Einige Aspekte der Modallogik $\mathbf{S5}_n$ mit Allgemeinwissen

Diplomarbeit

der Philosophisch-naturwissenschaftlichen Fakultät

der Universität Bern

vorgelegt von

Michel Krebs

2000

Leiter der Arbeit:

Prof. Dr. Gerhard Jäger,

Institut für Informatik und angewandte Mathematik

Epilog

Nach der Metaphysik des Aristoteles streben alle Menschen von Natur aus nach Wissen. Zwar ist die genaue Bedeutung des Angestrebten schwer fassbar, doch lässt sich im Alltag mit einer intuitiven Vorstellung davon, was Wissen überhaupt ist, gut leben. Die Aufgabe aber, etwa einer künstlichen Lebensform beizubringen, Informationen über ihre Umwelt zu sammeln um daraufhin bestimmte Tätigkeiten auszuführen, verlangt eine präzise Formulierung der für diese Handlung relevanten Eigenschaften von Wissen. So finden sich in jüngster Zeit im Bereich der theoretischen Informatik namhafte Untersuchungen, die sich der Herausforderung dieser Aufgabe stellen. Die vorliegende Arbeit beschäftigt sich mit einer winzigen Auswahl dieser Palette: Wissen wird als wahrer, gerechtfertigter Glaube mit den zusätzlichen Eigenschaften der positiven und negativen Introspektion betrachtet. Zur Adäquatheit dieser Definition werde ich mich nicht äussern. Jene Diskussion wird an andern Fakultäten kompetenter geführt als ich dies ausführen könnte.

Im Kapitel 1 stelle ich einen formalen Rahmen zur Darstellung von Wissen vor, der im 2. Kapitel um den Begriff des Allgemeinwissens erweitert wird. Mein Augenmerk liegt auf den komplexitätstheoretischen Eigenschaften der präsentierten Systeme, daher sehe ich mich gezwungen, in einem weiterm Kapitel die verwendeten Begriffe jener Theorie einzuführen. Für einen klassichen Fünfakter etwas spät, folgt im Kapitel 4 mit dem Algorithmus zur Erfüllbarkeit der Logik $S5_n$ wohl der Höhepunkt dieser Arbeit. Die im letzten Kapitel angesprochenen Fragen zur Logik mit Allgemeinwissen sind zwar sehr interessant, doch leider auch äusserst schwierig. Ich kann sie nicht in derselben Ausführlichkeit besprechen wie mir dies zuvor im Falle der Logik $S5_n$ möglich war. Vielleicht findet sich ein kluger Kopf, der sich ihrer annimmt.

Trotz der Bescheidenheit dieser Arbeit möchte ich an dieser Stelle allen Personen meinen aufrichtigen Dank aussprechen, die mir beim Verfassen dieser Arbeit geholfen oder für erholende Zerstreunung gesorgt haben. Zuvorderst gilt mein Dank Prof. Dr. Gerhard Jäger für die kompetente und gedul-

dige Betreuung. Die enorme Freiheit, die er mir bei dieser Arbeit gewährte, hob sich wohltuend von der grassierenden (Ver-)Planwirtschaft der Universitätsgremien ab. Bei Thomas Strahm bedanke ich mich für sein stets offenes Ohr, selbst wenn dies seine eigenen Arbeiten auf die lange Bank schob. Zu grossem Dank verpflichtet bin ich Luca Alberucci. Ohne seine stetige Bereitschaft zu erhellenden und konstruktiven Diskussionen wäre ich wohl nie auf einen grünen Zweig gekommen; und natürlich bedanke ich mich für all die Glacés, die ich auf seine Kosten genoss. Meiner Freundin Michèle Müller verdanke ich die Korrektur von Stilblüten und dreckfuhlern. Mein Grübeln selbst beim romantischen Tête-à-tête im Kerzenschein verlangten ihrer Gutmütigkeit einiges ab. Schliesslich danke ich meinen Eltern, ohne deren mentale und finanzielle Rückendeckung meine Ausbildung an der Universität nicht möglich gewesen wäre.

Bekanntlich gibt es immer wieder Personen jeglicher Couleur, welche die Bedeutung mathematischen Treibens in Zweifel ziehen, die ketzerische Frage nach dessen Sinn aufwerfen und die Menschheit von dieser Plage erlösen möchten. Ihnen allen widme ich die nebenstehenden Zeilen aus Goethes Faust.

Gebraucht der Zeit, sie geht so schnell von hinnen!
Doch Ordnung lehrt Euch Zeit gewinnen.
Mein teurer Freund, ich rat Euch drum
Zuerst Collegium Logicum.
Da wird der Geist Euch wohl dressiert,
In Spanische Stiefeln eingeschnürt,
Dass er bedächtiger so fortan
Hinschleiche die Gedankenbahn
Und nicht etwa, die Kreuz und Quer,
Irrlichteliere hin und her.
Dann lehrt man Euch manchen Tag,
Dass, was Ihr sonst auf einen Schlag
Getrieben, wie Essen und Trinken frei,
Eins! Zwei! Drei! dazu nötig sei.

Goethe, Faust I

Inhaltsverzeichnis

Epilog **i**

1 Die Logik S5$_n$ **1**
1.1 Motivation . . . 1
1.2 Die Syntax . . . 2
1.2.1 Das Alphabet . . . 2
1.2.2 Die Formeln . . . 2
1.2.3 Einige syntaktische Begriffe . . . 3
1.3 Die Semantik . . . 4
1.3.1 Kripke-Strukturen . . . 4
1.3.2 Gültigkeit und Erfüllbarkeit . . . 5
1.4 Axiomensysteme . . . 5
1.5 Korrektheit . . . 7
1.5.1 Eingeschränkte Kripke-Strukturen . . . 8
1.5.2 Der Korrektheitsbeweis . . . 8
1.6 Vollständigkeit . . . 10
1.6.1 Der Vollständigkeitsbeweis . . . 10
1.6.2 Äquivalente Modelle . . . 13
1.7 Die Entscheidbarkeit von S5$_n$ . . . 15
1.7.1 Der allgemeine Fall . . . 16
1.7.2 Der Spezialfall S5 . . . 20

2 Allgemeinwissen **23**
2.1 Ein Beispiel . . . 23
2.2 Die Logik S5$_n^C$ . . . 25
2.2.1 S5$_n^C$ als Erweiterung der Sprache S5$_n$ . . . 25
2.2.2 Axiomatisiertes Allgemeinwissen . . . 28
2.2.3 Die Vollständigkeit von S5$_n^C$ . . . 29

3 Komplexitätstheoretisches Rüstzeug **37**
3.1 Komplexitätstheoretische Begriffe 37
3.2 Beispiele von vollständigen Problemen 39
3.2.1 NP-Vollständigkeit 39
3.2.2 PSPACE-Vollständigkeit 39
3.2.3 EXPTIME-Vollständigkeit 40

4 Die Komplexität von $\mathbf{S5}_n$ **49**
4.1 Der Spezialfall S5 49
4.2 Der allgemeine Fall $S5_n$ 50
4.2.1 Gegenbeispiel Nr.1 50
4.2.2 $SATS5_n$ ist PSPACE-hart 54
4.2.3 PSPACE-Entscheidungsverfahren für $S5_n$ 58
4.3 Low cost $S5_n$ 70
4.3.1 Beschränkte Verschachtelungstiefe 70

5 Die Komplexität von $\mathbf{S5}_n^C$ **73**
5.1 Gegenbeispiel Nr.2 73
5.2 Die Sonderrolle von $S5^C$ 78
5.3 Untere EXPTIME Schranke für $S5_2^C$ 79
5.4 Obere EXPTIME Schranke für $S5_n^C$ 82
5.5 Konsequenzen 84

Literaturverzeichnis **87**

Kapitel 1

Die Logik $S5_n$

1.1 Motivation

Wissen soll mit Hilfe von Modallogik formalisiert werden. Ein schwieriges Unterfangen, streiten sich doch Philosophen seit 2000 Jahren darüber, welche Eigenschaften unter den Begriff Wissen fallen. Sogesehen müsste das Vorhaben zwangsläufig scheitern. Der Logiker ist nun aber nicht an einer universellen Definition von Wissen interessiert. Er fasst die diversen Attribute des Wissensbegriffs zu unterschiedlichen Systemen zusammen und überlässt es dem Modellierer, die zu einer vorliegenden Anwendung passende Kollektion auszuwählen.

Es werden im folgenden einige bekannte Wissenslogiken vorgestellt. Das Hauptinteresse dieser Arbeit liegt zwar auf den Systemen $S5_n$ und $S5_n^C$, doch präsentiere ich auch die Sprachen K_n, T_n, $S4_n$, resp. K_n^C, T_n^C und $S4_n^C$. Sie werden es erlauben, die späteren Beweise modulartig aufzubauen, in der Hoffnung, dies möge der Leserlichkeit dienen.

Der dem Gebiet unvertraute Leser bediene sich folgender Intuition: Eine endliche Zahl von Agenten, kreativerweise mit 1 ... n bezeichnet, räsoniere anhand gewisser „brute facts“ , zusammengefasst in der Menge Φ, über die Welt. Jedem Agenten wird ein modaler Operator K_i zugewiesen. Ausdrücke der Form $K_i\varphi$ sind dann als „Agent i weiss φ“ zu lesen. Der Rest dieses Kapitels dient der Formalisierung dieser Intuition.

Für eine ausführlichere Einführung sei auf [FHMV95] verwiesen.

1.2 Die Syntax

1.2.1 Das Alphabet

Definition 1.2.1. Das Alphabet der Sprachen K_n, T_n, $S4_n$, $S5_n$ besteht aus folgenden Grundzeichen:

a) Eine endliche Menge Φ von primitiven Propositionen

b) Logischen Symbolen $\neg, \wedge, \vee$

c) Modalen Operatoren $K_1, ..., K_n$ für $n \geq 1$

d) Hilfszeichen (,)

1.2.2 Die Formeln

Definition 1.2.2. Die Formeln der Sprachen K_n, T_n, $S4_n$, $S5_n$ werden durch endlichmalige Anwendung der folgenden Regeln erhalten:

a) Jedes Element $p \in \Phi$ ist eine Formel.

b) Ist φ eine Formel, dann ist auch $\neg\varphi$ eine Formel.

c) Sind φ, ψ Formeln, dann ist auch $(\varphi \wedge \psi)$ eine Formel.

d) Ist φ eine Formel, dann ist auch $K_i\varphi$, für jedes $i \in \{1, \ldots, n\}$, eine Formel.

Die Menge aller Formeln wird mit $\mathcal{L}_n(\Phi)$ bezeichnet.

Definition 1.2.3 (Abkürzungen in der Objektsprachen). In den Objektsprachen werden die üblichen Standardabkürzungen eingeführt:

a) $(\varphi \vee \psi)$ steht für $\neg(\neg\varphi \wedge \neg\psi)$

b) $(\varphi \Rightarrow \psi)$ steht für $(\neg\varphi \vee \psi)$

c) $(\varphi \Leftrightarrow \psi)$ steht für $(\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi)$

d) Es gilt Linksklammerung: $(\varphi \wedge \psi \wedge \chi)$ steht für $((\varphi \wedge \psi) \wedge \chi)$

e) Klammern werden weggelassen, wenn dadurch keine Unklarheiten entstehen.

f) **true** steht für eine beliebige aussagenlogische Tautologie

g) **false** steht für $\neg$**true**

1.2.3 Einige syntaktische Begriffe

Definition 1.2.4 (size). Die Grösse (engl. size) einer Formel φ, bezeichnet durch $|\varphi|$, ist ihre Länge über dem Alphabet $\Phi \cup \{\neg, \wedge, (,), K_1, \ldots, K_n\}$.

Definition 1.2.5 (depth). Die modale Tiefe (engl. depth) einer Formel φ, bezeichnet durch $dep(\varphi)$, wird induktiv definiert durch:

a) $dep(p) = 0$ für alle $p \in \Phi$

b) $dep(\neg\varphi) = dep(\varphi)$

c) $dep(\varphi \wedge \psi) = \max(dep(\varphi), dep(\psi))$

d) $dep(K_i\varphi) = dep(\varphi) + 1$, für alle $i \in \{1, \ldots, n\}$

Diese Definition lässt sich problemlos auf endliche Formelmengen übertragen:

Definition 1.2.6. Die modale Tiefe einer endlichen Formelmenge T wird durch die modale Tiefe ihrer Elemente definiert, formal:

$$dep(T) \doteq \max\{dep(\varphi) : \varphi \in T\}$$

Definition 1.2.7 (Subformel). Die Menge $\mathrm{Sub}(\varphi)$ der Subformeln einer Formel φ wird induktiv definiert durch:

a) $Sub(p) = \{p\}$ für alle $p \in \Phi$

b) $Sub(\neg\varphi) = \{\neg\varphi\} \cup Sub(\varphi)$

c) $Sub(\varphi \wedge \psi) = \{\varphi \wedge \psi\} \cup Sub(\varphi) \cup Sub(\psi)$

d) $Sub(K_i\varphi) = \{K_i\varphi\} \cup Sub(\varphi)$, für alle $i \in \{1, \ldots, n\}$

Eine Formel ψ wird als Subformel von φ bezeichnet, falls ψ ein Element von $Sub(\varphi)$ ist.

Lemma 1.2.8. *Für alle $\varphi \in \mathcal{L}_n(\Phi)$ gilt: $|\,Sub(\varphi)| \leq |\varphi|$.*

Beweis. Induktion über den Aufbau von φ.

a) $\varphi \equiv p : |\,Sub(p)| = \{p\} = |p| = 1$

b) $\varphi \equiv \neg\psi : |\,Sub(\varphi)| = |\,Sub(\neg\psi)| = 1 + |\,Sub(\psi)| \overset{\text{I.V.}}{\leq} |\varphi|$

c) $\varphi \equiv \varphi' \wedge \varphi'' : |\,Sub(\varphi)| = |\,Sub(\varphi' \wedge \varphi'')| = |\{\varphi\} \cup Sub(\varphi') \cup Sub(\varphi'')| \leq 1 + |\,Sub(\varphi')| + |\,Sub(\varphi'')| \overset{\text{I.V.}}{\leq} 1 + |\varphi'| + |\varphi''| \leq |\varphi|$

d) $\varphi \equiv K_i\psi : |\,Sub(\varphi)| = |\,Sub(K_i\psi)| = 1 + |\,Sub(\psi)| \overset{\text{I.V.}}{\leq} |\varphi|$

q.e.d.

1.3 Die Semantik

1.3.1 Kripke-Strukturen

Eine Formel der Art $K_i\varphi$ soll als „Agent i weiss φ“ interpretiert werden. Die *possible world semantic* liefert eine Möglichkeit, diese Intuition zu realisieren: Ein Agent i weiss die Aussage φ genau dann, wenn φ in allen für ihn anhand seines Wissens nicht unterscheidbaren Welten gültig ist. Kripke-Strukturen [Kri63] stellen diese Idee in einen formalen Rahmen.

Definition 1.3.1 (Kripke-Struktur). Eine Kripke-Struktur für n Agenten ist ein Tripel $M = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ mit:

a) Einer Menge $S \neq \emptyset$ von Zuständen oder möglichen Welten.

b) Einer Funktion $\pi : S \times \Phi \to \{\textbf{true, false}\}$, welche jeder Welt $s \in S$ eine Wahrheitfunktion auf Φ zuordnet.

c) $\mathcal{K}_i \subseteq S \times S$, für jedes $i \in \{1, \ldots, n\}$.

Die Gültigkeitsrelation stellt nun eine Verbindung zwischen den Formeln aus $\mathcal{L}_n(\Phi)$ und Kripke-Strukturen her.

Definition 1.3.2. Es sei M eine Kripke-Struktur, s eine Welt aus M und φ eine Formel aus $\mathcal{L}_n(\Phi)$. Die Relation $(M, s) \models \varphi$ wird induktiv über den Aufbau von φ definiert:

a) $(M, s) \models p$ gdw $\pi_s(p) = \textbf{true}$, für alle $p \in \Phi$.

b) $(M, s) \models \varphi \wedge \psi$ gdw $(M, s) \models \varphi$ und $(M, s) \models \psi$.

c) $(M, s) \models \neg\varphi$ gdw $(M, s) \not\models \varphi$.

d) $(M, s) \models K_i\varphi$ gdw $(M, t) \models \varphi$ für alle t mit $(s, t) \in \mathcal{K}_i$.

$(M, s) \models \varphi$ wird als „φ ist wahr in (M, s)“ oder „(M, s) erfüllt φ“ gelesen.

Kripke-Strukturen bieten den Vorteil, sich als gerichteter, angeschriebener Graph darstellen zu lassen. Die Welten aus S werden als Knoten interpretiert. Zwischen zwei Knoten s und t liegt genau dann eine mit i bezeichnete Kante, wenn $(s, t) \in \mathcal{K}_i$. Diese graphentheoretische Interpretation wird sich in den späteren Untersuchungen als äusserst nützlich erweisen.

1.3.2 Gültigkeit und Erfüllbarkeit

Definition 1.3.3 (Erfüllbarkeit). Eine Formel φ ist erfüllbar in M, falls $(M, s) \models \varphi$ für ein $s \in S$.

Eine Formel φ ist erfüllbar bezüglich einer Klasse $\mathcal{M}$ von Strukturen, falls φ in einem $M \in \mathcal{M}$ erfüllbar ist.

Definition 1.3.4 (Gültigkeit). Eine Formel φ ist gültig in M, falls $(M, s) \models \varphi$ für alle $s \in S$. „φ ist gültig in M“ wird als $M \models \varphi$ geschrieben.

Eine Formel φ ist gültig bezüglich einer Klasse $\mathcal{M}$ von Strukturen, geschrieben $\mathcal{M} \models \varphi$, falls $M \models \varphi$ für alle $M \in \mathcal{M}$.

Das folgende, leicht zu überprüfende Korollar verbindet die Begriffe Gültigkeit und Erfüllbarkeit.

Korollar 1.3.5. *Eine Formel φ ist genau dann erfüllbar, wenn $\neg\varphi$ nicht gültig ist.*

1.4 Axiomensysteme

Die diversen Eigenschaften des Wissensbegriffs werden nun durch unterschiedliche Axiomensysteme eingefangen (vgl.[HM92]). Folgende Axiome stehen zur Auswahl, wobei wie immer $1 \leq i \leq n$ gilt [1]:

- **P.** Alle Instanzen aussagenlogischer Axiome
- **K.** $(K_i\varphi \wedge K_i(\varphi \Rightarrow \psi)) \Rightarrow K_i\psi$
- **T.** $K_i\varphi \Rightarrow \varphi$
- **4.** $K_i\varphi \Rightarrow K_iK_i\varphi$
- **5.** $\neg K_i\varphi \Rightarrow K_i\neg K_i\varphi$

Schlussregeln:

- **R1.** Von $\vdash \varphi$ und $\vdash \varphi \Rightarrow \psi$ schliesse auf $\vdash \psi$ (Modus Ponens)
- **R2.** Von $\vdash \varphi$ schliesse auf $\vdash K_i\varphi$ (Notwendigkeit)

Diese Axiome können nun auf veschiedene Arten miteinander kombiniert werden. Dieses Zusammenfassen der Axiome führt zur nächsten Definition:

Definition 1.4.1 (Axiomensystem). Ein Axiomensystem $\mathcal{S}$ besteht aus einer Kollektion von Axiomenschemata und Schlussregeln.

[1] Strenggenommen handelt es sich hierbei nicht um Axiome, sondern um Axiomenschemata. Ich bitte die Leserschaft für diesen schlampigen, jedoch praktischen Wortgebrauch um Verzeihung.

Die für diese Arbeit interessanten Axiomensysteme sind in der Tabelle zusammengefasst:

Name	**Axiome**	**Schlussregeln**
K_n	**P, K**	**R1, R2**
T_n	**P, K, T**	**R1, R2**
$S4_n$	**P, K, T, 4**	**R1, R2**
$S5_n$	**P, K, T, 4, 5**	**R1, R2**

Definition 1.4.2 (Beweisbarkeit). Bezeichne $\mathcal{S}$ ein Axiomensystem aus $\{K_n, T_n, S4_n, S5_n\}$. Eine Formel φ ist beweisbar aus $\mathcal{S}$, in Symbolen $\mathcal{S} \vdash \varphi$, falls eine Folge $\varphi_1, \ldots, \varphi_n$ von Formel mit $\varphi_n = \varphi$ so existiert, dass für jedes φ_j mit $1 \leq j \leq n$ eine der folgenden Bedingungen zutrifft:

a) φ_j ist Instanz eines Axioms aus $\mathcal{S}$.

b) φ_j ist durch Anwendung einer Schlussregel auf Folgenglieder mit kleinerem Index entstanden.

