklusionen $A \sqsubseteq \forall R.A_1$ und $A_1 \sqsubseteq \forall W.D$. (Beachte, daß A_1^I o.E. wie oben definiert werden kann, da A_1 ein neues Konzept ist.)

2.) Erfüllt eine Interpretation I die Konzeptinklusion $A \sqsubseteq \forall R.A_1$ und $A_1 \sqsubseteq \forall W.D$, so erfüllt sie auch $A \sqsubseteq \forall RW.D$.

Beweis: zu 1.) Die Interpretation I erfüllt nach Definition die Konzeptinklusion $A_1 \sqsubseteq \forall W.D$. Nach Voraussetzung ist $A^I \subseteq (\forall RW.D)^I$, also $A^I \subseteq (\forall RW.D)^I \overset{\text{Def. I}}{=} (\forall R.A_1)^I$.
zu 2.) Ist $d \in A^I$, so gilt wegen $A^I \subseteq (\forall R.A_1)^I$ die Inklusion $R^I(d) \subseteq A_1^I$. Aus $A_1^I \subseteq (\forall W.D)^I$ folgt dann $R^I(d) \subseteq (\forall W.D)^I$; damit ist $d \in (\forall R.\forall W.D)^I = (\forall RW.D)^I$. □

Wird in einem Schema S die Konzeptinklusion $A \sqsubseteq \forall RW.D$ durch die Konzeptinklusionen $A \sqsubseteq \forall R.A_1$ und $A_1 \sqsubseteq \forall W.D$ für ein neues Konzept A_1 ersetzt, so gilt, daß das dadurch erhaltene Schema S' die Eigenschaften 1.) – 4.) bzgl. S erfüllt. Denn nach Lemma 7.5 kann zu jedem S-Modell I ein S'-Modell I' angeben werden, welches auf den Konzepten und Rollen von S mit I übereinstimmt; umgekehrt ist jedes S'-Modell I auch Modell von S. Durch Iteration der Transformation aus Lemma 7.5 erhält man ein $\mathcal{SLN}$-Schema. Damit gilt

Korollar 7.6.
Zu jedem $\mathcal{ALN}$-Schema S existiert ein $\mathcal{SLN}$-Schema S', das aus S mit linearem Aufwand gewonnen werden kann, so daß die Eigenschaften 1.) bis 4.) von Seite 98 erfüllt sind. □

Werden in $\mathcal{SLN}$-Schemata auch Konzeptinklusionen der Form $A \sqsubseteq \neg B$ mit *definiertem* Konzept B zugelassen, so nimmt dadurch die Ausdrucksstärke nicht zu. Durch $A \sqsubseteq \neg B$ wird lediglich ausgedrückt, daß die Konzepte A und B disjunkt sind. Dies läßt sich auch durch Einführung eines neuen (primitiven) Konzeptes P_B mit Hilfe der primitiven Negation beschreiben. Dazu wird die Inklusion $A \sqsubseteq \neg B$ durch die Inklusionen $A \sqsubseteq \neg P_B$ und $B \sqsubseteq P_B$ ersetzt. Es sei nun S ein Schema, das die Konzeptinklusion $A \sqsubseteq \neg B$ enthält und S' das aus S durch obige Ersetzung gewonnene Schema. Definiert man $P_B^I := B^I$ für ein S-Modell I (Beachte: P_B ist neues Konzept), so ist I auch S'-Modell. Umgekehrt ist ein Modell von S' auch S-Modell. Denn mit $B^I \subseteq P_B^I$ gilt $(\neg B)^I \supseteq (\neg P_B)^I$, woraus mit $A^I \subseteq (\neg P_B)^I$ die Inklusion $A^I \subseteq (\neg B)^I$ folgt. Bzgl. der Behandlung von Konsistenz, (lokaler) Gültigkeit und Subsumtion kann man sich also auf die primitive Negation beschränken.

Die in [BDNS97] eingeführten $\mathcal{SL}_{dis}$-Schemata sind demnach nicht ausdrucksstärker als $\mathcal{SLN}$-Schemata.

Im folgenden Abschnitt werden wir für $\mathcal{SLN}$-Schemata die angesprochene Reduktion der Konsistenz, der (lokalen) Gültigkeit und der Subsumtion bzgl. Schemata auf entsprechende Probleme bzgl. Terminologien angeben. Die daraus gewonnenen Komplexitätsaussagen für die jeweiligen Entscheidungsprobleme gelten mit der hier angegebenen Normalisierung auch für $\mathcal{ALN}$-Schemata bzw. $\mathcal{SL}_{dis}$-Schemata, da die Normalisierung nur linearen Aufwand benötigt.

7.2 Reduktion von Schemata auf Terminologien

Wir werden in diesem Abschnitt die Entscheidungsprobleme Konsistenz, (lokale) Gültigkeit sowie Subsumtion für $\mathcal{SLN}$-Schemata S auf entsprechende Probleme für $\mathcal{ALN}$-Terminologien T_S reduzieren. Diese Reduktion ermöglicht dann die bisherigen Ergebnisse dieser Arbeit zur Komplexität der Inkonsistenz und der Subsumtion auf Schemata zu übertragen.

Konkret erhalten wir die bereits zu Beginn dieses Kapitels genannten Komplexitätsresultate. Zudem stellt die Reduktion den Unterschied der Inferenz bzgl. Schemata auf der einen und bzgl. Terminologien auf der anderen Seite heraus.

7.2.1 Die Terminologie T_S

Bevor wir zum $\mathcal{SLN}$-Schema S die $\mathcal{ALN}$-Terminologie T_S definieren, transformieren wir S in ein Schema S' ohne Rolleninklusionen, so daß die Eigenschaften 1.) bis 4.) von Seite 98 gelten:

Sind A, B Konzepte und R ein Rollenname in S, so enthalte S' alle Konzeptinklusionen der Form $A \sqsubseteq B$, $A \sqsubseteq \neg B$, $A \sqsubseteq \forall R.B$ sowie $A \sqsubseteq \exists^{\leq n} R$ aus S. Es ist zu beachten, daß nach Definition der $\mathcal{SLN}$-Schemata für eine Inklusion der Form $A \sqsubseteq \neg B$, das Konzept B ein primitives Konzept in S ist. Sind A, B Konzepte in S und R ein Rollenname in S, so enthalte S' außerdem diejenigen Konzeptinklusionen der Form $A \sqsubseteq \exists^{\geq n} R$, $n \geq 1$, für die *keine* Rolleninklusionen zu R, d.h. Inklusionen der Form $R \sqsubseteq C_1 \times C_2$, in S existieren. Für Inklusionen $A \sqsubseteq \exists^{\geq n} R$, $n \geq 1$, und $R \sqsubseteq C_1 \times C_2$ in S enthalte S' die Konzeptinklusion $A \sqsubseteq \exists^{\geq n} R \sqcap \forall R.C_2 \sqcap C_1$. Das Schema S' enthalte neben den hier aufgeführten Konzeptinklusionen sonst keine Inklusionen, insbesondere keine Rolleninklusionen.[3]

Die Konstruktion von S' zeigt, daß nicht alle Informationen von S in S' explizit ausgedrückt werden. So wird z.B. in der S'-Konzeptinklusion $A \sqsubseteq \forall R.B$ zur S-Inklusion $A \sqsubseteq \forall R.B$ eine S-Rolleninklusion $R \sqsubseteq C_1 \times C_2$ nicht berücksichtigt. Man würde in S' vielleicht die Konzeptinklusion $A \sqsubseteq \forall R.B \sqcap \forall R.C_2 \sqcap (\exists^{\leq 0} R \sqcup (\exists^{\geq 1} R \sqcap C_1))$ erwarten. Für ein Individuum d in A impliziert die Rolleninklusion $R \sqsubseteq C_1 \times C_2$ nämlich, daß alle R-Nachfolger von d in C_2 liegen müssen und außerdem d keinen R-Nachfolger besitzt oder in C_1 enthalten ist. Die Disjunktion ($\sqcup$) ist für $\mathcal{ALN}$-Schemata jedoch nicht erlaubt. Außerdem zeigt sich, daß die in S' eingeführten Konzeptinklusionen ausreichend sind, die Eigenschaften 1.) – 4.) zu erfüllen.

Wir werden als nächstes mit S' die Terminologie T_S definieren und an T_S (statt an S') nachweisen, daß die Eigenschaften 1.) – 4.) in bezug auf T_S erhalten bleiben.

Definition 7.7 (Terminologie T_S zum Schema S).
Es sei S ein $\mathcal{SLN}$-Schema und das Schema S' wie oben aus S konstruiert. Zu jedem definierten Konzept A aus S (S') enthält T_S ein Axiom, welches man wie folgt erhält:
Es seien $A \sqsubseteq C_1, \ldots, A \sqsubseteq C_n$ alle Konzeptinklusionen in S' zum (definierten) Konzept A. Es sei $\overline{A}$ ein neues (primitives) Konzept. Das Axiom zu A in T_S ist dann folgendermaßen definiert: $A = \overline{A} \sqcap C_1 \sqcap \cdots \sqcap C_n$. ◇

Offensichtlich kann T_S mit linearem Aufwand aus S gewonnen werden.

Man hätte in Definition 7.7 statt ein Axiom der Form $A = \overline{A} \sqcap \cdots$ mit neuem primitiven Konzept $\overline{A}$ einzuführen, auch $A = A \sqcap \cdots$ setzen können. Die Aussagen zur Subsumtion müßten dann aber auf die deskriptive Semantik beschränkt werden. Die in Definition 7.7 angegebene Variante erlaubt dagegen auch Aussagen zur gfp-Semantik.

Bevor wir die Eigenschaften von T_S genauer untersuchen, betrachten wir die Terminologie T_S für das Schema aus Beispiel 7.2.

[3] Es sei bemerkt, daß S' kein $\mathcal{SLN}$-Schema ist, da die letztgenannten Konzeptinklusionen Konzeptkonjunktionen enthalten. Für eine derartige Konzeptinklusion könnten auch alternativ drei Konzeptinklusionen eingeführt werden, womit man wieder ein $\mathcal{SLN}$-Schema erhalten würde.

Beispiel 7.8 (Fortsetzung von Beispiel 7.2).
Das Schema S aus Beispiel 7.2 ist bereits ein $\mathcal{SLN}$-Schema. Die Terminologie T_S lautet wie folgt:

$$
\begin{aligned}
\mathsf{Angestellter} &= \overline{\mathsf{Angestellter}} \sqcap \exists^{\geq 1}\mathsf{gehalt} \sqcap \forall \mathsf{gehalt.Gehalt} \sqcap \mathsf{Angestellter} \sqcap \\
&\quad\ \exists^{\leq 1}\mathsf{gehalt} \sqcap \exists^{\geq 1}\mathsf{chef} \sqcap \forall \mathsf{chef.Manager} \sqcap \mathsf{Angestellter} \\
\mathsf{Manager} &= \overline{\mathsf{Manager}} \sqcap \mathsf{Angestellter} \sqcap \forall \mathsf{gehalt.HohesGehalt} \\
\mathsf{HohesGehalt} &= \overline{\mathsf{HohesGehalt}} \sqcap \mathsf{Gehalt}
\end{aligned}
$$

◇

Bevor wir S und T_S bzgl. Konsistenz untersuchen zeigen wir folgende Hilfsaussage:

Lemma 7.9.
Es sei T eine $\mathcal{ALN}$-Terminologie sowie $\mathcal{A}_T$ der zugehörige Semi-Automat, und es seien A und B Konzepte in T. Weiter sei W ein Wort mit $W \in L_{\mathcal{A}_T}(A, B)$ sowie I ein Modell von T. Für die Individuen $d, e \in dom(I)$ gelte $dW^I e$ und $d \in A^I$. Dann ist $e \in B^I$.

Beweis: Es sei $A = B_1, U_1, B_2, U_2, \ldots, U_n, B_{n+1} = B$ ein Pfad mit Label $W = U_1 \cdots U_n$ von A nach B, so daß (B_i, U_i, B_{i+1}) für alle $1 \leq i \leq n$ Transitionen in $\mathcal{A}_T$ sind. Wir zeigen die Behauptung per Induktion über die Länge n des Pfades.
Induktionsanfang (n=0): Es ist $A = B$ sowie $W = \varepsilon$ und somit $d = e$ wegen $dW^I e$. Aus $d \in A^I$ folgt also $e \in B^I$.

Induktionsschritt: Wegen $dW^I e$ existiert ein $f \in dom(I)$ mit $dU_1^I f (U_2 \cdots U_{n+1})^I e$. Das Axiom zu A hat die Form $A = \cdots \sqcap \forall U_1.B_2 \sqcap \cdots$. Da I Modell von T ist, muß wegen $d \in A^I$ und $dU_1^I f$ das Individuum f in B_2^I enthalten sein. Nach Induktionsvoraussetzung ist damit $e \in B^I$. □

Im nächsten Satz zeigen wir nun, daß die Konsistenz/Inkonsistenz von Konzepten aus S in T_S erhalten bleibt (Eigenschaft 1.), Seite 98).

Satz 7.10 (Konsistenz).
Es sei S ein $\mathcal{SLN}$-Schema und T_S die gemäß Definition 7.7 zugehörige $\mathcal{ALN}$-Terminologie. Für alle Konzepte A aus S gilt: A ist S-konsistent gdw. A ist T_S-konsistent bzgl. der gfp-Semantik.

Beweis: „⇒“: Es sei A ein S-konsistentes S-Konzept, d.h. es existiert ein S-Modell I mit $A^I \neq \emptyset$. Wir definieren ein gfp-Modell $\overline{I}$ zu T_S mit $A^{\overline{I}} \neq \emptyset$. Durch $\overline{I}$ werden dabei alle Konzepte und Rollen aus S wie durch I interpretiert. Zusätzlich wird das zu einem definierten Konzept B aus T_S gehörende primitive Konzept $\overline{B}$ durch $\overline{B}^{\overline{I}} := B^I$ interpretiert.
Behauptung: $\overline{I}$ ist ein gfp-Modell von T_S mit $A^{\overline{I}} \neq \emptyset$.
Denn: Um zu zeigen, daß $\overline{I}$ ein Modell von T_S ist, genügt es zu zeigen, daß jede S'-Konzeptinklusion $B \sqsubseteq C$ (vgl. Seite 100) bzgl. $\overline{I}$ erfüllt ist (*). Ist nämlich $B = \overline{B} \sqcap C_1 \sqcap \cdots \sqcap C_n$ das Axiom zu B in T_S, wobei $B \sqsubseteq C_i$, $1 \leq i \leq n$, die S'-Konzeptinklusionen von B sind, so gilt wegen $\overline{B}^{\overline{I}} = B^{\overline{I}}$ die Inklusion $(\overline{B} \sqcap C_1 \sqcap \cdots \sqcap C_n)^{\overline{I}} \subseteq B^{\overline{I}}$. Umgekehrt ist wegen $\overline{B}^{\overline{I}} = B^{\overline{I}}$ und $B^{\overline{I}} \subseteq C_i^{\overline{I}}$ (nach (*)) auch $B^{\overline{I}} \subseteq (\overline{B} \sqcap C_1 \sqcap \cdots \sqcap C_n)^{\overline{I}}$.