Jede Formel φ_j nenne ich eine Zeile dieses Beweises.

Definition 1.4.3 (Konsistenz).

a) Eine Formel φ ist $\mathcal{S}$-konsistent falls $\mathcal{S} \nvdash \neg\varphi$.

b) Eine endliche Menge $\{\varphi_1, \ldots, \varphi_k\}$ von Formeln aus $\mathcal{L}_n(\Phi)$ ist genau dann $\mathcal{S}$-konsistent, wenn $\varphi_1 \wedge \ldots \wedge \varphi_k$ $\mathcal{S}$-konsistent ist.

c) Eine unendliche Menge Σ von Formeln aus $\mathcal{L}_n(\Phi)$ ist genau dann $\mathcal{S}$-konsistent, wenn jede endliche Teilmenge Σ_0 von Σ $\mathcal{S}$-konsistent ist.

d) Eine Formel φ oder eine Menge Σ von Formeln ist genau dann $\mathcal{S}$-inkonsistent, wenn sie nicht $\mathcal{S}$-konsistent ist.

e) Eine Formelmenge Σ heisst genau dann maximal $\mathcal{S}$-konsistent, wenn jede echte Erweiterung Σ' von Σ $\mathcal{S}$-inkonsistent ist. Mit andern Worten: Σ ist maximal $\mathcal{S}$-konsistent gdw Σ ist $\mathcal{S}$-konsistent und für alle $\varphi \notin \Sigma$ ist $\Sigma \cup \{\varphi\}$ $\mathcal{S}$-inkonsistent.

Nachfolgendes Lemma wird sich auf dem Weg zum Vollständigkeitsbeweis der Axiomensysteme $K_n, T_n, S4_n, S5_n$ als nützlich erweisen:

Lemma 1.4.4 (Lindenbaum). *In einem beliebigen Axiomensystem $\mathcal{S}$, das **P** enthält und mit der Schlussregel **R1** arbeitet, kann jede konsistente Formelenge Σ zu einer maximal konsistenten Formelmenge Σ' erweitert werden. Weiter gelten, falls Σ eine maximal konsistente Formelmenge ist, folgende Eigenschaften:*

a) Für jede Formel φ gilt entweder $\varphi \in \Sigma$ oder $\neg\varphi \in \Sigma$.

b) $\varphi \wedge \psi \in \Sigma$ gdw $\varphi \in \Sigma$ und $\psi \in \Sigma$.

c) Falls φ und $\varphi \Rightarrow \psi \in \Sigma$, dann ist auch $\psi \in \Sigma$.

d) Falls $\mathcal{S} \vdash \varphi$, dann folgt $\varphi \in \Sigma$.

Beweis. Sei Σ eine konsistente Menge und $\{\varphi_i |\ i \in \mathbf{N}\}$ eine Aufzählung der Formeln in $\mathcal{L}_n(\Phi)$. Setze

$$\Sigma_0 \doteq \Sigma$$

$$\Sigma_{n+1} = \begin{cases} \Sigma_n \cup \{\varphi_n\} & : \text{ falls } \Sigma_n \cup \{\varphi_n\} \text{ konsistent} \\ \Sigma_n & : \text{ sonst} \end{cases}$$

$$\bar{\Sigma} = \bigcup_{n=0}^{\infty} \Sigma_n$$

$\bar{\Sigma}$ ist sicher konsistent. Wäre dies nicht der Fall, so wäre bereits ein Σ_n inkonsistent, im Widerspruch zur Konstruktion. Diese garantiert auch die Maximalität: Gäbe es ein φ_j, so dass $\varphi_j \notin \bar{\Sigma}$ und $\bar{\Sigma} \cup \{\varphi_j\}$ konsistent wäre, so würde auch $\varphi_j \notin \Sigma_n$ und $\Sigma_n \cup \{\varphi_j\}$ konsistent gelten, für alle n. Daraus folgt aber auch die Konsistenz von $\Sigma_j \cup \{\varphi_j\}$, und somit $\Sigma_j \cup \{\varphi_j\} \subseteq \bar{\Sigma}$, im Widerspruch zur Annahme.

Sei Σ im weiteren eine maximal konsistente Formelmenge. Sei zudem φ eine beliebige Formel aus $\mathcal{L}_n(\Phi)$. Annahme: φ und $\neg\varphi$ seien nicht in Σ. Dann wären $\Sigma \cup \{\varphi\}$ und $\Sigma \cup \{\neg\varphi\}$ inkonsistent, aufgrund der Maximalität von Σ. Somit wäre auch $\Sigma \cup \{\varphi \vee \neg\varphi\}$ inkonsistent. Da aber $\varphi \vee \neg\varphi$ eine aussagenlogische Tautologie darstellt, müsste daher Σ inkonsistent sein, ein Widerspruch.

Die weiteren Eigenschaften folgen nun leicht. Ist $\varphi \wedge \psi \in \Sigma$, so müssen auch φ resp. ψ in Σ liegen, ansonsten liegt $\neg\varphi$ oder $\neg\psi$ in Σ und Σ ist inkonsistent.

Sind φ und ψ in Σ, so folgt mit aussagenlogischem Schliessen sofort $\neg(\varphi \wedge \psi) \notin \Sigma$ und somit $\varphi \wedge \psi \in \Sigma$.

Liegen andererseits φ und $\varphi \Rightarrow \psi$ in Σ, so ist ψ beweisbar, womit $\neg\psi \notin \Sigma$, da Σ sonst inkonsistent wäre.

Eigenschaft (d) ist offensichtlich. Gilt $\varphi \notin \Sigma$, so folgt mit (a) sofort $\neg\varphi \in \Sigma$. Da zudem $\neg(\neg\varphi) \equiv \varphi$ gilt, wäre Σ wiederum inkonsistent. q.e.d.

1.5 Korrektheit

Die Korrektheit der in 1.4 eingeführten Axiomensysteme soll nun in diesem Abschnitt gezeigt werden.

1.5.1 Eingeschränkte Kripke-Strukturen

Als Modelle der vorgestellten Axiomensysteme werden, wie in 1.3.1 auf Seite 4 erwähnt, Kripke-Strukturen verwendet. Die unterschiedliche Ausdrucksstärke der Systeme spiegelt sich dabei in zusätzlichen Bedingungen an die binären Relationen $\mathcal{K}_i$. Die nachfolgende Tabelle fasst die Ergebnisse aus [Kri63] zusammen.

Axiomensystem	**Eigenschaften der $\mathcal{K}_i$'s**
K_n	beliebig
T_n	reflexiv
$S4_n$	reflexiv, transitiv
$S5_n$	reflexiv, transitiv, symmetrisch

Bemerkung 1.5.1. Im folgenden bezeichne $\mathcal{M}_n^r$ die Klasse der KripkeStrukturen für n Agenten und reflexiven Möglichkeitsrelationen $\mathcal{K}_i$. Entsprechend bezeichne $\mathcal{M}_n^{rt}$ die Klasse der Kripke-Strukturen für n Agenten und Möglichkeitsrelationen $\mathcal{K}_i$, die reflexiv und transitiv sind; schliesslich bezeichne $\mathcal{M}_n^{rst}$ die Klasse der Kripke-Strukturen für n Agenten mit reflexiven, symmetrischen und transitiven Möglichkeitsrelationen $\mathcal{K}_i$. Die $\mathcal{K}_i$ sind in diesem Falle somit Äquivalenzrelationen.

1.5.2 Der Korrektheitsbeweis

Ausgestattet mit den elementarsten Grundlagen kann nun die Korrektheitkeit der Logiken K_n, T_n, $S4_n$ und $S5_n$ bewiesen werden:

Theorem 1.5.2.

a) K_n ist ein korrektes Axiomensystem bezüglich $\mathcal{M}_n$.

b) T_n ist ein korrektes Axiomensystem bezüglich $\mathcal{M}_n^r$.

c) $S4_n$ ist ein korrektes Axiomensystem bezüglich $\mathcal{M}_n^{rt}$.

d) $S5_n$ ist ein korrektes Axiomensystem bezüglich $\mathcal{M}_n^{rst}$.

Beweis. Im Brennpunkt dieser Arbeit stehen die Logiken $S5_n$ und $S5_n^C$ (letztere wird später eingeführt). Ich führe den Korrektheits- und später auch den Vollständigkeitsbeweis daher nur für das System $S5_n$. Die Beweise für die andern Logiken lassen sich durch einfache Modifikationen des Nachfolgenden erreichen. Im folgenden gelte immer $\varphi, \psi, \psi' \in \mathcal{L}_n(\Phi), M \in \mathcal{M}^{rst}, s \in S$ und $i \in \{1, \ldots, n\}$.

Die Korrektheit behauptet: Gilt $S5_n \vdash \varphi$, dann auch $\mathcal{M}_n^{rst} \models \varphi$. Der zugehörige Beweis erfolgt induktiv über die Beweislänge n.

- n=1:
  - φ ist eine aussagenlogische Tautologie. φ ist daher wahr unter allen aussagenlogischen Belegungen. Da die Interpretation der Zeichen $\neg$ und $\wedge$ durch die Gültigkeitsrelation $\models$ genau gleich geschieht wie in der Aussagenlogik, ist φ auch unter allen Belegungen $\pi_s : \Phi \to \{\textbf{true, false}\}$ wahr. Somit gilt $\mathcal{M}_n^{rst} \models \varphi$.
  - φ ist eine Instanz eines Axiomes.
    - $\varphi \equiv ((K_i\psi \wedge K_i(\psi \Rightarrow \psi')) \Rightarrow K_i\psi')$. Es gelte das Antezedenz $(M, s) \models ((K_i\psi \wedge K_i(\psi \Rightarrow \psi'))$ für ein beliebiges $s \in S$. In allen Welten t mit $(s, t) \in \mathcal{K}_i$ gilt sowohl $(M, t) \models \psi$ als auch $(M, t) \models \psi \Rightarrow \psi'$. Mit aussagenlogischem Schliessen und der Definition der Gültigkeitsrelation $\models$ folgt, dass $(M, t) \models \psi'$. Da $(s, t) \in \mathcal{K}_i$, gilt nun $(M, s) \models K_i\psi'$. Somit $M \models K_i\psi'$.
    - $\varphi \equiv (K_i\psi \Rightarrow \psi)$. Es sei $(M, s) \models K_i\psi$. Für alle t mit $(s, t) \in \mathcal{K}_i$ gilt dann $(M, t) \models \psi$. Da die Relation $\mathcal{K}_i$ reflexiv ist, gilt ebenfalls $(M, s) \models \psi$.
    - $\varphi \equiv (K_i\psi \Rightarrow K_iK_i\psi)$. Angenommen, es gelte $(M, s) \models K_i\psi$. Für beliebige t und u mit $\{(s, t), (t, u)\} \subseteq \mathcal{K}_i$ gilt zudem $(s, u) \in \mathcal{K}_i$, mit der Transitivität von $\mathcal{K}_i$. Mit $(M, s) \models K_i\psi$ folgt nun $(M, u) \models \psi$. In der Welt t gilt somit $(M, t) \models K_i\psi$. Für t galt andererseits $(s, t) \in \mathcal{K}_i$, es ergibt sich $(M, s) \models K_iK_i\psi$.
    - $\varphi \equiv (\neg K_i\psi \Rightarrow K_i\neg K_i\psi)$. Für eine beliebige Welt s sei $(M, s) \models \neg K_i\psi$. Somit existiert ein u mit $(s, u) \in \mathcal{K}_i$ und $(M, u) \models \neg\psi$. Die Welt t sei so gewählt, dass $(s, t) \in \mathcal{K}_i$. Die Relation $\mathcal{K}_i$ ist symmetrisch und transitiv, somit gilt $(t, u) \in \mathcal{K}_i$ und folglich $(M, t) \models \neg K_i\psi$. Da nun aber $(s, t) \in \mathcal{K}_i$, gilt demnach $(M, s) \models K_i\neg K_i\psi$.
- n>1:
  - Auf φ wurde mit der Regel **R1** (Modus Ponens) geschlossen. Es existieren somit Beweiszeilen φ_i, φ_j mit $i, j < n$ wobei $\varphi_i \equiv \psi$ und $\varphi_j \equiv (\psi \Rightarrow \varphi)$. Aufgrund der Induktionsvoraussetzung gilt $\mathcal{M}_n^{rst} \models \psi$ und $\mathcal{M}_n^{rst} \models (\psi \Rightarrow \varphi)$. Für ein beliebiges s in M gilt daher $(M, s) \models \psi$ und $(M, s) \models (\psi \Rightarrow \varphi)$. Mit der Definition der Gültigkeit folgt sofort $(M, s) \models \varphi$. Da s und M beliebig gewählt wurden, folgt $\mathcal{M}_n^{rst} \models \varphi$.
  - Auf φ wurde mit der Regel **R2** (Notwendigkeit) geschlossen. Somit hat φ die Form $K_i\psi$. Es existiert daher eine Beweiszeile

φ_i, $i < n$, mit $\varphi_i \equiv \psi$. Mit der Induktionsvoraussetzung folgt $\mathcal{M}_n^{rst} \models \psi$. Für alle t eines beliebigen $M \in \mathcal{M}_n^{rst}$ gilt also $(M,t) \models \psi$. Ein beliebiges s in M werde fixiert. Sicher gilt $(M,t) \models \psi$ für alle t mit $(s,t) \in \mathcal{K}_i$ (ψ gilt in allen t's, somit sicher auch in denjenigen, welche mit s in Relation stehen). Also erfüllt s die Formel $K_i\psi$, d.h. $(M,s) \models K_i\psi$. Dies gilt für alle s und $M \in \mathcal{M}_n^{rst}$, ergo $\mathcal{M}_n^{rst} \models K_i\psi$.

q.e.d.

1.6 Vollständigkeit

Nachdem im vorhergehenden Abschnitt die Korrektheit der eingeführten Axiomensysteme bewiesen wurde, soll nun deren Vollständigkeit nachgewiesen werden.

1.6.1 Der Vollständigkeitsbeweis

Theorem 1.6.1.

a) K_n ist ein vollständiges Axiomensystem bezüglich $\mathcal{M}_n$.

b) T_n ist ein vollständiges Axiomensystem bezüglich $\mathcal{M}_n^r$.

c) $S4_n$ ist ein vollständiges Axiomensystem bezüglich $\mathcal{M}_n^{rt}$.

d) $S5_n$ ist ein vollständiges Axiomensystem bezüglich $\mathcal{M}_n^{rst}$.

Beweis. Auch dieser Beweis soll nur für die Logik $S5_n$ geführt werden. Seien wiederum $\varphi, \psi, \psi' \in \mathcal{L}_n(\Phi), M \in \mathcal{M}^{rst}, s \in S$ und $i \in \{1, \ldots, n\}$.

Für die Vollständigkeit ist nachzuweisen: Gilt $\mathcal{M}_n^{rst} \models \varphi$, dann auch $S5_n \vdash \varphi$. Es genügt zu zeigen:

Jede $S5_n$-konsistente Formel ist in einer Struktur aus $\mathcal{M}_n^{rst}$ erfüllbar. $(*)$

Angenommen, $(*)$ wäre bewiesen und φ sei eine gültige Formel. Wäre φ nicht beweisbar, so auch $\neg\neg\varphi$ nicht und $\neg\varphi$ wäre konsistent. Mit $(*)$ würde die Erfüllbarkeit von $\neg\varphi$ in einer Struktur aus $\mathcal{M}_n^{rst}$ folgen, im Widerspruch zur Gültigkeit von φ bzgl. $\mathcal{M}_n^{rst}$.

Es bleibt also $(*)$ zu zeigen. Dies gelingt über die Konstruktion einer speziellen Kripke-Struktur, der sogenannten *kanonischen* Kripke-Struktur

$M^c \in \mathcal{M}_n^{rst}$, in welcher jede $S5_n$-konsistente Formel erfüllbar ist. Zu jeder maximal $S5_n$-konsistenten Menge V enthält M^c genau eine Welt. Es gilt dann

$$(M^c, V) \models \varphi \text{ gdw } \varphi \in V. \quad (**)$$

Mit $(**)$ würde dann $(*)$ sofort folgen, denn laut 1.4.4 ist jede $S5_n$-konsistente Formel φ in einer maximal konsistenten Menge V enthalten. Mit $(**)$ folgte $(M^c, V) \models \varphi$, und φ wäre erfüllbar.

Doch nun zur Konstruktion der kanonischen Struktur M^c. Für jede Menge V sei $V/K_i \doteq \{\varphi \ : \ K_i\varphi \in V\}$. $M^c = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ wird definiert durch

$$\begin{aligned} S &\doteq \{V \ : \ V \text{ ist maximal konsistent}\} \\ \pi_V(p) &\doteq \begin{cases} \textbf{true} \text{ falls } p \in V \\ \textbf{false} \text{ falls } p \notin V\} \end{cases} \\ \mathcal{K}_i &\doteq \{(V, W) \ : \ V/K_i \subseteq W\} \end{aligned}$$

Es bleibt zu prüfen, ob die so definierte kanonische Struktur M^c die gewünschten Eigenschaften besitzt, d. h. gilt für alle maximal $S5_n$-konsistenten Mengen $V : (M^c, V) \models \varphi$ genau dann, wenn $\varphi \in V$? Der Beweis erfolgt über den Aufbau von φ.

- φ ist eine primitive Proposition. Die Behauptung $(**)$ folgt unmittelbar aus der Definition von $\pi_V(p)$.

- $\varphi \equiv \psi \wedge \chi$. Somit gilt: $(M^c, V) \models \varphi$ gdw $(M^c, V) \models \psi$ und $(M^c, V) \models \chi$. Die Induktionsvoraussetzung liefert $\psi \in V$ und $\chi \in V$. Mit Lemma 1.4.4 gilt $\psi \wedge \chi \in V$. Die Beweisrichtung ist umkehrbar, wie leicht einzusehen ist.

- $\varphi \equiv \neg\psi$. Es gelte $(M^c, V) \models \neg\psi$. Folglich $(M^c, V) \not\models \psi$. Mit der Induktionsvoraussetzung ergibt sich $\psi \notin V$ und mit Lemma 1.4.4 gilt $\neg\psi \in V$. Auch in diesem Fall ist die Umkehrung des Beweises offensichtlich.

- $\varphi \equiv K_i\psi$. Sei $\varphi \in V$, folglich $\psi \in V/K_i$. Falls nun $(V, W) \in \mathcal{K}_i$, dann $\psi \in W$, aufgrund der Definition von $\mathcal{K}_i$. Die Induktionsvoraussetzung liefert nun $(M^c, W) \models \psi$ für alle W mit $(V, W) \in \mathcal{K}_i$. Somit ergibt sich $(M^c, V) \models K_i\psi$.

  Gelte nun $(M^c, V) \models K_i\psi$. Folglich ist die Menge $V/K_i \cup \{\neg\psi\}$ inkonsistent, wie die Gegenannahme zeigt: Wäre $V/K_i \cup \{\neg\psi\}$ konsistent, so läge diese Menge laut Lemma 1.4.4 in einer maximal konsistenten

Obermenge W und nach Konstruktion wäre auch $(V, W) \in \mathcal{K}_i$. Mit der Induktionsvoraussetzung ergäbe sich $(M^c, W) \models \neg\psi$ und somit $(M^c, V) \models \neg K_i\psi$, im Widerspruch zur Annahme.