Wir zeigen (*): Es sei $B \sqsubseteq C$ eine S'-Konzeptinklusion. Ist $B \sqsubseteq C$ auch eine S-Konzeptinklusion, so gilt nach Definition von $\overline{I}$, und da I ein S-Modell ist, direkt $B^{\overline{I}} \subseteq C^{\overline{I}}$. Ist $B \sqsubseteq C$ keine Konzeptinklusion in S, so ist $C = \exists^{\geq n} R \sqcap \forall R.C_2 \sqcap C_1$, $n \geq 1$. In S sind also die Inklusionsaxiome $B \sqsubseteq \exists^{\geq n}R$ und $R \sqsubseteq C_1 \times C_2$ enthalten. Wegen $R^{\overline{I}} = R^I$ sowie $C_1^{\overline{I}} = C_1^I$ und $C_2^{\overline{I}} = C_2^I$ gilt $B^{\overline{I}} \subseteq (\exists^{\geq n}R)^{\overline{I}}$ und $R^{\overline{I}} \subseteq {C_1}^{\overline{I}} \times {C_2}^{\overline{I}}$. Damit ist $(\exists^{\geq n}R)^{\overline{I}} \subseteq C_1^{\overline{I}}$ (beachte: $n \geq 1$) und $(\forall R.C_2)^{\overline{I}} = dom(\overline{I})$, woraus insgesamt $B^{\overline{I}} \subseteq C^{\overline{I}}$ folgt. Dies zeigt, daß $\overline{I}$ Modell von T_S ist.

Wir zeigen nun, daß $\overline{I}$ sogar gfp-Modell von T_S ist. Es sei dazu I' ein Modell von T_S, so daß die primitiven Interpretationen zu I' und $\overline{I}$ übereinstimmen. Wegen $\overline{B}^{\overline{I}} = \overline{B}^{I'}$, $B^{\overline{I}} = \overline{B}^{\overline{I}}$ und $B^{I'} \subseteq \overline{B}^{I'}$ (I' erfüllt $B = \overline{B} \sqcap \cdots$) gilt $B^{I'} \subseteq B^{\overline{I}}$ für alle definierten Konzepte B. Also ist $\overline{I}$ ein gfp-Modell von T_S.

Wegen $A^I \neq \emptyset$ und $A^{\overline{I}} = A^I$ folgt weiter $A^{\overline{I}} \neq \emptyset$. Damit ist die Behauptung gezeigt, und es folgt für die gfp-Semantik die T_S-Konsistenz von A.

„$\Leftarrow$“: Es sei A ein Konzept in S, das bzgl. T_S und der gfp-Semantik konsistent ist. Wir zeigen die Existenz eines Modells I von S mit $A^I \neq \emptyset$. Da A konsistent bzgl. T_S ist, existiert nach Bemerkung 6.12 das kanonische gfp-Modell $I = I(A, d_0)$ zu A und Individuum d_0 mit $d_0 \in A^I$. Die zugehörige primitive Interpretation werde wiederum mit $J = J(A, d_0)$ bezeichnet.

Behauptung: I ist Modell von S.
Denn: Es sei $B \sqsubseteq C$ ($C \neq \exists^{\geq 0}R$) eine Konzeptinklusion in S. Das Axiom zu B in T_S hat die Form $B = \overline{B} \sqcap \cdots \sqcap C \sqcap \cdots$. Da I dieses Axiom erfüllt, ist $B^I \subseteq C^I$. Ist $C = \exists^{\geq 0}R$, so gilt wegen $C^I = dom(I)$ ebenfalls $B^I \subseteq C^I$. Dies zeigt, daß I alle S-Konzeptinklusionen erfüllt.

Es sei nun $R \sqsubseteq C_1 \times C_2$ eine Rolleninklusion in S und $(d, e) \in R^I$ für Individuen d, e. Es ist $(d, e) \in {C_1}^I \times {C_2}^I$ zu zeigen. Nach Lemma 6.11, 1.) existiert zu d eindeutig ein von A gefordertes Wort W mit $d_0 W^J d$. Aus Lemma 6.11, 1.) folgt wegen $d_0 W^J d R^J e$ ebenfalls, daß das Wort WR von A gefordert wird; insbesondere gilt also $W \in L_{\mathcal{A}_{T_S}}(A, \exists^{\geq n}R)$ für ein $n \geq 1$. Nach Definition von T_S tritt $\exists^{\geq n}R$ in einem Axiom nur in der Form $\cdots \sqcap \exists^{\geq n}R \sqcap \cdots$ auf. Demnach existiert ein definiertes Konzept C in T_S, so daß $A, W, C, \varepsilon, \exists^{\geq n}R$ ein Pfad mit Label W in $\mathcal{A}_{T_S}$ ist. Das Axiom zu C hat die Form $C = \overline{C} \sqcap \cdots \sqcap \exists^{\geq n}R \sqcap \cdots$. Nach Definition von T_S ist dann $C \sqsubseteq \exists^{\geq n}R$ eine S-Konzeptinklusion. Da $R \sqsubseteq C_1 \times C_2$ eine S-Rolleninklusion ist, gilt für das Axiom zu C genauer: $C = \overline{C} \sqcap \cdots \sqcap \exists^{\geq n}R \sqcap \forall R.C_2 \sqcap C_1 \sqcap \cdots$. Lemma 7.9 liefert $d \in C_1^I$ wegen $d_0 W^I d$ und $W \in L_{\mathcal{A}_{T_S}}(A, C_1)$. Aus $d_0 (WR)^I e$ und $WR \in L_{\mathcal{A}_{T_S}}(A, C_2)$ folgt mit Lemma 7.9 außerdem $e \in C_2^I$. Damit ist $(d, e) \in {C_1}^I \times {C_2}^I$. Dies zeigt, daß I die S-Rolleninklusionen erfüllt. Damit ist I Modell von S, also ist die Behauptung bewiesen.

Zusammen mit $d_0 \in A^I$ folgt aus der Behauptung die Konsistenz von A bzgl. S. □

Der Beweis von Satz 7.10 macht deutlich, daß ein Modell I von S leicht zu einem T_S-Modell erweitert werden kann, so daß die gemeinsamen Konzepte von S und T_S identisch interpretiert werden. Der Beweis von rechts nach links in Satz 7.10 benutzt, daß zu einem Individuum $d \in dom(I)$ nur dann ein R-Nachfolger existieren muß, falls er explizit gefordert wird. Die Überführung der Inklusionen $A \sqsubseteq \exists^{\geq n}R$, $n \geq 1$, und $R \sqsubseteq C_1 \times C_2$ von Schema S in die Konzeptinklusion $A \sqsubseteq \exists^{\geq n}R \sqcap \forall R.C_2 \sqcap C_1$ berücksichtigt damit die Eigenschaften, die von der Rolleninklusion gefordert werden. Es ist deshalb nicht nötig, die Rolleninklusionen selbst in S' aufzunehmen.

Satz 7.10 gilt genauso, wenn statt der gfp-Semantik die deskriptive Semantik zugrunde gelegt wird (vgl. Lemma 5.20). Der Satz gilt ebenfalls, wenn statt der Axiome $A = \overline{A} \sqcap \cdots$ Axiome der Form $A = A \sqcap \cdots$ in T_S gewählt werden. Die Richtung „$\Leftarrow$“ des Beweisen wird sowohl für die gfp- als auch für die deskriptive Semantik analog zum vorgestellten

Beweis geführt. In der Richtung von links nach rechts zeigt man für das S-Modell I direkt, daß I Modell von T_S ist, d.h. man weist statt der Behauptung im obigen Beweis die folgende Behauptung nach: (*) I ist Modell von T_S ($A^I \neq \emptyset$ gilt nach Voraussetzung). Für die deskriptive Semantik ist der Beweis damit abgeschlossen. Da i. allg. nicht gilt: I ist gfp-Modell von T_S, kann für die gfp-Semantik der Beweis auf diesem Wege nicht geführt werden. Jedoch folgt die nachzuweisende Konsistenz des T_S-Konzeptes A für die gfp-Semantik aus (*), d.h. der Konsistenz bzgl. der deskriptiven Semantik, und Lemma 5.20.

Da nach Korollar 7.4 die Gültigkeit und die lokale Gültigkeit eines Schemas übereinstimmen, folgt mit Satz 7.10 das

Korollar 7.11 (Gültigkeit).
Mit den Bezeichnungen aus Satz 7.10 gilt: S ist gültig gdw. alle S-Konzepte bzgl. T_S und (der deskriptiven Semantik bzw.) der gfp-Semantik konsistent sind. □

In Kapitel 6.2 wurde gezeigt, daß Inkonsistenz, und damit Konsistenz, eines Konzeptes mit polynomialem Platzaufwand entscheidbar sind. Satz 7.10 und Korollar 7.11 implizieren somit, daß sowohl Konsistenz als auch Gültigkeit für $\mathcal{SLN}$-($\mathcal{ALN}$-, $\mathcal{SL}_{dis}$-)Schemata mit einem PSPACE-Algorithmus entscheidbar sind.

In [BDNS97] wurde nachgewiesen, daß das Gültigkeitsproblem für $\mathcal{SL}_{dis}$-Schemata in PSPACE enthalten ist. Um die Unerfüllbarkeit des Konzeptterms $A_1 \sqcap \cdots \sqcap A_n$ bzgl. eines $\mathcal{SL}_{dis}$-Schemas S zu zeigen, wurde ein Graph $\mathcal{G}_S$ definiert, so daß der Konzeptterm $A_1 \sqcap \cdots \sqcap A_n$ genau dann inkonsistent ist, wenn vom $\mathcal{G}_S$-Knoten $\{A_1, \ldots, A_n\}$ aus ein Pfad zu einem sogenannten Konfliktknoten existiert. Um die Inkonsistenz des Konzeptterms $A_1 \sqcap \cdots \sqcap A_n$ bzgl. der Terminologie T_S zu testen, ist nach Satz 6.15 zu entscheiden, ob $\varepsilon\text{-}closure(\{A_1, \ldots, A_n\})$ ein Ausschlußzustand ist. Algorithmus 6.14 berechnet dazu nichtdeterministisch eine Folge von Mengen, bis eine Menge erreicht wird, die widersprüchliche Zahlenrestriktionen bzw. ein Paar bestehend aus P und $\neg P$ für ein primitives Konzept P enthält. Eine derartige Menge entspricht einem Konfliktknoten im Graphen $\mathcal{G}_S$. Es zeigt sich, daß Algorithmus 6.14 im wesentlichen einen Teil des Graphen $\mathcal{G}_S$ durchläuft. Damit wird die Inkonsistenz für Schemata und die für Terminologien algorithmisch auf ähnliche Weise behandelt.

In [BDNS97] wurde die co-NP-Härte der Gültigkeit von $\mathcal{SL}_{dis}$-Schemata gezeigt. Satz 7.10 erlaubt damit, die co-NP-Härte der Konsistenz für $\mathcal{ALN}$- ($\mathcal{FLN}$-)Terminologien zu zeigen, und somit die Aussage des Satzes 6.17. Dazu benutzen wir, daß ein Problem (hier die Konsistenz) genau dann co-NP-hart ist, wenn das komplementäre Problem (hier die Inkonsistenz) NP-hart ist.

Beweis von Satz 6.17: Nach [DHL$^+$92] ist die Konsistenz (Erfüllbarkeit) sogenannter eingeschränkter $\mathcal{ALE}$-Konzeptterme co-NP-vollständig[4]. In [BDNS97], Lemma 4.5 wird gezeigt, daß zu jedem eingeschränkten $\mathcal{ALE}$-Konzeptterm C ein $\mathcal{SL}_{dis}$-Schema S_C existiert, so daß folgende Aussage gilt:

$$S_C \text{ ist gültig gdw. } C \text{ konsistent ist.} \tag{7.1}$$

Damit ist die Konsistenz von eingeschränkten $\mathcal{ALE}$-Konzepttermen auf die Gültigkeit von Schemata polynomial reduziert worden, also die co-NP-Härte der Gültigkeit gezeigt.

[4]Da der Beweis aus [BDNS97] zur co-NP-Härte der Gültigkeit hier nur skizziert wird, ist die genaue Kenntnis von eingeschränkten $\mathcal{ALE}$-Konzepttermen an dieser Stelle nicht wichtig, weshalb hier auf eine formale Definition verzichtet wird.

Die Beweisidee zu (7.1), „$\Rightarrow$“ ist die folgende: In S_C wird für jedes Teilkonzept D von C ein Konzeptterm A_D eingeführt. Ein (beliebiges) Modell I von S_C wird (auf bestimmte Weise) zu einer Interpretation (diese sei auch mit I bezeichnet) von C erweitert. Man zeigt nun per Induktion über den Termaufbau, daß (+) $A_D^I \subseteq D^I$ für alle Teilterme D von C gilt. Ist außerdem I nicht nur S_C-Modell, sondern liefert I außerdem die Gültigkeit von S_C, d.h. die Extensionen aller Konzepte aus S_C werden bzgl. I ungleich der leeren Menge interpretiert, so gilt insbesondere $A_C^I \neq \emptyset$. Dies impliziert aber $C^I \neq \emptyset$ wegen $A_C^I \subseteq C^I$, also die Konsistenz von C.

Da (+) allein mit der Voraussetzung nachgewiesen werden kann, daß I ein S_C-Modell ist, genügt für die Konsistenz von C die S_C-Konsistenz von A_C. Man überzeugt sich in der Tat leicht, daß die Gültigkeit von S_C nicht gefordert werden muß. Es gilt also: Ist A_C S_C-konsistent, so ist C konsistent.

Die Umkehrung dieser Aussage gilt ebenfalls, denn: Ist C konsistent, so impliziert (7.1) die Gültigkeit von S_C; insbesondere ist demnach A_C konsistent.

Dies zeigt insgesamt:

$$A_C \text{ ist } S_C\text{-konsistent gdw. C konsistent ist.} \tag{7.2}$$

Nach Satz 7.10 ist die linke Seite dieser Äquivalenzaussage gleichbedeutend damit, daß A_C konsistent bzgl. der zu S_C gehörenden $\mathcal{ALN}$-Terminologie T_{S_C} und der gfp- bzw. der deskriptiven Semantik ist. Nun ist nach Definition von S_C (vgl. [BDNS97]) und Konstruktion von T_{S_C} die Terminologie T_{S_C} sogar azyklisch. Damit stimmen gfp-, lfp- und deskriptive Semantik für T_{S_C} überein. Das Konzept A_C in T_{S_C} ist also genau dann bzgl. der gfp- und der deskriptiven Semantik konsistent, wenn es bzgl. der lfp-Semantik konsistent ist.

Damit ist die Konsistenz von C für alle drei Semantiken auf die Konsistenz von A_C bzgl. der Terminologie T_{S_C} reduziert. Dies schließt den Beweis zu Satz 6.17 ab. $\square$

Die co-NP-Härte gilt wegen (7.1) auch für die Gültigkeit und wegen (7.2) auch für die Konsistenz von $\mathcal{ALN}$-(bzw. $\mathcal{SLN}$-)Schemata.

Nachdem wir uns davon überzeugt haben, daß sich T_S bzgl. der Konsistenz (Eigenschaft 1.), Seite 98) und der (lokalen) Gültigkeit (Eigenschaft 2.) bzw. 3.)) auf den Konzepten von S wie S verhält, betrachten wir im folgenden Satz die Subsumtion (Eigenschaft 4.)).

Satz 7.12 (Subsumtion).
Es sei S ein $\mathcal{SLN}$-Schema und T_S die zugehörige $\mathcal{ALN}$-Terminologie. Für alle Konzepte A, B aus S gilt: $A \sqsubseteq_S B$ gdw. $A \sqsubseteq_{T_S} B$ (gdw. $A \sqsubseteq_{gfp,T_S} B$).

Beweis: „$\Leftarrow$“: Gilt $A \not\sqsubseteq_S B$, so existiert ein Modell I von S und ein Individuum d mit $d \in A^I \setminus B^I$. Es sei $\overline{I}$ wie im Beweis zu Satz 7.10 („$\Leftarrow$“) definiert. Damit ist $\overline{I}$ ein (gfp-)Modell von T_S, in dem die Konzepte aus S wie in I interpretiert werden. Also gilt $d \in A^{\overline{I}} \setminus B^{\overline{I}}$ und somit $A \not\sqsubseteq_{T_S} B$ (bzw. $A \not\sqsubseteq_{gfp,T_S} B$).