Ist $V/K_i \cup \{\neg\psi\}$ nun aber inkonsistenz, so muss bereits eine endliche Teilmenge $\{\varphi_1, \ldots, \varphi_k, \neg\psi\} \subseteq V/K_i \cup \{\neg\psi\}$ inkonsistent sein. Es gilt daher:

$$\mathrm{S5}_n \vdash \neg(\varphi_1 \wedge \ldots \wedge \varphi_k \wedge \neg\psi)$$

Mit aussagenlogischem Schliessen ergibt sich

$$\begin{gathered}
\mathrm{S5}_n \vdash (\varphi_1 \wedge \ldots \wedge \varphi_k) \Rightarrow \psi \\
\mathrm{S5}_n \vdash (\varphi_1 \wedge \ldots \wedge \varphi_{k-1}) \Rightarrow (\varphi_k \Rightarrow \psi) \\
\mathrm{S5}_n \vdash (\varphi_1 \wedge \ldots \wedge \varphi_{k-2}) \Rightarrow (\varphi_{k-1} \Rightarrow (\varphi_k \Rightarrow \psi)) \\
\vdots \\
\mathrm{S5}_n \vdash \varphi_1 \Rightarrow (\varphi_2 \Rightarrow (\ldots(\varphi_k \Rightarrow \psi)\ldots))
\end{gathered}$$

Die Regel **R2** liefert

$$\mathrm{S5}_n \vdash K_i(\varphi_1 \Rightarrow (\varphi_2 \Rightarrow (\ldots(\varphi_k \Rightarrow \psi)\ldots))) \qquad (\clubsuit)$$

Axiom **K** lässt sich umformulieren zu

$$\vdash K_i(\varphi \Rightarrow \psi) \Rightarrow (K_i\varphi \Rightarrow K_i\psi)$$

Angewandt auf ($\clubsuit$) ergibt sich

$$\begin{aligned}
\mathrm{S5}_n \vdash & K_i(\varphi_1 \Rightarrow (\varphi_2 \Rightarrow (\ldots(\varphi_k \Rightarrow \psi)\ldots))) \\
& \Rightarrow (K_i\varphi_1 \Rightarrow K_i(\varphi_2 \Rightarrow (\varphi_3 \Rightarrow (\ldots(\varphi_k \Rightarrow \psi)\ldots))))
\end{aligned}$$

Die Iteration dieses Prozesses erzeugt

$$\begin{aligned}
\mathrm{S5}_n \vdash & K_i(\varphi_1 \Rightarrow (\varphi_2 \Rightarrow (\ldots(\varphi_k \Rightarrow \psi)\ldots))) \\
& \Rightarrow (K_i\varphi_1 \Rightarrow (K_i\varphi_2 \Rightarrow (\ldots(K_i\varphi_k \Rightarrow K_i\psi)\ldots)))
\end{aligned}$$

Mit dem Modus Ponens folgt schliesslich

$$\mathrm{S5}_n \vdash K_i\varphi_1 \Rightarrow (K_i\varphi_2 \Rightarrow (\ldots(K_i\varphi_k \Rightarrow K_i\psi)\ldots))$$

Erneutes aussagenlogisches Räsonieren liefert

$$\mathrm{S5}_n \vdash (K_i\varphi_1 \wedge \ldots \wedge K_i\varphi_k) \Rightarrow K_i\psi$$

resp.

$$\mathrm{S5}_n \vdash \neg(K_i\varphi_1 \wedge \ldots \wedge K_i\varphi_k \wedge \neg K_i\psi)$$

Die Menge $\{K_i\varphi_1, \ldots, K_i\varphi_k, \neg K_i\psi\}$ ist damit inkonsistent. Nun gilt aber $\varphi_1, \ldots, \varphi_k \in V/K_i$ und folglich $K_i\varphi_1, \ldots, K_i\varphi_k \in V$. Andererseits ist V maximal konsistent, daher muss $\neg K_i\psi \notin V$ gelten. Mit Lemma 1.4.4 folgt schliesslich das gewünschte Resultat $K_i\psi \in V$.

Es bleibt noch nachzuweisen, ob $M^c \in \mathcal{M}_n^{rst}$, d.h. sind die $\mathcal{K}_i$'s Äquivalenzrelationen?

- *Reflexivität:* Sei $\varphi \in V/K_i$. Es gilt dann: $\varphi \in V/K_i$ gdw $K_i\varphi \in V$. Sind in (M^c, V) alle Instanzen von **T** gültig, u.a. $(M^c, V) \models K_i\varphi \Rightarrow \varphi$, dann auch $(M^c, V) \models \varphi$. Es folgt somit $\varphi \in V$ und daher $V/K_i \subseteq V$. Per definitionem also $(V, V) \in \mathcal{K}_i$.

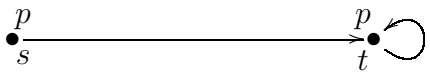
- *Transitivität:* Seien $(V, W), (W, X) \in \mathcal{K}_i$ und bei (M^c, V) gelten alle Instanzen von **4**. Somit folgt: $\varphi \in V/K_i$ gdw $K_i\varphi \in V$ gdw $(M^c, V) \models K_i\varphi$. Mit **4** und **R1** gilt $(M^c, V) \models K_iK_i\varphi$, somit $K_iK_i\varphi \in V$ und $K_i\varphi \in V/K_i$. Nach Voraussetzung folglich $K_i\varphi \in W$, was $\varphi \in W/K_i$ und $\varphi \in X$ nach sich zieht. Es gilt also $V/K_i \subseteq X$ und daher $(V, X) \in \mathcal{K}_i$.

- *Symmetrie:* Sei $(V, W) \in \mathcal{K}_i$. Zu zeigen ist $(W, V) \in \mathcal{K}_i$, d.h. $W/K_i \subseteq V$ oder $\varphi \notin V$ impliziert $\varphi \notin W/K_i$. Sei also $\varphi \notin V$. Dann $\varphi \notin V/K_i$ ($\mathcal{K}_i$ ist reflexiv), daher $K_i\varphi \notin V$. Mit Lemma 1.4.4 gilt $\neg K_i\varphi \in V$, mit **5** und **R1** $K_i\neg K_i\varphi \in V$. Somit $\neg K_i\varphi \in V/K_i$ und mit der Voraussetzung $\neg K_i\varphi \in W$. Erneut folgt mit Lemma 1.4.4 $K_i\varphi \notin W$ und mit der Reflexivität von $\mathcal{K}_i$ auch $\varphi \notin W/K_i$.

q.e.d.

1.6.2 Äquivalente Modelle

Nach dem Vollständigkeitstheorem ist jede Struktur M aus $\mathcal{M}_n^{rst}$ ein Modell von $\mathrm{S5}_n$. Die Umkehrung gilt leider nicht. Es gibt Modelle von $\mathrm{S5}_n$, die nicht in $\mathcal{M}_n^{rst}$ enthalten sind, wie das Beispiel zeigt.



Ein S5-Modell mit nicht reflexivem $\mathcal{K}$.

Nun, die Sache ist nicht ganz so schlimm. Zu jedem Modell von $S5_n$ existiert ein äquivalentes Modell aus $\mathcal{M}_n^{rst}$. Die genaue Bedeutung dieser Aussage wird im weiteren präzisiert.

Für eine beliebige Menge $A \subseteq \mathcal{S} \times \mathcal{S}$ bezeichne A^r den reflexiven, A^t den transitiven und A^s den symmetrischen Abschluss von A.

Natürlich können diese Abschlüsse auch kombiniert werden; so bezeichnet etwa A^{rst} den reflexiven, symmetrischen und transitiven Abschluss von A.

Definition 1.6.2. Sei $M = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ eine beliebige Struktur. Der reflexive, symmetrische und transitive Abschluss der Struktur M bezeichnet die Struktur $M^{rst} = (S, \pi, \mathcal{K}_1^{rst}, \ldots, \mathcal{K}_n^{rst})$.

Auch hier sind weitere Abschlusskombinationen denkbar, deren Definition analog zur oben erwähnten verläuft.

Definition 1.6.3. Für jede Kripke-Struktur M und jeden Zustand s aus M bezeichne das Paar (M, s) eine *Situation*.

Definition 1.6.4. Seien M und M' Kripke-Strukturen und s, s' Zustände aus M resp. M'. Die Situationen (M, s) und (M', s') heissen äquivalent, im Zeichen $(M, s) \equiv (M', s')$, falls $(M, s) \models \varphi$ gdw $(M', s') \models \varphi$ gilt, für alle Formeln $\varphi \in \mathcal{L}_n(\Phi)$.

Lemma 1.6.5.

a) Ist M ein Modell von T_n, dann auch sein reflexiver Abschluss M^r; zudem gilt $M^r \in \mathcal{M}_n^r$ und $(M, s) \equiv (M^r, s)$ für alle Zustände s aus M.

b) Ist M ein Modell von $S4_n$, dann auch sein reflexiver und transitiver Abschluss M^{rt}; zudem gilt $M^{rt} \in \mathcal{M}_n^{rt}$ und $(M, s) \equiv (M^{rt}, s)$ für alle Zustände s aus M.

c) Ist M ein Modell von $S5_n$, dann auch sein reflexiver, transitiver und symmetrischer Abschluss M^{rst}; zudem gilt $M^{rst} \in \mathcal{M}_n^{rst}$ und $(M, s) \equiv (M^{rst}, s)$ für alle Zustände s aus M.

Beweis. Ich beschränke mich auf den Nachweis der Behauptung (c). Der Beweis verläuft induktiv über den Aufbau von φ. Ist φ eine primitive Proposition oder von der Form $\neg\psi$, resp. $\psi \wedge \psi'$, so folgt die Behauptung unmittelbar

aus der Induktionsvoraussetzung. Sei somit $\varphi \equiv K_i\psi$. Gilt $(M^{rst}, s) \models K_i\psi$, so folgt $(M^{rst}, t) \models \psi$ für alle t mit $(s,t) \in \mathcal{K}_i^{rst}$. Mit der Induktionsvoraussetzung folgt $(M,t) \models \psi$, für alle t mit $(s,t) \in \mathcal{K}_i^{rst}$. Da $\mathcal{K}_i \subseteq \mathcal{K}_i^{rst}$, ergibt sich sofort $(M,s) \models K_i\psi$. Seien nun $(M,s) \models K_i\psi$ und $(s,t) \in \mathcal{K}_i^{rst}$. Dann existiert eine Folge $s_0, \ldots, s_k$ mit $s_0 = s, s_k = t$ und für alle $0 \leq j < k$ entweder

a) $(s_j, s_{j+1}) \in \mathcal{K}_i$ oder

b) $(s_{j+1}, s_j) \in \mathcal{K}_i$ oder

c) $s_i = s_{j+1}$.

Die Behauptung folgt nun mit einer Nebeninduktion über j, genauer, ich zeige $(M, s_j) \models \psi$ und $(M, s_j) \models K_i\psi$ für alle $0 \leq j \leq k$.
Sei $j = 1$.

a) Gilt $(s, s_1) \in \mathcal{K}_i$, dann folgt mit $(M,s) \models K_i\psi$ sofort $(M, s_1) \models \psi$. Zudem ergibt sich mit Axiom **4** ebenso $(M, s_1) \models K_i\psi$.

b) Ist $(s_1, s) \in \mathcal{K}_i$, und sei $(M, s_1) \models \neg K_i\psi$ angenommen. Dann folgt mit Axiom **5** unmittelbar $(M,s) \models \neg K_i\psi$, ein Widerspruch. Also gilt $(M, s_1) \models K_i\psi$. Mit Axiom **3** ergibt sich nun problemlos $(M, s_1) \models \psi$.

c) Es gilt $s = s_1$. Somit $(M, s_1) \models K_i\psi$ und mit Axiom **3** offenbar $(M, s_1) \models \psi$.

Sei nun $j > 1$ und weiterhin $(M,s) \models K_i\psi$. Für alle $j < k$ gilt $(M, s_j) \models K_i\psi$ und $(M, s_j) \models \psi$. Insbesondere, für $j = k-1$, $(M, s_{k-1}) \models K_i\psi$ und $(M, s_{k-1}) \models \psi$. Dieselben Argumente wie im Falle der Induktionsverankerung zeigen nun, dass $(M, s_k) \models K_i\psi$ und $(M, s_k) \models \psi$. Da $s_k = t$, folgt mit der Hauptinduktionsvoraussetzung $(M^{rst}, t) \models \psi$. Nun ist t ein beliebiger Zustand mit $(s,t) \in \mathcal{K}_i^{rst}$, also $(M^{rst}, s) \models K_i\psi$. q.e.d.

1.7 Die Entscheidbarkeit von S5$_n$

In den Theoremen 1.6.1 und 1.5.2 wurde gezeigt, dass die gültigen Formeln von $\mathcal{M}_n^{rst}$ wirklich die aus S5$_n$ beweisbaren sind. Leider ist die dort beschriebene Methode nicht konstruktiv; sie bietet keine Hilfe zur konkreten Überprüfung einer Formel auf Gültigkeit in $\mathcal{M}_n^{rst}$ resp. Beweisbarkeit aus S5$_n$. Dieses Problem soll in diesem Kapitel angegangen werden.

Zwei Möglichkeiten drängen sich dazu auf: Ein semantischer Test auf Erfüllbarkeit einer Formel φ in $\mathcal{M}_n^{rst}$ oder ein syntaktischer Test auf Beweisbarkeit einer Formel φ aus S5$_n$. Ich werde im folgenden die erstgenannte

Strategie verfolgen, was nach Korollar 1.3.5 bekanntlich keine Einschränkung darstellt.

Gesucht ist demnach ein Verfahren, dass die Frage nach der Gültigkeit einer Formel φ in $\mathcal{M}_n^{rst}$ entscheidet. Solche Verfahren sind auch unter dem Begriff *model-checking* bekannt. Meist werden sie auf endliche Kripke-Strukturen angewandt, d.h. auf Kripke-Strukturen, deren Zustandsmenge S endlich ist. Dies soll auch hier getan werden.

Vorerst aber ein kleiner Vorgriff auf Kapitel 3:

Definition 1.7.1. Sei g eine totale Funktion von $\mathbb{N}$ nach $\mathbb{N}$. Die Menge $\mathcal{O}(g(n))$ besteht aus allen Funktionen $f : \mathbb{N} \to \mathbb{N}$ für welche eine Konstante $c > 0$ so existiert, dass $f(n) \leq cg(n)$ für alle $n \in \mathbb{N}$ gilt. Formal:

$$\mathcal{O}(g(n)) \doteq \{f \mid f : \mathbb{N} \to \mathbb{N}, \exists c > 0 : f(n) \leq cg(n) \text{ für alle } n \in \mathbb{N}\}$$

1.7.1 Der allgemeine Fall

Definition 1.7.2. Sei $M = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ eine endliche Kripke-Struktur. $\|M\|$ wird definiert durch

$$\|M\| \doteq |S| + |\mathcal{K}_1| + \cdots + |\mathcal{K}_n|.$$

Lemma 1.7.3. *Sei $M = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ und $\varphi \in \mathcal{L}(\Phi)_n$. Es gibt einen Algorithmus, der die Erfüllbarkeit von φ in M in der Zeit $\mathcal{O}(\|M\| \cdot |\varphi|)$ entscheidet.*

Beweis. Sei $\varphi_1, \varphi_2, \ldots, \varphi_m$ eine der Länge nach geordnete Liste der Subformeln von φ. Es gilt dann $\varphi_m = \varphi$ und, falls φ_i eine Subformel von φ_j ist, auch $i < j$. Mit Lemma 1.2.8 folgt $m \leq |\varphi|$. Eine Induktion über k zeigt, dass jeder Zustand s von M in der Zeit $\mathcal{O}(k \cdot \|M\|)$ mit φ_k oder $\neg\varphi_k$ angeschrieben werden kann, je nachdem ob φ oder $\neg\varphi$ wahr ist bei s.

- $k = 1$: Somit $\varphi_1 \equiv p$, für ein $p \in \Phi$. Gilt $\pi_s(p) = \mathbf{true}$, dann wird s mit p angeschrieben, ansonsten mit $\neg p$. Dieser Vorgang ist sicher in der Zeit $\mathcal{O}(k \cdot \|M\|)$ ausführbar.

- $k > 1$:

  - $\varphi_k \equiv \neg\varphi_i$, mit $i < k$. Mit der Induktionsvoraussetzung ist s bereits mit φ_i oder $\neg\varphi_i$ angeschrieben. Entsprechend der Semantik der Negation wird nun s mit φ_k resp. $\neg\varphi_k$ markiert. Der hierzu notwendige Zeitaufwand liegt sicher in $\mathcal{O}(k \cdot \|M\|)$.

- $\varphi_k \equiv \varphi_i \wedge \varphi_j$, mit $i, j < k$. Da s nach der Induktionsvoraussetzung bereits mit den Konjunktionsteilnehmern, resp. deren Negation angeschrieben ist, lässt sich auch dieser Vorgang in der Zeit $\mathcal{O}(k \cdot \|M\|)$ ausführen.
- $\varphi_k \equiv K_i\varphi_j$, mit $j < k$. Der Zustand s wird mit $K_i\varphi_j$ angeschrieben, falls jedes t, wobei $(s,t) \in \mathcal{K}_i$ gilt, mit φ_j markiert ist. Nach der Induktionsvoraussetzung sind aber alle Zustände bereits mit φ_j resp. $\neg\varphi_j$ markiert. Da s höchstens $|S|$ Nachfolger besitzt, kann dieser Schritt in der Zeit $\mathcal{O}(k \cdot \|M\|)$ ausgeführt werden.

Nun gilt $\varphi_m = \varphi$. Ist φ in M erfüllbar, so gibt es einen Zustand s, der mit φ angeschrieben ist. Falls φ in M unerfüllt bleibt, existiert andrerseits ein mit $\neg\varphi$ markierter Zustand t. Das Kennzeichen des Zustands s mit φ_m resp. $\neg\varphi_m$ ist aber wie oben gezeigt in der Zeit $\mathcal{O}(m \cdot \|M\|)$ durchführbar. q.e.d.

Gemäss dem Beweis zum Vollständigkeitssatz aus 1.6.1 ist jede $S5_n$-konsistente Formel in einer Struktur aus $\mathcal{M}_n^{rst}$ erfüllbar. Diese Aussage wird nun verschärft:

Theorem 1.7.4. *Sei φ eine $S5_n$-konsistente Formel. Dann ist φ in einer Struktur $M_{fin} = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ aus $\mathcal{M}_n^{rst}$ mit höchstens $2^{|\varphi|}$ Zuständen erfüllbar. Zudem gilt für jede primitive Proposition $p \in \Phi$: Falls $p \notin Sub(\varphi)$, dann $(M_{fin}, s) \not\models p$ für alle $s \in S$.*

Die im Theorem erwähnte Struktur M_{fin} unterscheidet sich von der kanonischen Struktur M^c aus 1.6.1: M^c ist sicher nicht endlich! Obiges Theorem ist also ein echter Gewinn.

Beweis. Der Beweis verläuft ähnlich wie in 1.6.1, mit dem Unterschied, bloss maximal $S5_n$-konsistente Teilmengen von $Sub(\varphi)$ zu betrachten.
Es sei

$$Sub^+(\varphi) = Sub(\varphi) \cup \{\neg\psi : \psi \in Sub(\varphi)\}.$$

und

$$Con(\varphi) \doteq \{V : V \subseteq Sub^+(\varphi) \text{ und } V \text{ maximal } S5_n\text{-konsistent}\}$$

Eine leichte Abänderung des Beweises zum Theorem von Lindenbaum 1.4.4 zeigt, dass jede $S5_n$-konsistente Teilmenge von $Sub^+(\varphi)$ zu einem Element aus $Con(\varphi)$ erweitert werden kann. Die Modifikationen am Beweis von 1.4.4 sind äusserst minimal, ich verzichte daher auf eine Wiederholung.

Eine Menge $V \in Con(\varphi)$ enthält zu jeder Formel $\psi \in Sub(\varphi)$ entweder ψ oder $\neg\psi$, aber nicht beide, V wäre sonst inkonsistent. Daraus folgt:

$$|\, Con(\varphi)| \leq 2^{|\, Sub(\varphi)|} \leq 2^{|\varphi|}$$

Nun zur Konstruktion von $M_{fin} = (S_{fin}, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$:

$$\begin{array}{rcl} S_{fin} & \doteq & \{V \ : \ V \in Con(\varphi)\} \\ \pi_V(p) & \doteq & \left\{ \begin{array}{l} \textbf{true} \text{ falls } p \in V \\ \textbf{false} \text{ falls } p \notin V \} \end{array} \right. \\ \mathcal{K}_i & \doteq & \{(V,W) \ : \ V/K_i = W/K_i\} \end{array}$$

Offensichtlich gilt dank dieser Konstruktion für alle $p \in \Phi$ und $V \in S_{fin}$: Falls $p \notin Sub(\varphi)$, dann $(M_{fin}, V) \not\models p$. Weiter ist zu zeigen, dass für alle $V \in Con(\varphi)$ und $\psi \in Sub^+(\varphi)$ gilt:

$$\psi \in V \text{ gdw } (M_{fin}, V) \models \psi \tag{5}$$

Die Beweismethode ist ähnlich wie im Beweis von 1.6.1. Dem eingeschränkten Formeluniversum muss aber Rechnung getragen werden: mit $K_i\gamma \in V$ folgt nicht $K_iK_i\gamma \in V$, denn $K_iK_i\gamma$ ist womöglich gar keine Subformel von φ. Ich gehe daher kurz auf die Unterschiede ein:

- Ist ψ von der Form $\psi \equiv p, \psi \equiv \neg\xi$ oder $\psi \equiv \xi \wedge \xi'$, so ist (5) nach 1.6.1 sicher erfüllt, da sich die Konstruktionen von M^c und M_{fin} ja nur in den Möglichkeitsrelationen $\mathcal{K}_i$ unterscheiden.

- Sei $\psi \equiv K_i\xi$. Mit $\psi \in V$ folgt $\xi \in V/K_i$. Falls $(V,W) \in \mathcal{K}_i$, dann nach Konstruktion der $\mathcal{K}_i$ auch $\xi \in W/K_i$ und $K_i\xi \in W$. Nun ist W eine maximal $S5_n$-konsistente Teilmenge von $Sub^+(\varphi)$, W ist demnach, relativ zu $Sub^+(\varphi)$, deduktiv abgeschlossen. Mit dem Axiom **T** $(K_i\xi \Rightarrow \xi)$ liegt somit auch ξ in W, d.h. $V/K_i \subseteq W$. Mit der Induktionsvoraussetzung folgt nun $(M_{fin}, W) \models \xi$ für alle W mit $(V,W) \in \mathcal{K}_i$. Somit gilt $(M_{fin}, V) \models K_i\xi$.

  Gelte andererseits $(M_{fin}, V) \models K_i\xi$. Sei $K_i(V) \doteq \{K_i\alpha : K_i\alpha \in Sub^+(\varphi) \text{ und } K_i\alpha \in V\}$. Man beachte: $V/K_i = W/K_i$ gdw $K_i(V) = K_i(W)$. Weiter sei $\overline{K_i(V)} \doteq \{\neg K_i\beta : K_i\beta \in Sub^+(\varphi) \text{ und } K_i\beta \notin V\}$. Offensichtlich ist $K_i\beta \in \overline{K_i(V)}$ gdw $\neg K_i\beta \in V$. Ich behaupte nun, dass $K_i(V) \cup \overline{K_i(V)} \cup \{\neg\xi\}$ inkonsistent ist. Wäre dem nicht so, dann $K_i(V) \cup \overline{K_i(V)} \cup \{\neg\xi\} \subseteq W$ für ein $W \in Con(\varphi)$. Insbesondere $K_i(V) \subseteq K_i(W)$. Falls aber $K_i\delta \notin K_i(V)$, dann $K_i\delta \notin V$, also $\neg K_i\delta \in V$ und somit $\neg K_i\delta \in \overline{K_i(V)}$. Mit $\overline{K_i(V)} \subseteq W$ folgt

$\neg K_i\delta \in W$ und schliesslich $K_i\delta \notin W$. Der langen Rede kurzer Sinn: $K_i(W) \subseteq K_i(V)$. Es gilt demnach $K_i(V) = K_i(W)$, d.h. $(V, W) \in \mathcal{K}_i$. Somit $(M_{fin}, W) \models \xi$. Mit der Induktionsvoraussetzung ergibt sich $\xi \in W$. Da aber auch $\neg\xi \in W$, ist W inkonsistent, im Widerspruch zur Annahme.

Die Menge $K_i(V) \cup \overline{K_i(V)} \cup \{\neg\xi\}$ ist demnach inkonsistent. Daraus folgt:

$$\mathrm{S5}_n \vdash \neg(K_i\alpha_1 \wedge \cdots \wedge K_i\alpha_n \wedge \neg K_i\beta_1 \wedge \cdots \wedge \neg K_i\beta_m \wedge \neg\xi)$$

Mit aussagenlogischem Schliessen folgt

$$\begin{gathered}\mathrm{S5}_n \vdash (K_i\alpha_1 \wedge \cdots \wedge K_i\alpha_n \wedge \neg K_i\beta_1 \wedge \cdots \wedge \neg K_i\beta_m) \Rightarrow \xi \\ \vdots \\ \mathrm{S5}_n \vdash K_i\alpha_1 \Rightarrow (\ldots(K_i\alpha_n \Rightarrow (\neg K_1\beta \Rightarrow (\ldots(\neg K_i\beta_m \Rightarrow \xi)))))\end{gathered}$$

Die Anwendung der Regel **R2** ergibt

$$\mathrm{S5}_n \vdash K_i(K_i\alpha_1 \Rightarrow (\ldots(K_i\alpha_n \Rightarrow (\neg K_1\beta \Rightarrow (\ldots(\neg K_i\beta_m \Rightarrow \xi))))))$$

Mit Axiom **K** und dem Modus Ponens folgt

$$\mathrm{S5}_n \vdash K_iK_i\alpha_1 \Rightarrow (\ldots(K_iK_i\alpha_n \Rightarrow (K_i\neg K_i\beta_1 \Rightarrow (\ldots(K_i\neg K_i\beta_m \Rightarrow K_i\xi)))))$$

Nun lassen sich in $\mathrm{S5}_n$ folgende Äquivalenzen zeigen:

$$\begin{aligned}\mathrm{S5}_n &\vdash K_i\varrho \equiv K_iK_i\varrho \\ \mathrm{S5}_n &\vdash \neg K_i\varrho \equiv K_i\neg K_i\varrho\end{aligned}$$

Angewandt auf obige Formel

$$\mathrm{S5}_n \vdash K_i\alpha_1 \Rightarrow (\ldots(K_i\alpha_n \Rightarrow (\neg K_i\beta_1 \Rightarrow (\ldots(\neg K_i\beta_m \Rightarrow K_i\xi)))))$$

Erneuter Einsatz der Aussagenlogik liefert sukzessive

$$\begin{aligned}\mathrm{S5}_n &\vdash (K_i\alpha_1 \wedge \cdots \wedge K_i\alpha_n \wedge \neg K_i\beta_1 \wedge \cdots \wedge \neg K_i\beta_m) \Rightarrow K_i\xi \\ \mathrm{S5}_n &\vdash \neg(K_i\alpha_1 \wedge \cdots \wedge K_i\alpha_n \wedge \neg K_i\beta_1 \wedge \cdots \wedge \neg K_i\beta_m \wedge \neg K_i\xi)\end{aligned}$$

Folglich ist die Menge $\{K_i\alpha_1, \ldots, K_i\alpha_n, \neg K_i\beta_1, \ldots, \neg K_i\beta_m, \neg K_i\xi\}$ inkonsistent. Nun gilt aber $K_i\alpha_i \in V$ ebenso wie $\neg K_i\beta_i \in V$ und da V maximal $\mathrm{S5}_n$-konsistent ist (bzgl. $Sub^+(\varphi)$ natürlich), ergibt dies $\neg K_i\xi \notin V$ resp. $K_i\xi \in V$. Damit ist (5) bewiesen.

- Es bleibt noch $M_{fin} \in \mathcal{M}_n^{rst}$ zu prüfen. Mit der Definition der $\mathcal{K}_i$ folgt dies aber auf dem Fuss.

q.e.d.

Bemerkung 1.7.5. Die aus dem Beweis von Theorem 1.6.1 bekannte Definition der Relationen $\mathcal{K}_i \doteq \{(V, W) : V/K_i \subseteq W\}$ ist für endliche Strukturen nicht weiter passend. Ist φ z. B. die Formel Kp, dann sind $V_1 = \{Kp, p\}, V_2 = \{\neg Kp, p\}$ und $V_3 = \{\neg Kp, \neg p\}$ die maximal $S5_n$-konsistenten Mengen. Somit sind $V_1/K = \{p\}, V_2/K = V_3/K = \emptyset$. Demnach $V_1/K \subseteq V_2$ und $V_2/K \subseteq V_3$, aber $V_1/K \not\subseteq V_3$, eine Verletzung der Transitivität. Zwar gilt $V_1/K \subseteq V_2$, doch produziert $(V_1, V_2) \in \mathcal{K}$ mulmige Gefühle. Jede maximal $S5_n$-konsistente Erweiterung $\tilde{V}_1$ von V_1 wird auch KKp enthalten, und folglich $Kp \in \tilde{V}_1/K$. Die Beziehung $\tilde{V}_1/K \subseteq \tilde{V}_2$ ist dann aber nicht mehr denkbar.

Mit Theorem 1.7.4 ist es nun möglich, die Frage nach der Erfüllbarkeit einer Formel φ bezüglich $\mathcal{M}_n^{rst}$ zu klären. Es werden schlicht alle Kripke-Strukturen M aus $\mathcal{M}_n^{rst}$ mit $|S| = 2^{|\varphi|}$ und $M \models \neg p$ für alle $p \notin Sub^+(\varphi)$ konstruiert. Die Anzahl dieser Strukturen mag riesig sein, doch ist sie zumindest endlich. Schliesslich wird die Erfüllbarkeit von φ in jeder dieser Strukturen überprüft, z.B. mit dem Algorithmus aus Lemma 1.7.3. Wird φ in einer dieser Strukturen erfüllt, so ist φ sicher erfüllbar bezüglich $\mathcal{M}_n^{rst}$. Andererseits, falls φ bezüglich $\mathcal{M}_n^{rst}$ erfüllbar ist, so muss φ nach Theorem 1.7.4 in einer Struktur M_{fin} erfüllbar sein.

Korollar 1.7.6. *Das Gültigkeitsproblem für $\mathcal{M}_n^{rst}$ und das Beweisbarkeitsproblem für $S5_n$ sind entscheidbar.*

Beweis. Da φ $S5_n$-beweisbar gdw φ gültig in $\mathcal{M}_n^{rst}$ gdw $\neg\varphi$ nicht erfüllbar in $\mathcal{M}_n^{rst}$ gilt, wird einfach $\neg\varphi$ auf Erfüllbarkeit getestet. q.e.d.

1.7.2 Der Spezialfall S5

Die oben präsentierten Resultate betreffen die Logiken $S5_n$ für alle $n \geq 1$. Im Falle von $n = 1$ ergeben sich aber einige nette Vereinfachungen, welche auch Aufnahme in diese Arbeit finden sollen.

Lemma 1.7.7. *Sei φ eine S5-konsistente Formel. Dann ist φ erfüllbar in einer Struktur $M = (S, \pi, \mathcal{K})$ mit $\mathcal{K} = \{(s, t) : s, t \in S\}$, d.h. $\mathcal{K}$ ist die Allrelation.*

Beweis. Sei φ S5-konsistent. Dann existieren nach Theorem 1.6.1 eine Struktur $M' = (S', \pi', \mathcal{K}')$ mit $M' \in \mathcal{M}^{rst}$ und ein $s_0 \in S'$ so, dass $(M', s_0) \models \varphi$.

Setze $\mathcal{K}'(s_0) = \{t : (s_0, t) \in \mathcal{K}'\}$. Dank der Reflexivität von $\mathcal{K}'$ gilt $\mathcal{K}'(s_0) \neq \emptyset$. Da $\mathcal{K}'$ transitiv und symmetrisch ist, gilt $(s,t) \in \mathcal{K}'$ für alle $s, t \in \mathcal{K}'(s_0)$. Zudem, falls $s \in \mathcal{K}'(s_0)$ und $(s,t) \in \mathcal{K}'$, folgt mit der Transitivität von $\mathcal{K}'$, dass $t \in \mathcal{K}'(s_0)$.
Die Struktur $M = (S, \pi, \mathcal{K})$ wird nun definiert durch:

$$\begin{aligned} S &\doteq \mathcal{K}'(s_0) \\ \pi &\doteq \pi'|_{\mathcal{K}'(s_0)} \\ \mathcal{K} &\doteq \{(s,t) : s, t \in \mathcal{K}'(s_0)\} \end{aligned}$$

Offensichtlich ist $\mathcal{K}$ die Allrelation auf $\mathcal{K}'(s_0)$. Für alle $s \in \mathcal{K}'(s_0)$ und $\psi \in \mathcal{L}_n(\Phi)$ zeigt eine Induktion über den Formelaufbau, dass $(M, s) \models \psi$ gdw $(M', s) \models \psi$.

- $\psi \equiv p$: Die Behauptung folgt unmittelbar, da π und π' auf $\mathcal{K}'(s_0)$ übereinstimmen.
- $\psi \equiv \neg\eta$: $(M, s) \models \neg\eta$ gdw $(M, s) \not\models \eta \overset{\text{I.V.}}{\text{gdw}} (M', s) \not\models \eta$ gdw $(M', s) \models \neg\eta$.
- $\psi \equiv \eta \wedge \vartheta$: $(M, s) \models \eta \wedge \vartheta$ gdw $(M, s) \models \eta$ und $(M, s) \models \vartheta \overset{\text{I.V.}}{\text{gdw}}$ $(M', s) \models \eta$ und $(M', s) \models \vartheta$ gdw $(M', s) \models \eta \wedge \vartheta$.
- $\psi \equiv K\eta$: $(M, s) \models K\eta$ gdw $(M, t) \models \eta$ für alle $t \in S$ ($\mathcal{K}$ ist die Allrelation), d.h. $(M, t) \models \eta$ für alle $t \in \mathcal{K}'(s_0)$. Falls nun u ein beliebiger Zustand mit $(s, u) \in \mathcal{K}'$ ist, so folgt, da $s \in \mathcal{K}'(s_0)$, auch $u \in \mathcal{K}'(s_0)$ (wie oben erwähnt). Mit der Induktionsvoraussetzung gilt $(M', u) \models \eta$ für alle $u \in \mathcal{K}'(s_0)$. Somit ergibt sich $(M', s) \models K\eta$. Umkehrung: $(M', s) \models K\eta$ gdw $(M', t) \models \eta$ für alle $(s, t) \in \mathcal{K}'$. Sei $u \in \mathcal{K}'(s_0)$. Zu zeigen bleibt $(M, u) \models \eta$. Da auch $s \in \mathcal{K}'(s_0)$ folgt $(s, u) \in \mathcal{K}'$. Mit der Induktionsvoraussetzung gilt aber $(M, u) \models \eta$ für alle $(s, u) \in \mathcal{K}'$ und damit die Behauptung.

q.e.d.

Die Formeln von S5 lassen sich nun in Kripke-Strukturen mit kleiner Zustandsmenge erfüllen. Das folgende, von Ladner [Lad77] stammende Ergebnis, wird die Bestimmung der Komplexität des Erfüllbarkeitsproblems von S5 erheblich erleichtern.

Lemma 1.7.8. *Ein Formel φ der Sprache S5 ist genau dann erfüllbar, wenn sie in einer Struktur M aus $\mathcal{M}^{rst}$ mit höchstens $|\varphi|$ Zuständen erfüllbar ist.*

Beweis. Natürlich bleibt nur die Richtung von links nach rechts zu beweisen, die Umkehrung ist trivial. Sei also $(M, s) \models \varphi$, wobei $M = (S, \pi, \mathcal{K})$. Mit dem Resultat aus 1.7.7 darf ohne Einschränkung der Allgemeinheit angenommen werden, dass $\mathcal{K} = \{(t, u) : t, u \in S\}$. Sei $F \subseteq Sub(\varphi)$ definiert durch:

$$F \doteq \{K\psi : K\psi \in Sub(\varphi) \text{ und } (M, s) \models \neg K\psi\}$$

Zu jeder Formel $K\psi \in F$ gibt es einen Zustand $s_\psi \in S$ mit $(M, s_\psi) \models \neg\psi$. Man definiere weiter die Struktur $M' = (S', \pi', \mathcal{K}')$ durch:

$$\begin{aligned} S' &\doteq \{s\} \cup \{s_\psi : K\psi \in F\} \\ \pi' &\doteq \pi|_{S'} \\ \mathcal{K}' &\doteq \{(t, t') : t, t' \in S'\} \end{aligned}$$

Da $|F| < |\,Sub(\varphi)| \leq |\varphi|$, folgt $|S'| \leq |\varphi|$. Ich zeige nun für alle Zustände $s' \in S'$ und für alle $\psi \in Sub(\varphi)$ (inkl. φ selbst), dass $(M, s') \models \psi$ gdw $(M', s') \models \psi$. Wie so oft verläuft der Beweis mittels Induktion über den Aufbau von ψ.

- Die Fälle $\psi \equiv p, \psi \equiv \neg\kappa$ und $\psi \equiv \kappa \wedge \vartheta$ sind trivial. π unterscheidet sich von π' auf S' nicht, es treten in diesen Fällen somit keine Schwierigkeiten auf.

- $\psi \equiv K\vartheta$: Sei $s' \in S'$. Falls $(M, s') \models K\vartheta$, dann $(M, t) \models \vartheta$ für alle $t \in S$, somit gilt sicher auch $(M, t) \models \vartheta$ für alle $t \in S'$. Mit der Induktionsvoraussetzung folgt $(M', t) \models \vartheta$ für alle $t \in S'$, somit $(M', s') \models K\vartheta$. Zur Umkehrung: Gelte $(M, s') \not\models K\vartheta$. Es existiert also ein $t \in S$ mit $(M, t) \models \neg\vartheta$. Nun ist $\mathcal{K}$ die Allrelation auf S, somit gilt insbesondere auch $(s, t) \in \mathcal{K}$ und damit $(M, s) \models \neg K\vartheta$. Daraus folgt $K\vartheta \in F$ und $(M, s_\vartheta) \models \neg\vartheta$. Die Konstruktion von M' garantiert $s_\vartheta \in S'$ und mit der Induktionsvoraussetzung gilt $(M', s_\vartheta) \models \neg\vartheta$. Zudem ist $(s', s_\vartheta) \in \mathcal{K}'$, also $(M', s') \models \neg K\vartheta$. Daraus folgt, wie gewünscht, $(M', s') \not\models K\vartheta$.

Da nun $s \in S'$ und $(M, s) \models \varphi$ gemäss der Voraussetzung, ergibt sich $(M', s) \models \varphi$. q.e.d.

Kapitel 2

Allgemeinwissen

2.1 Ein Beispiel

Die Königin eines matriarchalischen Stammes - ich nenne sie der Einfachheit halber Amazonen, obwohl die echten Amazonen nicht verheiratet waren - versammelte eines Tages alle Frauen und verkündete ihnen, dass wenigstens einer ihrer Ehemänner untreu gewesen sei. „Denjenigen von euch, die herausfinden, dass ihr Ehemann sie betrügt, befehle ich, ihn zur Mitternachtsstunde des Tages, an dem ihr seine Untreue erkannt habt, zu töten."

Neuigkeiten verbreiten sich stets in Windeseile unter den weiblichen Stammesmitgliedern; nur den Betroffenen selbst wurden sie rücksichtsvoll verschwiegen. Alle Frauen erfuhren also sofort von der Untreue der andern Männer, nicht aber von der des eigenen Mannes. Auch die Nachricht von einer Exekution machte innerhalb eines Tages die Runde. Nun waren, wie der Chronist zu vermelden weiss, genau 40 Ehemänner untreu. Die Frage lautet also: Wurden welche von ihnen getötet, und wenn ja, wann?

Der aufmerksame Leser wird bemerkt haben, dass die Königin von mindestens einem untreuen Ehemann gesprochen hat. Wäre es genau einer gewesen, hätte dessen Ehefrau sofort Bescheid gewusst; wäre nämlich ein anderer der Ehebrecher gewesen, hätte die Frau ja davon gehört gehabt. Also wäre dieser Mann zur Mitternachtsstunde des Tages, an dem die Königin gesprochen hatte, von seiner Frau ins Jenseits befördert worden.