„$\Rightarrow$“: Gilt $A \not\sqsubseteq_{T_S} B$ (bzw. $A \not\sqsubseteq_{gfp,T_S} B$), so existiert ein (gfp-)Modell I von T_S und ein Individuum d mit $d \in A^I \setminus B^I$. Wir definieren nun ein S-Modell I', so daß $d \in A^{I'} \setminus B^{I'}$ gilt und damit $A \not\sqsubseteq_S B$:
$dom(I') := dom(I)$; $C^{I'} := C^I$ für alle Konzepte C in S; ist R ein S-Rollenname und sind $R \sqsubseteq C_1^1 \times C_2^1, \ldots, R \sqsubseteq C_1^n \times C_2^n$ alle S-Rolleninklusionen zu R, so sei $R^{I'} := R^I \cap ((C_1^1)^I \times (C_2^1)^I) \cap \cdots \cap ((C_1^n)^I \times (C_2^n)^I)$ (für $n = 0$ ist $R^{I'} = R^I$).

Nach Definition von I' sind die Rolleninklusionen von S bzgl. I' erfüllt.

Für eine S-Konzeptinklusion $C \sqsubseteq D$ (bzw. $C \sqsubseteq \neg D$) mit Konzepten C und D hat das Axiom zu C in T_S die Form $C = \overline{C} \sqcap \cdots \sqcap D \sqcap \cdots$ (bzw. $C = \overline{C} \sqcap \cdots \sqcap \neg D \sqcap \cdots$). Da I Modell von T_S ist sowie $C^{I'} = C^I$ und $D^{I'} = D^I$ gilt, folgt $C^{I'} \subseteq D^{I'}$ (und wegen $dom(I') = dom(I)$ auch $C^{I'} \subseteq (\neg D)^{I'}$). Die Konzeptinklusion $C \sqsubseteq D$ (bzw. $C \sqsubseteq \neg D$) ist bzgl. I' also erfüllt.

Ist $C \sqsubseteq \forall R.D$ eine S-Konzeptinklusion, so gilt wegen $R^{I'} \subseteq R^I$ und $D^{I'} = D^I$ die Inklusion $(\forall R.D)^I \subseteq (\forall R.D)^{I'}$. Weiter hat das Axiom zu C in T_S die Form $C = \overline{C} \sqcap \cdots \sqcap \forall R.D \sqcap \cdots$. Demnach gilt $C^I \subseteq (\forall R.D)^I$. Dies liefert zusammen mit $C^{I'} = C^I$ die Inklusion $C^{I'} \subseteq (\forall R.D)^{I'}$. Entsprechend zeigt man wegen $R^{I'} \subseteq R^I$, daß I' die Inklusion $C \sqsubseteq \exists^{\leq n} R$ aus S erfüllt.

Eine Konzeptinklusion $C \sqsubseteq \exists^{\geq 0} R$ in S wird von I' wegen $(\exists^{\geq 0} R)^{I'} = dom(I')$ trivial erfüllt.

Ist $C \sqsubseteq \exists^{\geq n} R, n \geq 1$, eine S-Konzeptinklusion und sind $R \sqsubseteq C_1^1 \times C_2^1, \ldots, R \sqsubseteq C_1^m \times C_2^m$ die S-Rolleninklusionen zu R, so ist das Axiom zu C in T_S von der Form $C = \overline{C} \sqcap \cdots \sqcap \exists^{\geq n} R \sqcap \forall R.C_2^1 \sqcap \cdots \sqcap \forall R.C_2^m \sqcap C_1^1 \sqcap \cdots \sqcap C_1^m \sqcap \cdots$. Da I ein (gfp-)Modell von T_S ist, gilt für ein Individuum $d \in C^I$, daß d mindestens n verschiedene R-Nachfolger $e_1, \ldots, e_n$ besitzt, für die $e_1, \ldots, e_n \in (C_2^1 \sqcap \cdots \sqcap C_2^m)^I$ gilt. Außerdem gilt $d \in (C_1^1 \sqcap \cdots \sqcap C_1^m)^I$. Damit sind nach Definition von $R^{I'}$ die Individuen $e_1, \ldots, e_n$ auch R-Nachfolger von d bzgl. I' . Also ist $d \in (\exists^{\geq n} R)^{I'}$. Damit erfüllt I' die Konzeptinklusion $C \sqsubseteq \exists^{\geq n} R$.

Insgesamt folgt, daß I' Modell von S ist. Wegen $A^{I'} = A^I$ und $B^{I'} = B^I$ folgt weiter $d \in A^{I'} \setminus B^{I'}$, also $A \not\sqsubseteq_S B$. □

Im Beweis zu Satz 7.12 („$\Leftarrow$“) nutzen wir wie im Satz 7.10 aus, daß ein Modell I von S leicht zu einem (gfp-)Modell von T_S erweitert werden kann. In der umgekehrten Richtung des Beweises ist dagegen die Konstruktion eines Modells zu S aus einem Modell zu T_S aufwendiger, da T_S im Gegensatz zu S keine expliziten Bedingungen an Rollen stellt. Wie in Satz 7.10 genügt aber die Berücksichtung der Rolleninklusionen in Verbindung mit Maximum-Restriktionen, um ein Modell zu S definieren zu können.

Würde man in Satz 7.12 in T_S nicht Axiome der Form $A = \overline{A} \sqcap \cdots$ wählen, sondern solche der Form $A = A \sqcap \cdots$, so gilt die „gfp-Variante“ des Satzes 7.12 im allgemeinen nicht mehr. Dies legt der Beweis nahe, da $\overline{I}$ („$\Leftarrow$“) nur noch ein Modell, nicht aber ein gfp-Modell von T_S sein muß.

Da nach Korollar 6.26 und Korollar 6.32 die Subsumtion bzgl. einer $\mathcal{ALN}$-Terminologie und der gfp- bzw. der deskriptiven Semantik mit einem PSPACE-Algorithmus entscheidbar ist, impliziert Satz 7.12, daß auch das Subsumtionsproblem bzgl. eines $\mathcal{SLN}$- (bzw. $\mathcal{ALN}$-, $\mathcal{SL}_{dis}$-) Schemas in PSPACE liegt.

In [BDNS97] wurde dieses PSPACE-Ergebnis für die Subsumtion bzgl. $\mathcal{SL}_{dis}$-Schemata gezeigt. Dabei liegt die Komplexität des Subsumtionsproblems für $\mathcal{SL}_{dis}$-Schemata nur im Testen der Konsistenz, denn nach [BDNS97], Proposition 4.15 gilt folgendes:

> Für ein $\mathcal{SL}_{dis}$-Schema S und Konzeptnamen $A, A_1, \ldots, A_m$ sowie den S-*konsistenten* Konzeptterm $A_1 \sqcap \cdots \sqcap A_m$ gilt: (7.3)
> $A_1 \sqcap \cdots \sqcap A_m \sqsubseteq_S A$ gdw. $A \in \varepsilon\text{-}closure_{\mathcal{A}_{T_S}}(\{A_1, \ldots, A_m\})$

In [BDNS97] ist die rechte Seite der Äquivalenz anders formuliert und zusätzlich wird die sogenannte „isa-Vollständigkeit“ von S gefordert. „isa-vollständig“ bedeutet, daß mit den Inklusionsaxiomen $R \sqsubseteq A_2 \times B$ und $A_1 \sqsubseteq \exists^{\geq 1} R$ die Konzeptinklusion $A_1 \sqsubseteq A_2$ in S enthalten sein muß; unabhängig von der „isa-Vollständigkeit“ erhält man jedoch wegen der besonderen Berücksichtigung von Maximum-Restriktionen die gleiche Terminologie T_S, so daß in (7.3) auf die „isa-Vollständigkeit“ verzichtet werden kann. Weiterhin bezeichne die auf Konzepten definierte Relation $\preceq_S$ den reflexiv-transitiven Abschluß von Konzeptinklusionen der Form $A \sqsubseteq B$ in S. Damit ist in [BDNS97] die rechte Seite der Äquivalenzaussage wie folgt formuliert: es existiert ein A_i mit $A_i \preceq_S A$. Man überzeugt sich mit der Definition von T_S leicht, daß $A \preceq_S B$ gleichbedeutend mit $B \in \varepsilon\text{-}closure(\{A\})$ ist. Damit ist die Formulierung von (7.3) zulässig.