Bei genau zwei untreuen Ehemännern hätten beide zur Mitternachtsstunde des zweiten Tages von ihren Ehefrauen den Todesstoss empfangen. Bei der Nachricht, dass zur Mitternacht des ersten Tages niemand exekutiert wurde, wäre den Ehefrauen der beiden ein Licht aufgegangen. Da sie jeweils nur von einem untreuen Ehemann gehört gehabt hätten (alle andern Ehefrauen hätten dagegen gewusst, dass es wenigstens zwei waren), wäre ihnen sofort

klar geworden, dass ihr eigener Ehemann der zweite Wüstling war.

Inzwischen kann der Leser sich wohl schon denken, wie die Geschichte weiter geht. Keine Exekution zur Mitternacht des n-ten Tages heisst, dass wenigstens n+1 Ehemänner untreu gewesen sein müssen. In der Frühe des 40. Tages hätten mithin alle Frauen gewusst, dass wenigsten 40 Männer die Ehe gebrochen hatten. Für die mit den treuen Ehemännern wäre dies keine Überraschung gewesen, denn sie hätten die 40 Ehebrecher ja gekannt. Eine Frau mit untreuem Mann aber hätte nur von 39 untreue Männern gewusst; also musste ihr eigener der 40. sein. Diese Frauen hätten also am 40. Tag nach der Eröffunung der Königin um Mitternacht den Seitensprüngen ihrer Männer ein für allemal ein Ende bereitet.

Soviel zur Ethnologie, zurück nun zur Logik. Eine in diesem Zusammenhang interessante Feststellung ist folgende: Der Chronist berichtet von 40 untreuen Ehemännern. Demnach wusste jede Ehefrau lange vor der königlichen Ankündigung, dass mindestens ein Ehemann untreu war. Die Königin offenbarte den Frauen somit eine Mitteilung, die diese bereits kannten. Worin lag aber der Informationsgehalt dieser Eröffnung? Hätte die Königin diese Aussage nicht auch weglassen können, ohne dass die Untaten der Ehemänner im Dunkeln blieben?

Zur Beantwortung dieser Frage möchte ich eine einfachere Situation betrachten: Das Volk der Amazonen bestehe neben der Königin aus bloss drei Frauen und allesamt wurden sie von ihren Ehemänner betrogen. Die Frauen seien mit den Buchstaben A,B und C bezeichnet. Jede der Frauen weiss demnach, dass ihre beiden Freundinnen betrogen wurden. Die Frauen wissen aber noch mehr: Ehefrau A kennt die Namen der fehlbaren Ehemänner von B und C, Frau B weiss hingegen die Namen der Übeltäter von A und C. Den Namen des Mannes von Frau C erfuhr B aber von A, also weiss Frau A, dass B mindestens einen Ehebrecher kennt. Weitere Gedankengänge dieser Art zeigen, dass jeder Frau weiss, dass jede ihrer Freundinnen über einen Ehebruch informiert ist.

Amazone A könnte nun wie folgt räsonieren:„Angenommen, mein Eheman ist mir treu. B könnte ebenfalls von der Loyalität ihres Mannes ausgehen. In diesem Fall käme B zum Schluss, dass C nicht wissen könnte, dass überhaupt ein Ehebruch begangen wurde.“ Die Offenbarung der Königin schliesst aber gerade diese Situation aus: A weiss nun, dass B weiss, dass auch C von einer gehörnten Ehefrau erfahren hat. Die Aussage „Mindestens ein Ehemann war untreu“ wird Allgemeinwissen. Die Königin liefert mit ihrer Eröffnung demnach weitere, bisher unbekannte Information, welche des Rätsels Lösung überhaupt erst ermöglicht.

2.2 Die Logik $\mathbf{S5}_n^C$

Die Unterschiede zwischen „Agent i weiss, dass p“, „alle Agenten wissen, dass p“ und „die Aussage p ist Allgemeinwissen“ sind subtil und umgangssprachlich nur schwer fassbar, wie das Beispiel zeigt. Die seriöse Analyse der Differenzen verlangt einen formalen Rahmen. Dieser kann beispielsweise durch eine natürliche Erweiterung der Sprache $\mathrm{S5}_n$ bereitgestellt werden.

2.2.1 $\mathbf{S5}_n^C$ als Erweiterung der Sprache $\mathbf{S5}_n$

Definition 2.2.1. Sei $\mathrm{L}_n \in \{\mathrm{K}_n, \mathrm{T}_n, \mathrm{S4}_n, \mathrm{S5}_n\}$. Das Alphabet der Sprache L_n^C besteht aus folgenden Grundzeichen:

a) Eine endliche Menge Φ von primitiven Propositionen

b) Logischen Symbolen $\neg, \wedge, \vee$

c) Modalen Operatoren $K_1, ..., K_n, E, C$, für $n \geq 1$

d) Hilfszeichen (,)

Definition 2.2.2. Sei $\mathrm{L}_n \in \{\mathrm{K}_n, \mathrm{T}_n, \mathrm{S4}_n, \mathrm{S5}_n\}$. Die Formeln der Sprache L_n^C werden durch endlichmalige Anwendung der folgenden Regeln erhalten:

a) Ist φ eine Formel von L_n, dann ist φ auch eine Formel von L_n^C.

b) Ist φ eine Formel von L_n^C, dann ist auch $E\varphi$ eine Formel von L_n^C.

c) Ist φ eine Formel von L_n^C, dann ist auch $C\varphi$ eine Formel von L_n^C.

Die Menge aller Formeln von L_n^C wird mit $\mathcal{L}_n^C(\Phi)$ bezeichnet.

Sinngemäss angepasst wird $|\varphi|$, die Grösse der Formel φ. $|\varphi|$ bezeichne die Länge von φ über dem Alphabet $\Phi \cup \{\neg, \wedge, (,), K_1, \ldots, K_n, E, C\}$.

Eine gravierendere Anpassung erfährt dagegen der Begriff der Subformel:

Definition 2.2.3. Die Menge $\mathrm{Sub}_C(\varphi)$ der Subformeln einer Formel φ aus $\mathcal{L}_n^C(\Phi)$ wird induktiv definiert durch:

a) Gilt $\psi \in \mathrm{Sub}(\varphi)$, dann auch $\psi \in \mathrm{Sub}_C(\varphi)$.

b) $\mathrm{Sub}_C(C\psi) = \{C\psi, E(\psi \wedge C\psi), \psi \wedge C\psi, K_1(\psi \wedge C\psi), \ldots, K_n(\psi \wedge C\psi)\} \cup \mathrm{Sub}_C(\psi)$

c) $\mathrm{Sub}_C(E\psi) = \{E\psi, K_1\psi, \ldots, K_n\psi\} \cup \mathrm{Sub}_C(\psi)$

Bemerkung 2.2.4. Diese auf den ersten Blick etwas seltsam anmutende Definition der Menge $Sub_C\,\varphi$ erhält im Abschnitt 2.2.2 ihre Berechtigung. Der Leser möge doch seine Ungeduld bis dahin bändigen oder andernfalls einen kurzen Ausblick riskieren.

Trotz den zusätzlichen Subformeln von $\mathrm{Sub}_C(\varphi)$ bleibt das Wachstum von $|\mathrm{Sub}_C(\varphi)|$ moderat, wie folgendes Lemma zeigt:

Lemma 2.2.5. *Für alle* $\varphi \in \mathcal{L}_n^C(\Phi)$ *gilt:* $|Sub_C(\varphi)| \leq (n+3)|\varphi|$.

Beweis. Induktion über die Struktur von φ.

a) $\varphi \in \mathcal{L}_n(\Phi)$: Mit Lemma 1.2.8 gilt $|\mathrm{Sub}_C(\varphi)| = |\mathrm{Sub}(\varphi)| \leq |\varphi| \leq (n+3)|\varphi|$

b) $\varphi \equiv E\psi$: $|\mathrm{Sub}_C(E\psi)| = |\{E\psi, K_1\psi, \ldots, K_n\psi\}| + |\mathrm{Sub}_C(\psi)| \overset{\text{I.V.}}{\leq} (n+1) + (n+3)|\psi| \leq (n+3) + (n+3)|\psi| = (n+3)(|\psi|+1) = (n+3)|E\psi|$

c) $\varphi \equiv C\psi$: $|\mathrm{Sub}_C(C\psi)| = \{C\psi, E(\psi \wedge C\psi), \psi \wedge C\psi, K_1(\psi \wedge C\psi), \ldots, K_n(\psi \wedge C\psi)\}| + |\mathrm{Sub}_C(\psi)| \overset{\text{I.V.}}{\leq} (n+3) + (n+3)|\psi| = (n+3)(|\psi|+1) = (n+3)|C\psi|$

q.e.d.

Das einleitende Beispiel legt die Vermutung Nahe, Allgemeinwissen als unendliche Konjunktion der Form „jeder Agent weiss φ und jeder Agent weiss, dass jeder Agent φ weiss und jeder Agent weiss, dass jeder Agent weiss, dass jeder Agent φ weiss usw.“ aufzufassen. Diese Intention führt zu nachfolgender Definition. Doch zuvor noch eine notwendige Hilfsdefinition:

Definition 2.2.6.

$$\begin{aligned} E^1\varphi &\doteq E\varphi \\ E^{k+1}\varphi &\doteq E(E^k\varphi) \text{ für } k \geq 1 \end{aligned}$$

Definition 2.2.7. Sei $M = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ eine beliebige Kripke-Struktur. Die Semantik der Operatoren E und C wird folgendermassen definiert:

a) $(M,s) \models E\varphi$ gdw $(M,s) \models K_i\varphi$ für alle $i \in \{1, \ldots, n\}$

b) $(M,s) \models C\varphi$ gdw $(M,s) \models E^k\varphi$ für $k = 1, 2, \ldots$

Auf Seite 4 wurde auf die graphentheoretische Interpretation der Kripke-Strukturen hingewiesen. Diese Deutung erweist sich bezüglich der Operatoren E und C als besonders hilfreich.

Definition 2.2.8 (Erreichbarkeit).

a) Ein Zustand t ist *erreichbar vom Zustand s in $k \geq 1$ Schritten*, falls Zustände $s_0, s_1, \ldots, s_k$ derart existieren, dass folgendes gilt:

   (a) $s = s_0, t = s_k$

   (b) für alle j mit $0 \leq j \leq k-1$ gibt es einen Agenten i_j so, dass $(s_j, s_{j+1}) \in \mathcal{K}_{i_j}$.

b) Ein Zustand t ist *erreichbar vom Zustand s*, falls t von s in k Schritten erreichbar ist, mit beliebigem k.

Ein Zustand t ist demnach vom Zustand s genau dann erreichbar, wenn im Graphen ein Pfad von s nach t führt.

Lemma 2.2.9.

a) $(M, s) \models E^k\varphi$ *gdw* $(M, t) \models \varphi$ *für alle t, die von s in k Schritten erreichbar sind.*

b) $(M, s) \models C\varphi$ *gdw* $(M, t) \models \varphi$ *für alle von s erreichbaren Zustände t.*

Beweis. a) Induktion über k.

- $k = 1$: $(M, s) \models E\varphi$ gdw $(M, s) \models K_i\varphi$ für alle $i \in \{1, \ldots, n\}$. Somit existiert zu jedem K_i ein t_i mit $(M, t_i) \models \varphi$ und $(s, t_i) \in \mathcal{K}_i$. Folglich gilt φ in allen Zuständen t, welche von s in einem Schritt erreichbar sind.

  Gelte andererseits $(M, t) \models \varphi$ für alle t, die in einem Schritt von s erreichbar sind. Somit gibt es zu jedem t einen Agenten i_t so, dass $(s, t) \in \mathcal{K}_{i_t}$. Es gilt daher $(M, s) \models K_{i_t}\varphi$ für alle von s in einem Schritt erreichbaren t. Daraus folgt $(M, s) \models K_i$ für alle $i \in \{1, \ldots, n\}$. Gibt es nämlich für ein j kein t mit $(s, t) \in \mathcal{K}_j$, dann gilt sicher $(M, s) \models K_j\varphi$, aufgrund der Semantik der Operatoren K_i.

- $k > 1$: $(M, s) \models E^{k+1}\varphi$ gdw $(M, s) \models E(E^k\varphi)$. Gemäss der Induktionsverankerung gilt $(M, t) \models E^k\varphi$ für alle von s in einem Schritt erreichbaren t. Mit der Induktionsvoraussetzung folgt $(M, t') \models \varphi$ für alle von t in k Schritten erreichbaren t'. Da t von s in einem Schritt erreichbar war, ist t' von s in $k+1$ Schritten erreichbar. Es gilt also $(M, t') \models \varphi$ für alle von s in $k+1$ Schritten erreichbaren t'.

Sei nun $(M,t) \models \varphi$ für alle von s in $k+1$ Schritten erreichbaren t. s' sei von s in einem Schritt erreichbar. Sicher gilt nun $(M,t) \models \varphi$ für alle von s' in k Schritten erreichbaren t. Mit der Induktionsvoraussetzung folgt $(M,s') \models E^k\varphi$ für alle von s' in k Schritten erreichbaren t. Da s' ein beliebiger von s in einem Schritt erreichbaren Zustand ist, liefert die Induktionsverankerung $(M,s) \models E(E^k\varphi)$ und damit $(M,s) \models E^{k+1}\varphi$.

b) Die Aussage b) folgt unmittelbar aus der Aussage a).

q.e.d.

Die graphentheoretische Interpretation der Kripke-Struktur bringt eine weitere interessante Eigenschaft der Operatoren E und C ans Licht. Sei $\mathcal{E} \doteq \mathcal{K}_1 \cup \cdots \cup \mathcal{K}_n$ und $\mathcal{C}$ sei der transitive Abschluss von $\mathcal{E}$. Es ergibt sich

$$(M,s) \models E\varphi \quad \text{gdw} \quad (M,t) \models \varphi \text{ für alle } t \text{ mit } (s,t) \in \mathcal{E}$$
$$(M,s) \models C\varphi \quad \text{gdw} \quad (M,t) \models \varphi \text{ für alle } t \text{ mit } (s,t) \in \mathcal{C}$$

Die Operatoren E und C verhalten sich demnach wie zwei imaginäre Individuen, deren Möglichkeitsrelationen durch $\mathcal{E}$ resp. $\mathcal{C}$ gegeben sind.

2.2.2 Axiomatisiertes Allgemeinwissen

Allgemeinwissen wird nun in die zuvor in 1.4 präsentierten Systeme eingegliedert. Die zusätzlichen Axiome lauten:

7. $E\varphi \equiv K_1\varphi \wedge \cdots \wedge K_n\varphi$
8. $C\varphi \Rightarrow E(\varphi \wedge C\varphi)$

Die neue Schlussregel:

R3. Von $\vdash \varphi \Rightarrow E(\psi \wedge \varphi)$ schliesse auf $\vdash \varphi \Rightarrow C\psi$ (Induktionsregel)

Das in diesem Zusammenhang betrachtete System $\mathrm{S5}_n^C$ besteht somit aus den Axiomen **P, K, T, 4, 5, 7, 8**[1] und den Schlussregeln **R1, R2, R3**.

Korollar 2.2.10. *Für alle Formeln $\varphi \in \mathcal{L}_n^C(\Phi)$ gilt:*

$$C\varphi \equiv E(\varphi \wedge C\varphi)$$

[1] In der Literatur bezeichnet Axiom **6** die Aussage $\neg K_i(\mathit{false})$. Der Verzicht auf eine Umbenennung verursacht das Diskontinuum in der Auzählung.

Beweis. Die Implikation $C\varphi \Rightarrow E(\varphi \wedge C\varphi)$ folgt mit Axiom **8**. Andererseits gilt mit der Äquivalenz **7** $\vdash E(\varphi \wedge C\varphi) \Rightarrow E\varphi \wedge EC\varphi$. Erneute Anwendung von Axiom **8** liefert $\vdash E(\varphi \wedge C\varphi) \Rightarrow E\varphi \wedge E(C\varphi \Rightarrow E(\varphi \wedge C\varphi))$ und $\vdash E(\varphi \wedge C\varphi) \Rightarrow E\varphi \wedge E^2(\phi \wedge C\varphi)$, damit auch $\vdash E(\varphi \wedge C\varphi) \Rightarrow E(E(\varphi \wedge C\varphi) \wedge \varphi)$. Mit der Schlussregel **R3** folgt nun $\vdash E(\varphi \wedge C\varphi) \Rightarrow C\varphi$. q.e.d.

Bemerkung 2.2.11. Die Äquivalenzen $E\varphi \equiv K_1\varphi \wedge \cdots \wedge K_n\varphi$ und $C\varphi \equiv E(\varphi \wedge C\varphi)$ klären nun auch die mysteriöse Definition der Subformeln $Sub_C(\varphi)$ in 2.2.3 auf. $E\varphi$ als Abkürzung für $K_1\varphi \wedge \cdots \wedge K_n\varphi$ enthält natürlich $K_i\varphi$ als Subformel, für alle $i \in \{1, \ldots, n\}$. Analoge Überlegungen gelten für $C\varphi$.

2.2.3 Die Vollständigkeit von $S5_n^C$

Theorem 2.2.12.

a) K_n^C *ist ein vollständiges und korrektes Axiomensystem bezüglich* $\mathcal{M}_n$.

b) T_n^C *ist ein vollständiges und korrektes Axiomensystem bezüglich* $\mathcal{M}_n^r$.

c) $S4_n^C$ *ist ein vollständiges und korrektes Axiomensystem bezüglich* $\mathcal{M}_n^{rt}$.

d) $S5_n^C$ *ist ein vollständiges und korrektes Axiomensystem bezüglich* $\mathcal{M}_n^{rst}$.

Beweis. Ich beschränke mich auch hier auf den Fall von $S5_n^C$.

Korrektheit: Der Beweis verläuft wie so oft induktiv über die Beweislänge n. Unter Berücksichtigung des Resultates von 1.5.2 bleiben nur die in 2.2.2 neu eingeführten Axiome und Schlussregeln zu überprüfen.

- $n = 1$:
  - $\varphi \equiv (E\varphi \equiv K_1\varphi \wedge \cdots \wedge K_n)$ Die Gültigkeit von φ folgt unmittelbar aus Definition 2.2.7.
  - $\varphi \equiv (C\varphi \Rightarrow E(\varphi \wedge C\varphi))$. Es sei $(M, s) \models C\varphi$. Mit Lemma 2.2.9 folgt $(M, t) \models \varphi$ für alle von s erreichbaren t. Der Zustand u sei von s in einem Schritt erreichbar. Sicher gilt $(M, u) \models \varphi$. Da jeder von u erreichbare Zustand auch von s erreichbar ist, gilt $(M, t) \models \varphi$ für alle von u erreichbaren t. Also folgt $(M, u) \models \varphi \wedge C\varphi$ für alle von s in einem Schritt erreichbaren u und daher $(M, s) \models E(\varphi \wedge C\varphi)$.
- $n > 1$:
  Auf φ wurde mit der Regel **R3** geschlossen. Somit hat φ die Form $\chi \Rightarrow C\psi$. Es existiert eine Beweiszeile $\varphi_i, i < n$, mit $\varphi_i \equiv (\chi \Rightarrow E(\psi \Rightarrow \chi))$.

Mit der Induktionsvoraussetzung folgt $\mathcal{M}_n^{rst} \models \chi \Rightarrow E(\psi \wedge \chi)$. Es bleibt $\mathcal{M}_n^{rst} \models \varphi$ zu zeigen. Gelte das Antezedens von φ, d.h. $(M, s) \models \chi$. Ich zeige mittels Induktion über k, dass für alle k gilt: $(M, t) \models \psi \wedge \chi$ für alle von s in k Schritten erreichbaren t. Sei t von s in einem Schritt erreichbar. Da $\mathcal{M}_n^{rst} \models \chi \Rightarrow E(\psi \wedge \chi)$, gilt $(M, s) \models E(\psi \wedge \chi)$. Nun ist t von s in einem Schritt erreichbar, mit Lemma 2.2.9 gilt folglich $(M, t) \models \psi \wedge \chi$. Sei $k = k' + 1$. Somit existiert ein von s in k' Schritten erreichbarer Zustand t', so dass t in einem Schritt von t' ereichbar ist. Mit der Induktionsvoraussetzung gilt $(M, t') \models \psi \wedge \chi$. Dieselbe Argumentation wie bei der Induktionsverankerung liefert nun $(M, t) \models \psi \wedge \chi$. Somit gilt $(M, t) \models \psi \wedge \chi$ und natürlich auch $(M, t) \models \psi$ für alle von s erreichbaren t. Also $(M, s) \models C\psi$ und da s und M beliebig gewählt wurden, auch $\mathcal{M}_n^{rst} \models C\psi$.

Vollständigkeit: Der Beweis orientiert sich an den Ausführungen zum Theorem 1.7.4 auf Seite 17. Wieder ist die Erfüllbarkeit jeder $\mathrm{S5}_n^C$-konsistenten Formel das Ziel. Zu diesem Zwecke wird erneut eine kanonische Struktur M_φ konstruiert.

Sei $Sub_C(\varphi)$ wie in 2.2.3 definiert. Weiter seien

$$Sub_C^+(\varphi) \doteq \{\chi : \chi \in Sub_C(\varphi) \text{ oder } \neg\chi \in Sub_C(\varphi)\}$$

und

$$Con_C(\varphi) \doteq \{V : V \subseteq Sub_C^+(\varphi) \text{ und } V \text{ maximal } \mathrm{S5}_n^C\text{-konsistent}\}.$$

$M_\varphi = (S_\varphi, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ wird folgendermassen definiert:

$$\begin{aligned} S_\varphi &\doteq \{V : V \in Con_C(\varphi)\} \\ \pi_V(p) &\doteq \begin{cases} \textbf{true} \text{ falls } p \in V \\ \textbf{false} \text{ falls } p \notin V\} \end{cases} \\ \mathcal{K}_i &\doteq \{(V, W) : V/K_i = W/K_i\} \end{aligned}$$

Lemma 2.2.5 erzwingt $|S_\varphi| \leq 2^{(n+3)|\varphi|}$, die Struktur M_φ ist somit endlich. Wie im Beweis von Theorem 1.7.4 möchte ich wieder zeigen, dass für alle $\psi \in Sub_C^+(\varphi)$ und maximal $\mathrm{S5}_n^C$-konsistenten Mengen V gilt:

$$(M_\varphi, V) \models \psi \text{ gdw } \psi \in V$$

Der Beweis erfolgt wieder über den Aufbau von ψ.

- $\psi \equiv p$ oder $\psi \equiv \chi \wedge \chi'$ oder $\psi \equiv \neg\chi$ oder $\psi \equiv K_i\xi$: Diese Beweise verlaufen nahezu identisch wie in Theorem 1.7.4. Die Änderungen an den Argumenten sind sehr gering, ihre Wiederholung ist nicht notwendig.

- $\psi \equiv E\chi$. Ist $E\chi \in Sub_C^+(\varphi)$, dann auch $K_i\chi \in Sub_C^+(\varphi)$, für alle $i \in \{1, \ldots, n\}$. Für $V \in Con_C(\varphi)$ gilt: $E\chi \in V$ gdw $K_i\chi \in V$ für alle $i \in \{1, \ldots, n\}$ (mit Axiom **7**). In Lemma 1.7.4 wurde andererseits $K_i\chi \in V$ gdw $(M_\varphi, V) \models K_i\chi$ bewiesen. Somit ergibt sich: $E\chi \in V$ gdw $(M_\varphi, V) \models K_1\chi \wedge \cdots \wedge K_n\chi$, was nichts anderes bedeutet als $(M_\varphi, V) \models E\chi$.

- $\psi \equiv C\chi$. Sei $C\chi \in V$. Ich zeige mittels Induktion über k, dass für alle k gilt: Ist W von V in k Schritten erreichbar, dann $\chi \in W$ und $C\chi \in W$. Sei $k = 1$. Aus $C\chi \in V$ folgt mit **8** und $V \in Con_C(\varphi)$ sofort $E(\chi \wedge C\chi) \in V$. Ist nun W in einem Schritt von V erreichbar, d.h. $(V, W) \in \mathcal{K}_i$ für ein $i \in \{1, \ldots, n\}$, dann $\chi \wedge C\chi \in W$. Da zudem $W \in Con_C(\varphi)$, folgt natürlich $\chi \in W$ und $C\chi \in W$. Sei nun $k = k' + 1$ und W sei von V in k Schritten erreichbar. Es existiert dann ein W' so, dass W' von V in k' Schritten, W von W' in einem Schritt erreichbar sind. Mit der Induktionsvoraussetzung folgt $\chi \in W'$ und $C\chi \in W'$. Die Argumentation der Induktionsverankerung liefert nun $\chi \in W$ und $C\chi \in W$. Damit gilt $\chi \in W$ für alle W die von V erreichbar sind. Mit der Hauptinduktionsvoraussetzung folgt $(M_\varphi, W) \models \chi$ für alle von V erreichbaren W. Also folgt $(M_\varphi, V) \models C\chi$.

  Die Umkehrung erweist sich als „pièce de résistance.“ Gelte $(M_\varphi, V) \models C\chi$. Für jede Menge $W \in Con_c(\varphi)$ bezeichne φ_W die Konjunktion der Formeln in W. Weiter sei $\mathcal{W} = \{W \in Con_C(\varphi) : (M_\varphi, W) \models C\chi\}$ und $\varphi_\mathcal{W} = \bigvee_{W \in \mathcal{W}} \varphi_W$. Da $Con_C(\varphi)$ eine endliche Menge ist, gilt $\varphi_\mathcal{W} \in \mathcal{L}_n^C(\Phi)$. Es ist nun klar, weshalb die Einschränkung des Universums auf $Sub_C^+(\varphi)$ nötig ist.

  Angenommen, ich könnte

$$\mathrm{S5}_n^C \vdash \varphi_\mathcal{W} \Rightarrow E(\chi \wedge \varphi_\mathcal{W}) \tag{1}$$

  beweisen, dann folgte mit der Induktionsregel **R3** unmittelbar

$$\mathrm{S5}_n^C \vdash \varphi_\mathcal{W} \Rightarrow C\chi.$$

  Da nun $V \in \mathcal{W}$, gilt

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow \varphi_\mathcal{W}$$

  und daher

$$\mathrm{S5}_n^C \vdash \varphi_V \Rightarrow C\chi$$

Daraus folgt nun wie gewünscht $C\chi \in V$. Wäre andernfalls $C\chi \notin V$, dann $\neg C\chi \in V$, im Widerspruch zum obigen Resultat.

Es bleibt somit (1) zu zeigen. Der Leser besorge sich vor der Lektüre der nachfolgenden Zeilen am besten eine Tasse starken Kaffees, zwischen den einzelnen Häppchen des Beweises wird eine Stärkung willkommen sein.

a) Ich zeige erst für jede Menge $W \in \mathcal{W}$ und $i \in \{1, \ldots, n\}$

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i\chi.$$

Da $W \in \mathcal{W}$, gilt $(M_\varphi, W) \models C\chi$. Mit Axiom **8** folgt $(M_\varphi, W) \models E\chi$. Die Induktionsvoraussetzung liefert $E\chi \in W$. Nun kommt Axiom **7** ins Spiel und erzwingt $K_i\chi \in W$, für alle $i \in \{1, \ldots, n\}$. Dies ergibt aber sicher $\mathrm{S5}_n \vdash \varphi_W \Rightarrow K_i\chi$.

b) Im zweiten Schritt zeige ich für alle $i \in \{1, \ldots, n\}$, $W \in \mathcal{W}$ und $W' \notin \mathcal{W}$

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i\neg\varphi_{W'}.$$

Da $W \in \mathcal{W}$ und $W' \notin \mathcal{W}$ gilt $(M_\varphi, W) \models C\chi$ und $(M_\varphi, W') \not\models C\chi$. Es gibt somit ein von W' erreichbares Z mit $(M_\varphi, Z) \models \neg\chi$. Damit darf W' nicht von W erreichbar sein, da sonst auch Z von W erreichbar wäre. D.h. $(W, W') \notin \mathcal{K}_i$ für alle $i \in \{1, \ldots, n\}$ und nach Definition der $\mathcal{K}_i$ somit $W/K_i \neq W'/K_i$, für alle $i \in \{1, \ldots, n\}$. Es gibt daher zu jeden $i \in \{1, \ldots, n\}$ eine Formel ξ_i mit $K_i\xi_i \in W$ und $K_i\xi_i \notin W'$ oder umgekehrt. Sei vorerst $K_i\xi_i \in W$ und $K_i\xi_i \notin W'$. Da $W' \in Con_C(\varphi)$ folgt $\neg K_i\xi_i \in W'$ und damit

$$\mathrm{S5}_n^C \vdash \varphi_{W'} \Rightarrow \neg K_i\xi_i, \text{resp. } \mathrm{S5}_n^C \vdash K_i\xi_i \Rightarrow \neg\varphi_{W'}.$$

Mit **R2** folgt

$$\mathrm{S5}_n^C \vdash K_i(K_i\xi_i \Rightarrow \neg\varphi_{W'})$$

und mit **K**, **4** und **R1** somit

$$\mathrm{S5}_n^C \vdash K_i\xi_i \Rightarrow K_i\neg\varphi_{W'}.$$

Nun ist $K_i\xi_i \in W$, also

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i\neg\varphi_{W'}$$

Gilt nun andererseits $K_i\xi_i \notin W$ und $K_i\xi_i \in W'$, so folgt

$$\mathrm{S5}_n^C \vdash \varphi_{W'} \Rightarrow K_i\xi_i, \text{ resp. } \mathrm{S5}_n^C \vdash \neg K_i\xi_i \Rightarrow \neg\varphi_{W'}$$

Axiom **2** liefert

$$\mathrm{S5}_n^C \vdash K_i(\neg K_i\xi_i \Rightarrow \neg\varphi_{W'}),$$

mit **K**, **5** und **R1** folgt

$$\mathrm{S5}_n^C \vdash \neg K_i\xi_i \Rightarrow K_i\neg\varphi_{W'}$$

Da $\neg K_i\xi_i \in W$ ergibt sich

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i\neg\varphi_{W'}.$$

c) Sicher gilt auch für belieblige $\alpha, \beta \in \mathcal{L}_n^C(\Phi)$ und $i \in \{1, \ldots, n\}$

$$\mathrm{S5}_n^C \vdash (K_i\alpha \wedge K_i\beta) \Rightarrow K_i(\alpha \wedge \beta),$$

denn zusammen mit den Tautologien

$$\begin{aligned}\mathrm{S5}_n^C &\vdash (K_i\alpha \wedge K_i\beta) \Rightarrow K_i\alpha\\ \mathrm{S5}_n^C &\vdash (K_i\alpha \wedge K_i\beta) \Rightarrow K_i\beta\\ \mathrm{S5}_n^C &\vdash \alpha \Rightarrow (\beta \Rightarrow (\alpha \wedge \beta)),\end{aligned}$$

mit **R2**

$$\mathrm{S5}_n^C \vdash K_i(\alpha \Rightarrow (\beta \Rightarrow (\alpha \wedge \beta)))$$

und dem Kettenschluss folgt die Behauptung

$$\mathrm{S5}_n^C \vdash (K_i\alpha \wedge K_i\beta) \Rightarrow K_i(\alpha \wedge \beta).$$

d) So ergibt sich nun mit (a),(b) und (c) problemlos

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i(\chi \wedge (\bigwedge_{W' \notin \mathcal{W}} \neg\varphi_{W\prime})). \tag{2}$$

e) Weiter beweise ich

$$\mathrm{S5}_n^C \vdash \bigvee_{W \in Con_C(\varphi)} \varphi_W.$$

Gegenannahme: Sei $\bigvee_{W \in Con_C(\varphi)} \varphi_W$ aus $\mathrm{S5}_n^C$ nicht beweisbar. Dann ist $\neg(\bigvee_{W \in Con_C(\varphi)} \varphi_W)$ $\mathrm{S5}_n^C$-konsistent. Es gibt somit eine maximal $\mathrm{S5}_n^C$-konsistente Menge T, welche $\{\neg \bigvee_{W \in Con_C(\varphi)} \varphi_W\}$ umfasst. Da T u. a. eine konsistente Menge ist, gilt für jede Formel $\xi \in Sub_C^+(\varphi)$: $\xi \in T$ oder $\neg\xi \in T$. Sei $W_T = Sub_C^+ \cap T$. Natürlich gilt $W_T \in Con_C(\varphi)$ und $\varphi_{W_T} \in T$. Aussagenlogisches Räsonieren liefert

$$\mathrm{S5}_n^C \vdash \varphi_{W_T} \Rightarrow \bigvee_{W \in Con_C(\varphi)} \varphi_W.$$

Dann folgt aber $\bigvee_{W \in Con_C(\varphi)} \varphi_W \in T$, was der Voraussetzung $\{\neg \bigvee_{W \in Con_C(\varphi)} \varphi_W\} \subseteq T$ widerspricht. $\bigvee_{W \in Con_C(\varphi)} \varphi_W$ ist demnach beweisbar aus $\mathrm{S5}_n^C$.

f) Die folgende Äquivalenz ist leicht einzusehen:

$$\mathrm{S5}_n^C \vdash \bigvee_{W \in Con_C(\varphi)} \varphi_W \equiv \bigvee_{W \in \mathcal{W}} \varphi_W \vee \bigvee_{W' \notin \mathcal{W}} \varphi_{W'}$$

Offensichtlich ist $\varphi_{\mathcal{W}} \equiv \bigvee_{W \in \mathcal{W}} \varphi_W$. Da für jede Menge $U \in Con_C(\varphi)$ gilt: Entweder $U \in \mathcal{W}$ oder $U \notin \mathcal{W}$, folgt

$$\mathrm{S5}_n^C \vdash \varphi_{\mathcal{W}} \equiv \neg(\bigvee_{W' \notin \mathcal{W}} \varphi_{W'}). \tag{3}$$

g) Bereits bewiesen wurde (2):

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i(\chi \wedge (\bigwedge_{W' \notin \mathcal{W}} \neg\varphi_{W'})).$$

Eine kleine Umformung ergibt:

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i(\chi \wedge \neg(\bigvee_{W' \notin \mathcal{W}} \varphi_{W'}))$$

und ebenso

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i\chi \wedge K_i\neg(\bigvee_{W' \notin \mathcal{W}} \varphi_{W'}).$$

Aus (3) und **R2** folgt

$$\mathrm{S5}_n^C \vdash K_i\varphi_{\mathcal{W}} \equiv K_i\neg(\bigvee_{W' \notin \mathcal{W}} \varphi_{W'}).$$

Alles in allem ergibt sich für alle $W \in \mathcal{W}$ und $i \in \{1, \ldots, n\}$

$$\mathrm{S5}_n^C \vdash \varphi_W \Rightarrow K_i(\chi \wedge \varphi_{\mathcal{W}}) \tag{4}$$

h) Zur Erinnerung: $\varphi_{\mathcal{W}} = \bigvee_{w \in \mathcal{W}} \varphi_W$. Das Resultat aus (4) liefert

$$\mathrm{S5}_n^C \vdash \bigvee_{W \in \mathcal{W}} \varphi_W \Rightarrow K_1(\chi \wedge \varphi_{\mathcal{W}}) \wedge \cdots \wedge K_n(\chi \wedge \varphi_{\mathcal{W}}),$$

was nicht anderes bedeutet als

$$\mathrm{S5}_n^C \vdash \varphi_{\mathcal{W}} \Rightarrow E(\chi \wedge \varphi_W)$$

und (1) ist damit bewiesen.

Somit gilt:

$$(M_\varphi, V) \models \psi \text{ gdw } \psi \in V$$

Da die Möglichkeitsrelationen $\mathcal{K}_i$ wie in 1.6.1 definiert wurden, gilt selbstverständlich $M_\varphi \in \mathcal{M}_n^{rst}$. q.e.d.

Nach diesem harten Stück Arbeit empfehle ich zur Erholung ein Glas 18-jährigen Glenmorrangie, zwei Finger breit und ohne Eis. Cheers.

Kapitel 3

Komplexitätstheoretisches Rüstzeug

3.1 Komplexitätstheoretische Begriffe

Aus den Kapiteln 1 und 2 folgt die Entscheidbarkeit der in 1.4 und 2.2.2 vorgestellten Wissenslogiken. Damit kann für jede Formel $\varphi \in \mathcal{L}_n(\Phi)$ geprüft werden, ob sie in $\mathcal{M}_n^{rst}$ erfüllbar ist oder nicht. Soll dieser Test algorithmisch durchgeführt werden, so stellt sich natürlich die Frage nach den dazu nötigen Ressourcen. Die Komplexitätstheorie bietet die Möglichkeit, diesen Aufwand messbar zu machen. Da ich beabsichtige, das Erfüllbarkeitsproblem der Logiken $\mathrm{S5}_n$ und $\mathrm{S5}_n^C$ in dieser Arbeit etwas genauer unter die Lupe zu nehmen, sollen in diesem Kapitel einige Grundbegriffe der Komplexitätstheorie zusammengefasst werden. Auf eine umfangreiche Einführung möchte ich aus Platzgründen verzichten, der Leser halte sich z. B. an [BC94] oder [GarJo79].

Im Mittelpunkt der Komplexitätstheorie steht die Entscheidung, ob ein Objekt Element einer bestimmten Menge ist oder nicht. Für den vorliegenden Fall bedeutet dies die Frage nach der Zugehörigkeit einer Formel $\varphi \in \mathcal{L}_n(\Phi)$ zur Menge aller bzgl. einer Klasse von Strukturen gültigen Formeln. Fragestellungen dieser Art werden auch Entscheidungsprobleme genannt. Zur Berechnung der Antwort wird ein gewisser Umfang an Rechenzeit und Speicherplatz notwendig sein, welcher mit der Eingabegrösse variert. Als Mass der Eingabegrösse dient die Formellänge $|\varphi|$. Die benötigten Ressourcen Zeit und Platz werden demnach als Funktionen von $|\varphi|$ betrachtet. An ihnen wird die Schwierigkeit, resp. die Komplexität, des untersuchten Entscheidungsproblems gemessen.

Ist in einem beliebigen Punkt einer Berechnung der Nachfolgeschritt eindeutig festgelegt, so spricht man von einem deterministischen Algorithmus.

Hingegen besitzt ein nicht-deterministisches Programm die Möglichkeit, den auf einen beliebigen Punkt in einer Berechnung folgende Schritt aus einer endlichen Anzahl von Alternativen auszuwählen. Eine deterministische Berechnung wird entweder mit einem Akzeptieren der Eingabe (d. h. das Objekt gehört zur betrachteten Menge) oder deren Verwerfung enden (d. h. das Objekt gehört nicht zur betrachteten Menge). Andererseits akzeptiert eine nicht-deterministische Berechnung die Eingabe genau dann, wenn eine passende Folge von Alternativen zur Akzeptanz führt.

Die in dieser Arbeit vorkommenden Komplexitätsklassen sind P, NP, PSPACE und EXPTIME. P bezeichnet dabei die Klasse aller Entscheidungsprobleme, für die es einen polynomialen Lösungsalgorithmus gibt, d. h. die Zeit, welche der Algorithmus zum Auffinden der Lösung benötigt, wächst polynomial mit der Eingabegrösse. Die Klasse NP umfasst alle Entscheidungsprobleme, die mit einem nicht-deterministischen, polynomialen Algorithmus gelöst werden können. Für diese Probleme ist es in polynomial vielen Schritten möglich, eine positive Lösung hinzuschreiben und anschliessend zu verifizieren. Variert bei einer deterministischen Berechung der Lösung eines Entscheidungsproblems der Speicherplatz polynomial mit der Eingabegrösse, so liegt das Problem in der Klasse PSPACE. Schliesslich bezeichnet EXPTIME die Klasse all jener Entscheidungsprobleme, deren Lösungen mit einer deterministischen Berechnung in exponentieller Zeit gefunden werden können[1]. Ohne grösseren Aufwand lassen sich $\mathrm{P} \subseteq \mathrm{NP} \subseteq \mathrm{PSPACE} \subseteq \mathrm{EXPTIME}$ und $\mathrm{P} \neq \mathrm{EXPTIME}$ zeigen. Zwar wird vermutet, dass auch die andern Inklusionen strikt sind, doch fehlt bis dato der Beweis. Besonders das P = NP-Problem kam zu einiger Berühmtheit, beträgt doch das Preisgeld zu seiner korrekten Lösung nicht weniger als 1 Million U. S.-Dollars. Wie verzwickt die Angelegenheit ist, zeigt sich im Fall der polynomialen Platzbeschränkung: Nach dem Satz von Savitch gilt PSPACE = *NPSPACE*. Der Nicht-Determinismus fügt hier der Berechnung keine zusätzliche Leistungsstärke bei.