Bemerkung 7.13.
Für beliebige Terminologien gilt (7.3) nicht. Betrachtet man z.B. die aus den zwei Axiomen $\mathsf{A} = \mathsf{C}$ und $\mathsf{B} = \mathsf{C}$ bestehende Terminologie T mit einem atomaren Konzept C, so sind die Konzepte A, B konsistent sowie äquivalent, obwohl weder $\mathsf{A} \in \varepsilon\text{-}closure(\{\mathsf{B}\})$ noch $\mathsf{B} \in \varepsilon\text{-}closure(\{\mathsf{A}\})$ gilt.

Für Terminologien T_S, die von Schemata stammen, hat jedes Axiom die Form $A = \overline{A} \sqcap \cdots$ mit einem atomaren Konzept $\overline{A}$, welches nur in diesem Axiom auftritt. Es handelt sich also um spezielle Terminologien. Dies führt dazu, daß ein konsistentes Konzept A von einem Konzept B nur dann subsumiert wird, falls die Subsumtion explizit in der T-Box enthalten ist, d.h. im Automaten $\mathcal{A}_{T_S}$ ein ε-Pfad von A nach B führt. ◇

Darüber hinaus ergeben sich für das Subsumtionsproblem bzgl. (schwach-)azyklischer Schemata im Vergleich zu (schwach-)azyklischen Terminologien sogar Komplexitätsunterschiede.

7.2.2 Schwach-azyklische Terminologien und Schemata

Definition 7.14 (schwach-azyklisch).
Eine $\mathcal{ALN}$-Terminologie T heißt *schwach-azyklisch* gdw. für alle nicht-leeren Wörter $W \in \Sigma^+$ und alle (definierten) Konzepte A in T gilt: $W \notin L_{\mathcal{A}_T}(A, A)$.

Ein Schema S heißt *schwach-azyklisch*, falls die zugehörige Terminologie T_S schwach-azyklisch ist. ◇

Bevor wir auf die Unterschiede von schwach-azyklischen Terminologien und schwach-azyklischen Schemata bzgl. der Subsumtion eingehen, werden wir zunächst einige Komplexitätsaussagen zur Konsistenz formulieren.

Konsistenz für schwach-azyklische Terminologien ist „leichter“ zu entscheiden als für beliebige.

Satz 7.15.
Das Konsistenzproblem (bzgl. der gfp-, der lfp- sowie der deskriptiven Semantik) für schwach-azyklische $\mathcal{ALN}$- ($\mathcal{FLN}$-)Terminologien liegt in co-NP.

Beweis: Es sei T eine $\mathcal{ALN}$- ($\mathcal{FLN}$-)Terminologie, und es sei $\mathcal{A}_T$ der zugehörige Semi-Automat ohne Worttransitionen (vgl. Bemerkung 5.2) sowie A ein Konzept in T. Nach Satz 6.15 bzw. Satz 6.39 ist A genau dann inkonsistent, wenn die Menge $\varepsilon\text{-}closure_{\mathcal{A}_T}(\{A\})$ ein Ausschlußzustand ist. Um die Behauptung des Satzes zu zeigen, ist also nur nachzuweisen, daß bzgl. einer schwach-azyklischen Terminologie für den Test auf Ausschlußzustand ein NP-Algorithmus ausreichend ist. In der Tat genügt es im Algorithmus 6.14 und im Algorithmus 6.38 für schwach-azyklische Terminologien, die while-Schleife nur bis $z = n + 1$, $n = |Q|$, laufen zu lassen, denn:

Es sei $F \subseteq Q$. Für $F_0 := F$ und $F_i := next_\varepsilon(F_{i-1}, R_i)$, $R_i \in \Sigma$ (bel.), $i \geq 1$, gilt: $F_{n+1} = \emptyset$. Wäre $F_{n+1} \neq \emptyset$, so existieren definierte Konzepte $A_0, \ldots, A_n$ sowie ein Konzept, eine Zahlenrestriktion oder eine primitive Negation A_{n+1}, so daß $A_0, R_1, A_1, R_2, \ldots, R_{n+1}, A_{n+1}$ ein Pfad in $\mathcal{A}_T$ ist. Da $\mathcal{A}_T$ nur n Zustände besitzt, existieren Zahlen i, j mit $0 \leq i < j \leq n$ und $A_i = A_j$. Damit gilt $\varepsilon \neq R_{i+1} \cdots R_j \in L(A_i, A_j)$. Dies widerspricht der Voraussetzung, daß T schwach-azyklisch ist.

Es kann also für Algorithmus 6.14 und Algorithmus 6.38 die while-Bedingung $z < 2^{|Q|}$ in $z < |Q| + 1$ geändert werden, da für $z = |Q| + 1$ die Variable F gleich der leeren Menge ist. Für Algorithmus 6.38 muß dann zudem in (2) immer „nein" statt „ja" ausgegeben werden, da eine Zustandsmenge nicht aufgrund von Definition 6.37, 1.) Ausschlußzustand sein kann. Damit erhalten wir die gewünschten NP-Algorithmen. $\square$

Wie im Beweis zu Satz 6.17 (Seite 103) erwähnt, ist das Schema S_C azyklisch (d.h. T_{S_C} ist azyklisch). Dies liefert, daß Konsistenz und Gültigkeit (schwach-)azyklischer $\mathcal{SLN}$- (bzw. $\mathcal{SL}_{dis}$-)Schemata und (schwach-)azyklischer $\mathcal{ALN}$-($\mathcal{FLN}$-)Terminologien co-NP-hart sind. Zusammen mit Satz 7.15 gilt also für Terminologien

Korollar 7.16.
Das Konsistenzproblem (bzgl. der gfp-, der lfp- sowie der deskriptiven Semantik) für (schwach-)azyklische $\mathcal{ALN}$- ($\mathcal{FLN}$-)Terminologien ist co-NP-vollständig. $\square$

Satz 7.15 zusammen mit Satz 7.10 liefert, daß die Konsistenz (schwach-)azyklischer $\mathcal{SLN}$-Schemata in co-NP enthalten ist. Korollar 7.4 impliziert damit, daß auch die Gültigkeit (schwach-)azyklischer $\mathcal{SLN}$-Schemata ein co-NP-Problem ist. Damit gilt, daß das Konsistenz- und das Gültigkeitsproblem für (schwach-)azyklische $\mathcal{SLN}$- (bzw. $\mathcal{ALN}$-, $\mathcal{SL}_{dis}$-)Schemata ebenfalls co-NP-vollständig sind.[5]

Der angesprochene Komplexitätsunterschied der Subsumtion bzgl. Schemata auf der einen und der Subsumtion bzgl. Terminologien auf der anderen Seite ergibt sich nun aus

Korollar 7.17.
Die Subsumtion für (schwach-)azyklische $\mathcal{SL}_{dis}$-Schemata ist NP-vollständig.

Beweis: Da die Konsistenz für (schwach-)azyklische $\mathcal{SL}_{dis}$-Schemata co-NP-vollständig ist, ist die Inkonsistenz NP-vollständig. Mit Proposition 4.15 aus [BDNS97] (vgl. auch (7.3)) kann man für die Subsumtion folgenden NP-Algorithmus für das Problem $A \sqsubseteq_S B$ angeben:

1.) Teste mit NP-Algorithmus die S-Inkonsistenz von A.

2.) Liefert 1. die Ausgabe „ja", so gib „ja" aus.

3.) Liefert 1. die Ausgabe „nein", so gib „ja" aus, falls $B \in \varepsilon\text{-}closure_{\mathcal{A}_{T_S}}(\{A\})$ gilt, sonst gib „nein" aus.

Vollständigkeit: Wird A von B S-subsumiert, dann sind zwei Fälle zu unterscheiden:
a) Das Konzept A ist S-inkonsistent. Damit existiert in 1.) eine Berechnung mit Ausgabe „ja". Dies liefert in 2.) die Ausgabe „ja".