Für eine beliebige Komplexitätsklasse $\mathcal{C}$ bezeichnet die Klasse co-$\mathcal{C}$ diejenigen Mengen, deren Komplement in $\mathcal{C}$ liegt. Ist nun $\mathbf{A}$ ein deterministischer Algorithmus, der die Zugehörigkeit zur Menge A entscheidet, so lässt sich aus $\mathbf{A}$ leicht ein Algorithmus $\mathbf{A}'$ gewinnen, der das Komplement von A in denselben Platz-und Zeitschranken entscheidet: $\mathbf{A}'$ akzeptiert x genau dann, wenn $\mathbf{A}$ die Eingabe x verwirft. Folglich gilt $\mathcal{C} = \text{co-}\mathcal{C}$ für jede deterministische Komplexitätsklasse $\mathcal{C}$. Für den Fall eines nicht-deterministischen Algorithmus' $\mathbf{B}$ sieht die Sache anders aus. $\mathbf{B}$ entscheidet die Zugehörigkeit

[1]Exponentiell in n heisst hier 2^{cn} für eine Konstante $c > 0$, in der Terminologie von [BC94] bezeichnet dies demnach die Klasse *LEXP*.

von x zu B aufgrund einer passenden Wahl von Alternativen. Wie soll aus $\mathbf{B}$ ein Algorithmus $\mathbf{B}'$ konstruiert werden, der x genau dann akzeptiert, wenn x von $\mathbf{B}$ verworfen wird? Diese Frage bleibt bisanhin unbeantwortet. So ist auch nicht klar, ob NP = co-NP gilt. Wäre P = NP, so folgte offensichtlich NP = co-NP. Vermutet wird aber NP $\neq$ co-NP.

Ein Entscheidungsproblem, resp. eine Menge A heisst *hart* bezüglich einer Komplexitätsklasse $\mathcal{C}$ (geschrieben als $\mathcal{C}$-*hart*), falls jedes Problem aus $\mathcal{C}$ in polynomialer Zeit auf A reduziert werden kann. Mit andern Worten, jedes $B \in \mathcal{C}$ kann mit der Funktion f aus P so auf A reduziert werden, dass $x \in B$ genau dann gilt, wenn $f(x) \in A$. Ist A hart bzgl. der Komplexitätsklasse $\mathcal{C}$ und gilt zusätzlich $A \in \mathcal{C}$, so heisst A *vollständig* bzgl. der Klasse $\mathcal{C}$, geschrieben als $\mathcal{C}$-*vollständig*.

Ich beende dieses informelle Kapitel mit drei Beispielen, die im weiteren Text Erwähnung finden.

3.2 Beispiele von vollständigen Problemen

3.2.1 NP-Vollständigkeit

Beispiel 3.2.1 (SAT). Gegeben sei eine aussagenlogische Formel $\varphi(x_1, \ldots, x_n)$ in den Variablen $x_1, \ldots, x_n$. Können die Variablen so mit Werten aus $\{\mathbf{true}, \mathbf{false}\}$ belegt werden, dass $\varphi \equiv \mathbf{true}$ gilt?

Der Nachweis der NP-Vollständigkeit von SAT gelang Cook [Co71] im Jahre 1971, als erstes NP-vollständiges Problem überhaupt.

Bekanntlich ist eine Formel φ genau dann gültig, wenn $\neg\varphi$ nicht erfüllbar ist. Das Gültigkeitsproblem der Aussagenlogik ist demnach co-NP-vollständig.

3.2.2 PSPACE-Vollständigkeit

Definition 3.2.2. Eine *quantifizierte Boolsche Formel* ist ein Ausdruck der Form

$$Q_1 v_1 Q_2 v_2 \ldots Q_m v_m A,$$

wobei A eine aussagenlogische Formel in den Variablen $v_1, \ldots, v_m$ ist und $Q_i \in \{\exists, \forall\}$ für $1 \leq i \leq m$ gilt.

Der Wert einer quantifizierten Boolschen Formel (QBF) wird bestimmt, indem man jeden Teilausdruck der Form $\exists x(E)$ durch $E_0 \vee E_1$ und jeden Teilausdruck der Form $\forall x(E)$ durch $E_0 \wedge E_1$ ersetzt, wobei E_0 bzw. E_1 gleich

E ist mit allen Auftreten von x im Bereich der Quantoren durch **true** bzw. **false** ersetzt.

Beispiel 3.2.3 (QBF). Sei $A = Q_1 v_1 \dots Q_m v_m A'$ eine quantifizierte Boolsche Formel in den Variablen $v_1, \dots, v_m$. Hat A den Wert **true**?

Stockmeyer und Meyer [SM73] gelang im Jahre 1973 der Nachweis der PSPACE-Vollständigkeit von QBF.

3.2.3 EXPTIME-Vollständigkeit

Als Beispiel einer EXPTIME-vollständigen Sprache soll das Erfüllbarkeitsproblem der Logik K^C dienen. Die Präsentation benötigt einige Vorbemerkungen.

Natürlich können Algorithmus, Komplexität und $\mathcal{C}$-Vollständigkeit mathematisch-präzise behandelt werden. Als formales Hilfsmittel eignet sich oft ein gedachtes, besonders einfaches Modell eines Computers, das *Turing-Maschine* genannt wird. Bezeichne nun DTIME$[t(n)]$ die von deterministischen Turnigmaschinen in der Zeit $\mathcal{O}(t(n))$, ASPACE$[s(n)]$ die von alternierenden Turingmaschinen in der Platzbeschränkung $\mathcal{O}(s(n))$ entscheidbaren Sprachen. Die genaue Bedeutung dieser Bezeichnungen wird im Verlauf dieses Kapitels erläutert. Nach einem Ergebnis von Chandra, Kozen und Stockmeyer [CKS81] gilt nun

$$\text{ASPACE}[s(n)] = \bigcup_{c>1} \text{DTIME}[c^{s(n)}].$$

Eine einfache Abänderung des Beweises von [FL79] zeigt, dass die Berechnung einer alternierenden Turingmaschine auf einer Eingabe x durch eine Formel aus K^C beschrieben werden kann. Damit ist SATK^C offensichtlich EXPTIME-hart.[2] Die obere EXPTIME-Schranke folgt daraufhin mit einer Modellkonstruktion nach Pratt [Pra79].

Alternierende Turnigmaschinen zählen nicht unbedingt zum Stoff, der in einer Einführungsvorlesung über Komplexitätstheorie vermittelt wird. Ich präzisiere daher die obigen Ausführungen und beweise exemplarisch, dass K^C EXPTIME-hart ist.

Definition 3.2.4. Eine einbändige alternierende Turingmaschine ist ein Tupel $AT = (Q, \Delta, \Gamma, b, \delta, q_0, U)$, mit

- einer endlichen Menge Q von Zuständen,

[2] SATK^C bezeichnet im weitern das Erfüllbarkeitsproblem der Logik K^C.

- einem Bandalphabet Γ mit $\Gamma \cap Q = \emptyset$,
- einem Eingabealphabet $\Sigma \subseteq \Gamma$,
- einem Blanksymbol $b \in \Gamma \setminus \Delta$,
- einer Übergangsrelation $\delta \subseteq (Q \times \Gamma) \times (Q \times \Gamma \times \{L, R\})$,
- einem Startzustand q_0,
- einer Menge $U \subseteq Q$ von universellen Zuständen und
- einer Menge $Q \setminus U$ von existenziellen Zuständen.

Wie üblich bezeichne Γ^* die Menge aller endlichen, Γ^+ die Menge aller endlichen und nicht-leeren Zeichenketten über Γ.

Eine *Konfiguration* ist ein Element aus $\Gamma^*Q\Gamma^+$ und repräsentiert eine Momentaufnahme in der Berechnung einer Turingmaschine. Die Zeichenkette aus Γ^* stellt den Bandinhalt links vom Schreib-/Lesekopf dar, das Element aus Γ^+ steht für die Bandinformation über und rechts vom Schreib-/Lesekopf. Die Maschine befindet sich im Zustand q aus Q und dessen Stellung in der Zeichenkette aus $\Gamma^*Q\Gamma^+$ gibt die Position des Schreib-/Lesekopfes an, was an späterer Stelle noch genauer ausgeführt wird. Eine *universelle Konfiguration* ist ein Element aus $\Gamma^*U\Gamma^+$, eine *existenzielle Konfiguration* entsprechend ein Mitglied von $\Gamma^*(Q \setminus U)\Gamma^+$.

Definition 3.2.5. Sei $\alpha = xq\sigma y$ eine Konfiguration mit $\sigma \in \Gamma, x, y \in \Gamma^*$ und $q \in Q$.

a) $\mathit{tape}(\alpha) = x\sigma y$,

b) $\mathit{pos}(\alpha) = l(x) + 1$, wobei $l(x)$ die Länge von x über Γ bezeichnet,

c) $\mathit{state}(\alpha) = q$.

Definition 3.2.6. Seien $\alpha = xq\sigma y$ und $\beta = x'q'\sigma'y'$ Konfigurationen mit $\sigma, \sigma' \in \Gamma, x, x', y, y' \in \Gamma^*$ und $q, q' \in Q$. β ist eine *nächste Konfiguration* von α, falls für ein $\tau \in \Gamma$ entweder

a) $(q, \sigma, q', \tau, L) \in \delta, x'\sigma' = x$ und $y' = \tau y$ oder

b) $(q, \sigma, q', \tau, R) \in \delta, x' = x\tau$ und $\sigma'y' = y$ oder $(y = y'$ und $\sigma' = b)$ gilt.

Eine *Berechnungssequenz* bezeichnet eine Folge von Konfigurationen $\alpha_1, \ldots, \alpha_k$, wobei α_{i+1} eine nächste Konfiguration von α_i ist, mit $1 \leq i < k$.

Definition 3.2.7. Eine *Spur* von AT ist eine Menge C von Paaren der Form (α, t), mit einer Konfiguration α und $t \in \mathbb{N}$, so dass gilt:

a) Falls $(\alpha, t) \in C$ und $state(\alpha) \in U$, dann existiert zu jeder nächsten Konfiguration β von α ein $t' < t$ mit $(\beta, t') \in C$.

b) Falls $(\alpha, t) \in C$ und $state(\alpha) \in (Q \setminus U)$, dann existiert eine nächste Konfiguration β von α und ein $t' < t$ mit $(\beta, t') \in C$.

Definition 3.2.8. Sei AT eine alternierende Turingmaschine. Die von AT akzeptierte Sprache ist gegeben durch

$$L(AT) \doteq \{x \in \Delta^* : (q_0 x, t) \in C \text{ für ein } t \in \mathbb{N} \text{ und eine Spur } C \text{ von } AT\}.$$

Die verschiedenen Berechnungssequenzen, welche auf die Eingabe x möglich sind, lassen sich als Graph visualisieren. Die Knoten des Graphen korrespondieren zu Konfigurationen, die Kanten verbinden je zwei Konfigurationen, falls die eine nächste Konfiguration der andern ist. Der entstehende Graph wird auch Berechnungsbaum genannt. Die Startkonfiguration $q_0 x$ bildet die Wurzel des Baumes. Jeder mit dieser Wurzel beginnende (endliche oder unendliche) Pfad repräsentiert eine Berechnungssequenz. Sicher ist auch der Verzweigungsgrad dieses Baumes beschränkt, da die Übergangsrelation δ selbst endlich ist.

Nach der Definition 3.2.8 liegt x genau dann in $L(AT)$, wenn es eine endliche, mit $q_0 x$ beginnende Berechnungssequenz gibt. Endliche Berechnungssequenzen enden gemäss der Definition 3.2.7 in universellen Konfigurationen. Eine derartige Sequenz $\alpha_1, \ldots, \alpha_k$ muss im Falle einer alternierenden Turingmaschine jedoch weitere Bedingungen erfüllen. Ist α_i eine universelle Konfiguration, so müssen alle mit α_i beginnenden Pfade akzeptierende Berechnungssequenzen darstellen (für den vorliegenden Fall bedeutet dies die Endlichkeit der Sequenz). Ist α_i hingegen eine existentielle Konfiguration, so muss lediglich ein mit α_i startender Pfad endlicher Länge existieren. Bild 2 zeigt ein einfaches Beispiel eines möglichen Berechungsbaumes einer alternierenden Turingmaschine. Ein mit der Wurzel beginnender akzeptierender Pfad wird in diesem Fall auch alternierender Baum genannt. $\forall$ steht für eine universelle, $\exists$ für eine existentielle Konfiguration, die Kanten des alternierenden Baumes sind markiert.

Eine Spur C heisst *c-platzbeschränkt*, falls α für alle $(\alpha, t) \in C$ höchstens c Bandzellen belegt. Gibt es zu jedem $x \in L(AT)$ der Länge n ein $t \in \mathbb{N}$ und eine Spur C von AT so, dass C $s(n)$-platzbeschränkt ist, mit $s : \mathbb{N} \to \mathbb{N}$, und $(q_0 x, t) \in C$ gilt, so *operiert AT im Platz $s(n)$*.

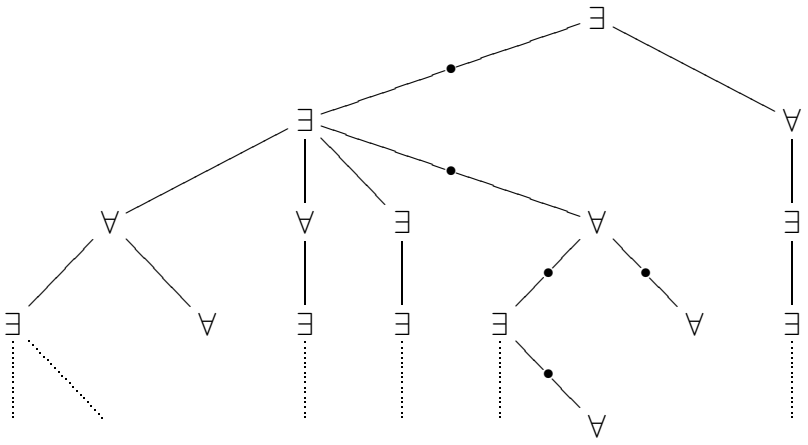


Bild 2: Beispiel eines alternierenden Baumes.

Definition 3.2.9. ASPACE[$s(n)$] bezeichne die Klasse aller Sprachen, die von im Platz $s(n)$ operierenden, alternierenden Turingmaschinen akzeptiert werden.

Operiert AT im Platz $s(n)$, so können in jeder Berechnung höchstens endlich viele verschiedene Konfigurationen auftreten. Diese Tatsache führt zu folgendem Lemma:

Lemma 3.2.10. *Sei AT eine im Platz $s(n)$ operierende Turingmaschine. Dann terminiert AT seine Berechnung auf x genau dann, wenn die Berechnung von AT auf x nach maximal $N = |\Gamma|^{s(n)} \cdot s(n) \cdot |Q|$ Schritten terminiert.*

Beweis. Offenbar bezeichnet N die Anzahl der möglichen, paarweise verschiedenen Konfigurationen. Beendet AT nach maximal N Schritten seine Berechnung nicht, so befindet sich die Maschine offenbar in einer Endlosschlaufe und wird demnach nie terminieren. Die Umkehrung der Behauptung ist trivial. q.e.d.

Korollar 3.2.11. *Ist AT eine im Platz $s(n)$ operierende Turingmaschine und $s(n) \geq n$. Dann wiederholt AT keine Konfiguration.*

Beweis. Sei N wie oben definiert. Sicher findet sich ein $m \in \mathbb{N}$ mit $m \geq 2$ und $m^{s(n)} \geq N$. Die alternierende Turingmaschine AT' führt dieselben Berechnungen aus wie AT, zählt aber mit einem m-stelligen Zähler die gemachten Schritte. Nach mehr als N Schritten verwirft AT' die Eingabe. Offenbar akzeptieren AT' und AT dieselben Sprachen, zudem benötigt der Zähler nicht mehr als $s(n)$ Zellen. AT' operiert demnach ebenfalls im Platz $s(n)$. q.e.d.

Das Ausschliessen von Schleifen in der Berechnung von alternierenden Turingmaschinen ist also keine Einschränkung. Damit wird aber für die Elemente von C die zweite Komponente überflüssig.

Definition 3.2.12. Eine *vereinfachte Spur* ist eine endliche Menge D von Konfigurationen mit:

a) Falls $\alpha \in D$ und $\alpha \in U$, dann liegt jede nächste Konfiguration β von α in D.

b) Falls $\alpha \in D$ und $\alpha \in (Q \setminus U)$, dann gibt es eine nächste Konfiguration β von α in D.

Eine vereinfachte Spur D akzeptiert $x \in \Delta^*$, falls $q_0 x \in D$.

Somit ergibt sich folgendes Lemma:

Lemma 3.2.13. *Sei AT eine alternierende Turingmaschine, welche keine Konfiguration wiederholt. Dann gilt:*

$$L(AT) = \{x \in \Delta^* : q_0 x \in D \text{ für eine vereinfachte Spur von } AT\}.$$

Beweis. Sei $L' = \{x \in \Delta^* : q_0 x \in D$ für eine vereinfachte Spur von $AT\}$ und $x \in L(AT)$. Dann existiert eine Spur C von AT mit $(q_0 x, t) \in C$. Die vereinfachte Spur D entsteht aus den ersten Komponenten der Elemente von C. Damit folgt $x \in L'$. Gilt andererseits $x \in L'$, dann sei t' die maximale Länge einer Berechnungssequenz mit $\alpha_1 = q_0$. Da AT keine Konfiguration wiederholt, existiert dieses Maximum. Mit $(q_0 x, t') \in C$ folgt nun die gesuchte Spur C: Die Konfigurationen werden von D übernommen und t sinkt beim Übergang von α zu seinem Nachfolger β um eine Einheit. q.e.d.

Doch nun endlich zum lange ersehnten Theorem.

Theorem 3.2.14. *Falls $s(n) \geq n, H \subseteq \Delta^*$ und $H \in ASPACE[s(n)]$, dann existiert eine Funktion f von Δ^* nach $\mathcal{L}_n^C$ mit:*

a) $x \in H$ gdw $f(x)$ ist K^C-erfüllbar.

b) Falls $l(x) = n$, dann $|f(x)| \in \mathcal{O}(s(n))$.

c) Falls $s(n) \in P$, dann ist auch $f \in P$.