[5] In [BDNS97] wurde gezeigt, daß die Gültigkeit für die Klasse der sogenannten für-alle-zykelfreien $\mathcal{SL}_{dis}$-Schemata, d.h. eine Unterklasse der schwach-azyklischen $\mathcal{SL}_{dis}$-Schemata, co-NP-vollständig ist.

b) Das Konzept A ist S-konsistent. Dann liefert 1.) immer „nein“. Nach [BDNS97], Proposition 4.15 gilt $B \in \varepsilon\text{-}closure_{\mathcal{A}_{T_S}}(\{A\})$, also liefert der Algorithmus im dritten Schritt die Ausgabe „ja“.

Insgesamt zeigt dies die Vollständigkeit des Algorithmus.

Korrektheit: Wird A nicht von B S-subsumiert, so muß A S-konsistent sein. Also liefert 1.) immer die Ausgabe „nein“. Da nach Proposition 4.15 nicht $B \in \varepsilon\text{-}closure_{\mathcal{A}_{T_S}}(\{A\})$ gelten kann, gibt der Algorithmus „nein“ aus. Dies zeigt, daß jeder Berechnungspfad mit Ausgabe „nein“ terminiert.

Komplexität: Offensichtlich ist der Algorithmus ein NP-Algorithmus, da der Test $B \in \varepsilon\text{-}closure_{\mathcal{A}_{T_S}}(\{A\})$ nur polynomiale Zeit benötigt. Damit liegt die Subsumtion in NP.

Weiter gilt: A ist S-inkonsistent gdw. $A \sqsubseteq_S \perp$. Das Bottom-Konzept $\perp$ läßt sich dabei durch das (neue) Konzept B mit den Konzeptinklusionen $B \sqsubseteq P$ und $B \sqsubseteq \neg P$ für ein primitives Konzept P ausdrücken. Damit ist Inkonsistenz auf Subsumtion reduzierbar, woraus die NP-Härte der Subsumtion folgt. □

Nach [Neb90b] ist die Subsumtion für azyklische $\mathcal{FL}_0$-Terminologien — anders als für $\mathcal{SL}_{dis}$-Schemata — co-NP-vollständig. Die Subsumtion zweier Konzepte A und B wird nach (5.2) durch die Inklusionsbeziehungen der Sprachen $L_{\mathcal{A}_T}(A, P)$ und $L_{\mathcal{A}_T}(B, P)$ für primitive Konzepte P bestimmt — für eine azyklische Terminologie T sind diese Sprachen endlich. Die Komplexität der Subsumtion liegt also im Vergleich von endlichen, regulären Sprachen.

Dagegen liegt die Komplexität der Subsumtion ($A \sqsubseteq_S B$) für ein Schema S allein in der Entscheidung der Inkonsistenz von A, die für die hier betrachteten azyklischen Schemata NP-vollständig ist. Ist die Inkonsistenz entschieden, so ist gemäß (7.3) dann nur noch eine „einfache“ syntaktische Bedingung zu testen. Zudem wird das Konsistenzproblem allein durch die Anwesenheit von Zahlenrestriktionen aufwendig. Läßt man — wie auch in $\mathcal{FL}_0$ — keine Zahlenrestriktionen zu, so kann die Konsistenz und damit die Subsumtion sogar mit polynomialem Aufwand entschieden werden; in diesem Fall ist T_S nämlich eine $\mathcal{AL}_0$-Terminologie. Die besondere Gestalt von T_S macht also einen (aufwendigen) Vergleich der Sprachen $L_{\mathcal{A}_{T_S}}(A, P)$ und $L_{\mathcal{A}_{T_S}}(B, P)$ unnötig (vgl. auch Bemerkung 7.13).

Die in diesem Abschnitt vorgestellte Reduktion von Schemata auf Terminologie macht deutlich, daß Schemata als spezielle Terminologien angesehen werden können und daß sich für diese speziellen Terminologien im Vergleich zu allgemeinen Terminologien je nach Sprachumfang Komplexitätsunterschiede für Entscheidungsprobleme ergeben können.

Kapitel 8

Zusammenfassung und verwandte Arbeiten

Zyklische Terminologien können natürliche Beschreibungen terminologischen Wissens darstellen. Darüber hinaus erlauben sie bei Verwendung von Fixpunktsemantiken — anders als die Prädikatenlogik erster Stufe — Aussagen über transitive Hüllen von Relationen.

Die gfp- und die lfp-Semantik setzen — im Gegensatz zur deskriptiven Semantik — primitive Interpretationen eindeutig zu Modellen einer Terminologie fort. An Beispielen haben wir jedoch gesehen, daß keine dieser drei Semantiken in allen Fällen den anderen vorzuziehen ist.

Den Schwerpunkt dieser Arbeit bildete die Charakterisierung der drei Semantiken, der Inkonsistenz bzgl. dieser Semantiken und der Subsumtion sowie die Ableitung von Entscheidungsalgorithmen und Komplexitätsaussagen aus diesen Charakterisierungen. Dabei wurden die Ergebnisse aus [Baa96] zur Sprache $\mathcal{FL}_0$ zunächst auf die ausdrucksstärkere Sprache $\mathcal{AL}_0$ und schließlich auf $\mathcal{ALN}$ ($\mathcal{FLN}$) erweitert.

Die Charakterisierungen der Semantiken bzgl. $\mathcal{FL}_0$ ließen sich jeweils leicht um primitive Negation und Zahlenrestriktion erweitern.

Primitive Negation und Zahlenrestriktion können in allen drei Semantiken zu inkonsistenten Konzepten führen — in $\mathcal{FL}_0$ können nur bzgl. der lfp-Semantik inkonsistente Konzepte auftreten. Aus diesem Grund wurde auch die Inkonsistenz in allen drei Semantiken charakterisiert. Wir haben gesehen, daß ein Konzept A genau dann inkonsistent ist, falls die Zustandsmenge $\varepsilon\text{-}closure(\{A\})$ ein Ausschlußzustand ist. Bzgl. $\mathcal{AL}_0$ konnte damit die Inkonsistenz eines Konzeptes mit polynomialem Aufwand entschieden werden. Für $\mathcal{ALN}$ ($\mathcal{FLN}$) haben wir (N)PSPACE-Algorithmen zur Entscheidung der Inkonsistenz angegeben; der Test auf Ausschlußzustand wird hier dadurch erschwert, daß zu Individuen durch Maximum-Restriktionen Nachfolger gefordert werden können. Darüber hinaus konnten wir für alle drei Semantiken die NP-Härte der Inkonsistenz zeigen. Für schwach-azyklische und azyklische $\mathcal{ALN}$-($\mathcal{FLN}$-)Terminologien haben wir zudem die NP-Vollständigkeit der Inkonsistenz bzgl. gfp-, lfp- und deskriptiver Semantik nachgewiesen.

Aufgrund von primitiver Negation und Zahlenrestriktion können Konzepte von Wörtern ausgeschlossen werden. Deshalb ist bei der Charakterisierung der Subsumtion für alle drei Semantiken jeweils die Berücksichtigung von Ausschlußwörtern nötig. Bzgl. $\mathcal{AL}_0$ konnten wir die regulären bzw. ω-regulären Sprachen E_A bzw. $E^f_{A,\omega}$ von Ausschlußwörtern zu einem Konzept A durch in der Größe der gegebenen Terminologie polynomiale Semi-

Automaten beschreiben. Dadurch war es möglich, das Subsumtionsproblem polynomial auf das Inklusionsproblem für reguläre und ω-reguläre Sprachen zu reduzieren. Daneben haben wir E_A und $E^f_{A,\omega}$ durch Ausschlußzustände charakterisiert und mit Hilfe dieser Charakterisierung (N)PSPACE-Entscheidungsalgorithmen zur Subsumtion formuliert. Dieser Ansatz konnte mit wenig Aufwand auch auf die Sprache $\mathcal{ALN}$ ($\mathcal{FLN}$) übertragen werden. Dazu war eine Anpassung des Begriffs „Ausschlußzustand“ nötig, da nun widersprüchliche Zahlenrestriktionen und von einem Konzept geforderte Wörter berücksichtigt werden mußten. Mit der in [Baa96] bewiesenen PSPACE-Vollständigkeit der Subsumtion bzgl. der gfp- und der lfp-Semantik in $\mathcal{FL}_0$ konnten wir die PSPACE-Vollständigkeit auch bzgl. der Sprachen $\mathcal{AL}_0$ und $\mathcal{ALN}$ ($\mathcal{FLN}$) nachweisen.

Die Ergebnisse zu (zyklischen) Terminologien wurden dazu benutzt, Konsistenz, (lokale) Gültigkeit und Subsumtion bzgl. $\mathcal{ALN}$- ($\mathcal{SLN}$-, $\mathcal{SL}_{dis}$-)Schemata zu entscheiden. Dazu wurden diese Probleme auf Konsistenz und Subsumtion in $\mathcal{ALN}$-Terminologien reduziert, wodurch wir die Existenz von PSPACE-Algorithmen für diese Schemaprobleme zeigen konnten. Weiterhin stellte sich heraus, daß der in [BDNS97] vorgestellte graphbasierte Algorithmus zur Entscheidung der Inkonsistenz bzgl. $\mathcal{SL}_{dis}$-Schemata ähnlich arbeitete, wie der hier vorgestellte Entscheidungsalgorithmus für Ausschlußzustände. Für Schemata (S) — wie auch für die zugehörigen speziellen Terminologien (T_S) — lag die Komplexität des Subsumtionsproblems, anders als für beliebige Terminologien, allein in der Entscheidung der Konsistenz von Konzepten. Für azyklische Schemata bzw. Terminologien ergeben sich sogar Komplexitätsunterschiede. So ist die Subsumtion für azyklische $\mathcal{SL}_{dis}$-Schemata NP-vollständig, wohingegen sie für azyklische $\mathcal{FL}_0$-Terminologien nach [Neb90b] co-NP-vollständig ist. Für $\mathcal{SL}_{dis}$-Schemata ohne Zahlenrestriktionen ist die Subsumtion sogar mit polynomialem Aufwand entscheidbar.

Die Charakterisierung der Semantik zyklischer Terminologien mit endlichen Automaten kann bei der Entscheidung helfen, welche Semantik in einer konkreten Situation die geeignetere ist. Für die Charakterisierung selbst stellte sich in dieser Arbeit — wie auch in [Baa96] — heraus, daß die gfp-Semantik im Vergleich zu den anderen Semantiken eine natürlichere Beschreibung durch endliche Automaten zuläßt. In [BDNS97] wird die Wahl der Semantik, wie angesprochen, dadurch gelöst, daß eine Terminologie in Schema und Viewdefinition geteilt wird. Für das Schema wird die deskriptive Semantik gewählt, da nur die möglichen Modelle für die primitiv definierten Konzepte und Rollen eingeschränkt werden sollen. Für die Viewdefinition wird dagegen eine Fixpunktsemantik (meist die gfp-Semantik) zugrunde gelegt, da hier die Konzepte eindeutig durch Konzepte und Rollen des Schemas definiert werden sollen. Schließlich kann man durch Beschränkung auf azyklische Terminologien die Entscheidung für eine der drei Semantiken völlig umgehen, da die drei Semantiken für derartige Terminologien übereinstimmen. Um keine Ausdrucksstärke zu verlieren, werden dazu in [Baa91] die Rollenkonstruktoren Vereinigung, Komposition und transitiver Abschluß eingeführt. Diese erlauben, wie zyklische Terminologien, Aussagen in Abhängigkeit von regulären Sprachen zu formulieren, insbesondere also die Darstellung von Aussagen über transitive Hüllen von Relationen.

Sprachindex

Literaturverzeichnis

[AU79] A.V. Aho und J.D. Ullman: Universality of data retrieval languages. In: *Proceedings of the 6th ACM Symposium on Principles of Programming Languages*, 1979.

[Baa91] 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.

[Baa96] F. Baader: Using automata theory for characterizing the semantics of terminological cycles. In: *Annals of Mathematics and Artificial Intelligence*, Band 18, Seiten 175–219. J.C. Baltzer AG, Science Publishers, 1996.

[BDNS97] M. Buchheit, F.M. Donini, W. Nutt und A. Schaerf: Refining the structure of terminological systems: Terminology = Schema + View. Konferenzversion: *Proceedings of the 12th National Conference of the American Association for Artificial Intelligence*, AAAI-94, Seattle, 1994. Die ausführliche Version, auf die sich auch diese Arbeit bezieht, erscheint in *Artificial Intelligence*, 1997.

[BH91] F. Baader und B. Hollunder: KRIS: Knowledge representation and inference system. In: *SIGART Bulletin*, Band 2, 1991.

[BPGL85] R.J. Brachman, V. Pigman-Gilvert und H.J. Levesque: An essential hybrid reasonig system: Knowledge and symbol level accounts in KRYPTON. In: *Proceedings of the 9th International Joint Conference on Artificial Intelligence*, Los Angeles, 1985.

[DHL+92] F. M. Donini, B. Hollunder, M. Lenzerini, A. Marchetti, D. Nardi und W. Nutt: The complexity of existential quantification in concept languages. In: *Artificial Intelligence*, Band 2–3, Seiten 309–327, 1992.

[Eil74] S. Eilenberg: *Automata, Languages and Machines*, Band A. Academic Press, New York/London, 1974.

[GJ79] M.R. Garey und D.S. Johnson: *Computers and Intractability: A Guide to the Theory of NP-Completeness.* Freeman, San Francisco, 1979.

[HU69] J.E. Hopcroft und J.D. Ullman: *Formal Languages and their Relation to Automata.* Addison-Wesley Publ. Co., Massachussetts, 1969.

[HU79] J.E. Hopcroft und J.D. Ullman: *Introduction to Automata Theory.* Addison-Wesley, 1979.

[Kö35] D. König: *Theorie der endlichen und unendlichen Graphen.* Chelsea Publishing Company, New York, 1935.

[LB87] H.J. Levesque und H.J. Brachman: Expressiveness and tractability in knowledge representation and reasonig. In: *Computational Intelligence*, Band 3, 1987.

[Llo87] J.W. Lloyd: *Logic Programming*, Abschnitt 1.5. Springer Verlag, Berlin, zweite erweiterte Auflage, 1987.

[MB87] R. MacGregor und R. Bates: The Loom knowledge representation Language. Technical Report ISI/RS-87-188, University of Southern California, Information Science Institute, 1987.

[Neb87] B. Nebel: On Terminological cycles. In: *KIT Report 58*, Technische Universität Berlin, 1987.

[Neb90a] B. Nebel: Reasoning and Revision in Hybrid Representation Systems. In: *Lecture Notes in Computer Science 422.* Springer Verlag, Berlin, 1990.

[Neb90b] B. Nebel: Terminological reasoning is inherently intractable. In: *Artificial Intelligence*, Band 43, Seiten 235–249, 1990.

[Neb91] B. Nebel: Terminological cycles: Semantics and computational properties. In: J. Sowa (Herausgeber): *Formal Aspects of Semantic Networks*, Seiten 331–361. Morgan Kaufmann, San Mateo, 1991.

[PSMB+91] P. Patel-Schneider, D.L. MacGuinnes, R.J. Brachman, L. Alperin Resnick und A. Borgida: The CLASSIC knowledge representation system: Guiding principles and implementation issues. In: *SIGART Bulletin*, Band 2, 1991.

[Ros82] J.G. Rosenstein: *Linear Ordering*, Kapitel 1 und 3. Academic Press, New York, 1982.

[SVW87] A.P. Sistla, M.Y. Vardi und P. Wolper: The complementation problem for Büchi automata with applications to temporal logic. In: *Theoretical Computer Science*, Band 49, 1987.

[Tho90] W. Thomas: Automata on infinite objects. In: J. van Leeuwen (Herausgeber): *Handbook of Theoretical Computer Science*, Band B. Elsevier Science Publishers, Amsterdam, 1990.