Beweis. Sei AT eine im Platz $s(n)$ operierende alternierende Turingmaschine die H akzeptiert. Weiter seien $x \in H, l(x) = n, m = s(n) + 1$ und die Bandzellen von AT seien wie üblich durchnummeriert. Die benötigten primitiven Propositionen sind:

$$\begin{array}{rll} P_{i,\sigma} & \text{mit} & 0 \leq i \leq m, \sigma \in \Gamma \\ H_i & \text{mit} & 0 \leq i \leq m \\ Q_q & \text{mit} & q \in Q \end{array}$$

Die Intuition ist klar: $P_{i,\sigma}$ bedeutet „die Zelle i enthält das Zeichen σ“, H_i steht für „der Schreib-/Lesekopf steht bei der Zelle i“ und Q_q meint „der Zustand ist q.“

Eine Belegung der primitiven Propositionen soll mit einer Konfiguration von AT korrespondieren. Die Formeln $g_1, \ldots, g_6$ zwingen die Belegungen, sich „programmgemäss“ zu verhalten.

g_1: AT befindet sich in genau einem Zustand.

$$\bigvee_{q \in Q} (Q_q \wedge \bigwedge_{q' \in Q \setminus \{q\}} \neg Q_{q'})$$

g_2: In jeder Zelle steht genau ein Zeichen.

$$\bigwedge_{i=0}^{m} \bigvee_{\sigma \in \Gamma} (P_{i,\sigma} \wedge \bigwedge_{\sigma' \in \Gamma \setminus \{\sigma\}} \neg P_{i,\sigma'})$$

g_3: Die ungelesenen Zellen bleiben unverändert.

$$\bigwedge_{i=0}^{m} \bigwedge_{\sigma \in \Gamma} (\neg H_i \wedge P_{i,\sigma} \Rightarrow K P_{i,\sigma})$$

g_4: Falls es genau eine Schreib-/Leseposition gibt, dann befindet sich die nächste Position eine Einheit links oder rechts von der aktuellen; 0 und m sind keine Schreib-/Lesepositionen.

$$\begin{aligned} &\bigwedge_{i=1}^{m-1} (H_i \Rightarrow K((H_{i-1} \wedge \neg H_{i+1}) \vee (\neg H_{i-1} \wedge H_{i+1}))) \\ &\quad \wedge (\neg H_{i-1} \wedge \neg H_{i+1} \Rightarrow K \neg H_i)) \wedge \neg H_0 \wedge \neg H_m \end{aligned}$$

g_5: Die universellen Zustände verhalten sich programmgemäss.

$$\bigwedge_{i=1}^{m-1} \bigwedge_{\sigma\in\Gamma} \bigwedge_{q\in U} \Big(H_i \wedge P_{i,\sigma} \wedge Q_q \Rightarrow \big(\bigwedge_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',R)\in\delta}} \neg K\neg(H_{i+1} \wedge P_{i,\sigma'} \wedge Q_{q'}) \wedge \bigwedge_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',L)\in\delta}} \neg K\neg(H_{i-1} \wedge P_{i,\sigma'} \wedge Q_{q'})\big)\Big)$$

g_6: Die existenziellen Zustände verhalten sich programmgemäss.

$$\bigwedge_{i=1}^{m-1} \bigwedge_{\sigma\in\Gamma} \bigwedge_{q\in Q\setminus U} \Big(H_i \wedge P_{i,\sigma} \wedge Q_q \Rightarrow \big(\bigvee_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',R)\in\delta}} \neg K\neg(H_{i+1} \wedge P_{i,\sigma'} \wedge Q_{q'}) \vee \bigvee_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',L)\in\delta}} \neg K\neg(H_{i-1} \wedge P_{i,\sigma'} \wedge Q_{q'})\big)\Big)$$

Wie üblich sind in g_5 und g_6 auftretende leere Konjunktionen und Disjunktionen als äquivalent zu **true** resp. **false** zu betrachten. Mit den Formeln g_1 bis g_6 definiert man nun

$$g \equiv \bigwedge_{i=1}^{6} g_i.$$

Sei $x = \sigma_1 \dots \sigma_n$. Die Formeln h beschreibt mit folgender Definition die Startkonfiguration:

$$h : Q_{q_0} \wedge H_1 \wedge \neg H_0 \wedge \bigwedge_{i=2}^{m} \neg H_i \wedge \bigwedge_{i=1}^{n} P_{i,\sigma_i} \wedge P_{0,b} \wedge \bigwedge_{i=n+1}^{m} P_{i,b}$$

Somit kann $f(x)$ definiert werden:

$$f(x) = h \wedge Cg.$$

Die Länge der Formel g ist offenbar linear in $s(n)$, $|h|$ wächst linear in $s(n)$ und n. Mit der Voraussetzung $s(n) \geq n$ gilt $|f(x)| \in \mathcal{O}(s(n))$, die Bedingung (b) von Theorem 3.2.14 ist also erfüllt. Ist $s(|x|)$ in polynomialer Zeit berechenbar, so auch $f(x)$, wie in Blick auf g zeigt. Bleibt noch Bedingung (a).

Sei D eine x akzeptierende, maximal $s(n)$ Zellen belegende vereinfachte

Spur von AT. Die Struktur $M = (S, \pi, \mathcal{K})$ wird folgendermassen definiert:

$$
\begin{aligned}
S &\doteq D \\
\pi_\alpha(P_{i,\sigma}) &\doteq \begin{cases} \textbf{true} & \text{falls } \mathit{tape}(\alpha)_i = \sigma \\ \textbf{false} & \text{sonst} \end{cases} \quad 0 \leq i \leq m, \sigma \in \Gamma \\
\pi_\alpha(H_i) &\doteq \begin{cases} \textbf{true} & \text{falls } \mathit{pos}(\alpha) = i \\ \textbf{false} & \text{sonst} \end{cases} \quad 0 \leq i \leq m \\
\pi_\alpha(Q_q) &\doteq \begin{cases} \textbf{true} & \text{falls } \mathit{state}(\alpha) = q \\ \textbf{false} & \text{sonst} \end{cases} \quad q \in Q \\
\mathcal{K} &\doteq \{(\alpha, \beta) : \beta \text{ ist eine nächste Konfiguration von } \alpha\}
\end{aligned}
$$

Dabei ist $\mathit{tape}(\alpha)_i$ das i-te Symbol von $\mathit{tape}(\alpha)$ falls $1 \leq i \leq l(\mathit{tape}(\alpha))$, und b sonst. Ich denke, aus der Konstruktion von g geht hervor, dass $(M, q_0x) \models f(x)$, falls $x \in H$.

Sei andererseits $f(x)$ in der Struktur $M = (S, \pi, \mathcal{K})$ erfüllbar, d. h. $(M, s_0) \models f(x)$ für ein $s_0 \in S$. Mit dem Beweis zum Theorem 2.2.12 darf die Endlichkeit von M angenommen werden. Weiter gilt nach den Ausführungen zur Definition 2.2.8, dass g in allen von s_0 erreichbaren Zuständen t gilt, schliesslich gilt h bei s_0. Damit beschreiben die bei jedem von s_0 erreichbaren Zustand t wahren primitiven Propositionen in natürlicher Art, erzwungen durch die Formeln g_1-g_4, eine eindeutige Konfiguration, ich nenne sie $\alpha(t)$. Doch Vorsicht ist angebracht: Mit $(t, u) \in \mathcal{K}$ ist u eine mögliche nächste Konfiguration von t einer beliebigen Turingmaschine, nicht notwendigerweise eine von AT.
Ich extrahiere aus der Situation (M, s_0) induktiv eine Menge $\tilde{D} \subseteq S$:

a) $s_0 \in \tilde{D}$

b) Sei $t \in \tilde{D}$ und $\alpha(t)$ eine universelle Konfiguration. Da $(M, t) \models g_5$, existiert für jede nächste Konfiguration β von $\alpha(t)$ ein Zustand u_β mit $\alpha(u_\beta) = \beta$ und $(t, u_\beta) \in \mathcal{K}$. Für jede solche Konfiguration β wird u_β zu $\tilde{D}$ hinzugefügt.

c) Sei $t \in \tilde{D}$ und $\alpha(t)$ eine existentielle Konfiguration. Nun gilt $(M, t) \models g_6$, somit gibt es für eine nächste Konfiguration β von $\alpha(t)$ einen Zustand u mit $\alpha(u) = \beta$ und $(t, u) \in \mathcal{K}$. In diesem Fall wird u der Menge $\tilde{D}$ hinzugefügt.

Sei weiter $D \doteq \{\alpha(s) : s \in \tilde{D}\}$. Da M endlich ist und AT keine Schlaufen ausführt, ist auch D endlich. Weiter gilt $q_0x \in D$, weil $s_0 \in D$ und $(M, s_0) \models h$. Damit ist D eine vereinfachte Spur, die q_0x akzeptiert. q.e.d.

Kombiniert man Theorem 3.2.14 mit [CKS81], so ist das Erfüllbarkeitsproblem der Logik K^C offenbar EXPTIME-hart. Die Vollständigkeit ergibt sich mit der Arbeit von Pratt [Pra79]. Jenes Resultat wird in dieser Arbeit aber nicht benötigt, daher soll nicht weiter darauf eingegangen werden. Das Korollar fasst die Ergebnisse zusammen:

Korollar 3.2.15. *Das Erfüllbarkeitsproblem der Logik K_n^C ist EXPTIME-vollständig.*

Kapitel 4

Die Komplexität von $S5_n$

4.1 Der Spezialfall S5

Die Erfüllbarkeitsfrage der Aussagenlogik diente im Kapitel 3 als Beispiel eines NP-vollständigen Problems. Obwohl nun die Ausdrucksstärke von S5 diejenige der Aussagenlogik übertrifft, steigt die Komplexität des Erfüllbarkeitsproblems von S5 überraschenderweise nicht an.

Theorem 4.1.1. *Das Erfüllbarkeitsproblem für S5 ist* NP*-vollständig (und damit ist das Gültigkeitsproblem für S5 co-*NP*-vollständig).*

Beweis. Die Aussagenlogik ist offensichtlich ein Spezialfall von S5. Cook bewies in seiner berühmten Arbeit aus dem Jahre 1971 (vgl. z. B. [GarJo79] oder [Co71]) die NP-Vollständigkeit des Erfüllbarkeitsproblems der Aussagenlogik. Entsprechend ist das Erfüllbarkeitsproblem von S5 (im folgenden SATS5 genannt) sicher NP-hart. Es bleibt die Frage, ob SATS5 in NP liegt, d.h. ob es einen Algorithmus aus NP gibt, der SATS5 entscheidet. Intuitiv kann dieser Algorithmus folgendermassen gewonnen werden: Eine Formel φ sei gegeben. Man wähle eine Struktur $M = (S, \pi, \mathcal{K})$ aus $\mathcal{M}^{rst}$, mit $|S| \leq |\varphi|$ und $\mathcal{K}$ als Allrelation. Nun prüfe man, ob φ in M erfüllt wird. Diese Idee wird nun etwas formaler ausgearbeitet.
Sei φ gegeben mit $|\varphi| = m$. Man rät nun eine Struktur $M = (S, \pi, \mathcal{K})$, wobei S eine Zustandsmenge mit $|S| = k \leq m$, $\mathcal{K} = \{(s,t) : s,t \in \mathcal{K}\}$ und $\pi_s(p) =$ **false** für alle $p \notin Sub(\varphi)$ ist. Das Raten beschränkt sich dabei auf die Wahl von k und der Wahrheitswerte $\pi_s(p)$ der primitiven Propositionen, die in φ vorkommen. Da bei höchstens m Zuständen höchstens m Wahrheitswerte zu bestimmen sind, lässt sich diese Konstruktion in $\mathcal{O}(m^2)$ durchführen. Der Algorithmus aus Lemma 1.7.3 prüft nun φ auf Erfüllbarkeit in M. Dieser Test kann deterministisch in der Zeit $\mathcal{O}(m^3)$ durchgeführt werden. Falls φ erfüllbar ist, dann muss φ nach Theorem 1.7.4 in einer der

erratenen Strukturen erfüllbar sein. Das beschriebene Verfahren entscheidet somit nichtdeterministisch in der Zeit $\mathcal{O}(m^3)$ die Erfüllbarkeit von φ in S5. q.e.d.

4.2 Der allgemeine Fall $S5_n$

Die Frage nach der Erfüllbarkeit war für die Logik S5 ein Problem aus NP. Ein solches Resultat ist im allgemeinen Fall von $S5_n$, $n \geq 2$, nicht zu erwarten, wie das folgende Gegenbeispiel aus [HM92] zeigt.

4.2.1 Gegenbeispiel Nr.1

Lemma 4.2.1. *Sei $\Phi = \{p_1, \ldots, p_m\}$. Es gibt dann eine Formel $\varphi_m^{S5_2}$ mit $|\varphi_m^{S5_2}| \in \mathcal{O}(m)$, die $S5_2$-erfüllbar[1] ist, aber jede Struktur aus $\mathcal{M}_2^{rst}$, welche φ erfüllt, besitzt mindestens 2^m Zustände.*

Beweis. Ich beweise das Lemma erst für die Logik S4. Zum Einen ist die Beweisidee in diesem Fall wesentlich sichtbarer, zum Andern zeige ich so „en passant“ die Gültigkeit des Lemmas für den Fall S4.

Die zu konstruierende Formel φ_m^{S4} soll auf der semantischen Seite die Existenz eines binären Baumes erzwingen, dessen Blätter sämtliche Wahrheitsfunktionen auf den primitiven Propositionen $\{p_1, \ldots, p_n\}$ repräsentieren.

Doch nun an die Arbeit. Zuerst füge ich der Menge $\{p_1, \ldots, p_m\}$ zusätzliche Propositionen $d_0, \ldots, d_{m+1}$ hinzu. In den später folgenden Ausführungen zum Beweis wird gezeigt, dass d_i genau in jenen Zuständen wahr ist, welche im baumartigen Modell eine graphentheoretische Tiefe $\geq i$ besitzen. *depth* wird also wie folgt definiert:

$$depth \equiv \bigwedge_{i=1}^{m+1} (d_i \Rightarrow d_{i-1})$$

Im zu konstruierenden baumartigen Modell soll die graphentheoretische Tiefe i (welche, wie später klar wird, durch die Proposition d_i repräsentiert wird) eines Zustandes den Wahrheitswert der Variablen p_i bestimmen. Anders ausgedrückt, ist p_i wahr (resp. falsch) bei einem Zustand s der Tiefe j, mit $j \geq i$, dann ist p_i bei allen Nachfolgern von s mit Mindesttiefe i wahr (resp. falsch). Das mag dem Leser etwas seltsam erscheinen, nimmt doch in einem Baum die Tiefe beim Übergang von einem Knoten zu dessen Nachfolger stets zu. Nun, im Fall von $S5_2$ ist die Angelegenheit etwas komplizierter, wie

[1] $S5_2$-erfüllbar bedeutet, dass φ in einem Modell von $S5_n$ erfüllbar ist.

später einsichtig wird. Die nachstehende Definition der Formel *determined* fasst die erwähnte Intention zusammen, obwohl dies natürlich erst im Verlauf des Beweises sichtbar wird.

$$\mathit{determined} \equiv \bigwedge_{i=1}^{m}(d_i \Rightarrow ((p_i \Rightarrow K(d_i \Rightarrow p_i)) \wedge (\neg p_i \Rightarrow K(d_i \Rightarrow \neg p_i))))$$

Weiter soll jeder Knoten der Tiefe i zwei Nachfolger der Tiefe $i+1$ so besitzen, dass p_{i+1} bei dem einen wahr und dem andern falsch ist. Die Formel *branching* drückt diese Idee aus:

$$\begin{aligned}\mathit{branching} \equiv \bigwedge_{i=0}^{m-1}((d_i \wedge \neg d_{i+1}) \Rightarrow \\ (\neg K\neg(d_{i+1} \wedge \neg d_{i+2} \wedge p_{i+1}) \wedge \neg K\neg(d_{i+1} \wedge \neg d_{i+2} \wedge \neg p_{i+1})))\end{aligned}$$

Schliesslich sei φ_m^{S4} folgende Formel:

$$\varphi_m^{S4} \equiv d_0 \wedge \neg d_1 \wedge K(\mathit{depth} \wedge \mathit{determined} \wedge \mathit{branching})$$

Da die Teilformeln *depth, determined* und *branching* alle nur linear in m anwachsen, wird $|\varphi_m^{S4}| \in O(m)$ durch eine kleine Buchhaltungsrechnung bestätigt. Weiter ist auch die Erfüllbarkeit von φ_m^{S4} leicht einzusehen, z. B. in einer einelementigen Struktur. Etwas genauer: $N = (S, \pi, \mathcal{K})$ wobei

$$\begin{aligned} S &\doteq \{s\} \\ \pi_s(d_0) &\doteq \textbf{true} \\ \pi_s(d_i) &\doteq \textbf{false} \text{ für alle } i \geq 1 \\ \pi_s(p_i) &\doteq \text{beliebig für alle } i \\ \mathcal{K} &\doteq \emptyset \end{aligned}$$

Ich gebe zu, das Beispiel wirft keine langen Schatten, aber immerhin, φ_m^{S4} ist erfüllbar. Natürlich widerspricht die Struktur N der Behauptung 4.2.1 nicht, denn $\mathcal{K}$ ist nicht reflexiv und damit gilt $N \notin \mathcal{M}^{rt}$.[2]

Sei nun $M = (S, \pi, \mathcal{K}) \in \mathcal{M}_1^{rt}$ und $(M, s) \models \varphi_m^{S4}$. Eine Induktion über j zeigt: Falls $j \leq m$, dann gibt es zu jeder beliebigen Wahrheitsfunktion v über den Variablen $p_1, \ldots, p_j$ einen von s erreichbaren Zustand t derart, dass $(M, t) \models d_j \wedge \neg d_{j+1}$ und $(M, t) \models p_i$ gdw $v(p_i) = \textbf{true}$, $i \in \{1, \ldots, j\}$.
Zur Induktionsverankerung: Da $(M, s) \models d_0 \wedge \neg d_1 \wedge \mathit{branching}$, gibt es zwei

[2]Man beachte: Wird Lemma 4.2.1 sinngemäss an die Logik S4 angepasst, so sind Strukturen aus $\mathcal{M}^{rt}$ zu betrachten.

Nachfolger t_0 und t_1 von s mit $(M, t_0) \models d_1 \wedge \neg d_2 \wedge p_1$ und $(M, t_1) \models d_1 \wedge \neg d_2 \wedge \neg p_1$.
Induktionsschritt: Sei $j \geq 1$ und v sei eine Wahrheitsfunktion über den Variablen $p_1, \ldots, p_{j+1}$. Nach der Induktionsvoraussetzung gibt es einen von s erreichbaren Zustand t mit $(M, t) \models d_j \wedge \neg d_{j+1}$ und $(M, t) \models p_i$ gdw $v(p_i) = \mathbf{true}$ für $i \in \{1, \ldots, j\}$. Nun ist t von s erreichbar und $\mathcal{K}$ ist transitiv, folglich $(M, t) \models \mathit{depth} \wedge \mathit{determined} \wedge \mathit{branching}$. Dank *branching* existiert ein Nachfolger t' von t mit $(M, t') \models d_{j+1} \wedge \neg d_{j+2}$ und $(M, t') \models p_{j+1}$ gdw $v(p_{j+1}) = \mathbf{true}$. Mit *depth* folgt $(M, t) \models d_1 \wedge \cdots \wedge d_j$. Diese Konjunktion zusammen mit *determined* stellt sicher, dass sich die Wahrheitswerte der p_i für $i \in \{1, \ldots, j\}$ beim Übergang von t zu t' nicht verändert haben, oder anders ausgedrückt: $(M, t') \models p_i$ gdw $v(p_i) = \mathbf{true}$ für $i \in \{1, \ldots, j\}$. Der Zustand t' erfüllt somit die an ihn gestellten Anforderungen. Mit $j = m$ wird nun jede Wahrheitsfunktion v über den Variablen $p_1, \ldots, p_m$ durch ein Blatt des Baumes repräsentiert. Bekanntlich gibt es 2^m Wahrheitsfunktionen über m Variablen und somit besitzt M mindestens 2^m Zustände.

Die Formel φ_m^{S4} muss nun zu einer in $S5_2$ erfüllbaren Formel umgestaltet werden. Zuerst wird jedes Auftreten des Operators K in φ_m^{S4} durch die Kombination von K_1K_2 ersetzt:

$$\mathit{depth} \equiv \bigwedge_{i=1}^{m+1} (d_i \Rightarrow d_{i-1})$$

$$\mathit{determined}' \equiv \bigwedge_{i=1}^{m} (d_i \Rightarrow ((p_i \Rightarrow K_1K_2(d_i \Rightarrow p_i)) \wedge (\neg p_i \Rightarrow K_1K_2(d_i \Rightarrow \neg p_i))))$$

$$\mathit{branching}' \equiv \bigwedge_{i=0}^{m-1} ((d_i \wedge \neg d_{i+1}) \Rightarrow (\neg K_1K_2\neg(d_{i+1} \wedge \neg d_{i+2} \wedge p_{i+1}) \wedge \neg K_1K_2\neg(d_{i+1} \wedge \neg d_{i+2} \wedge \neg p_{i+1})$$

Entscheidend am Gelingen der obigen Konstruktion war die Tatsache, dass falls $(M, s) \models \varphi_m^{S4}$ und $M \in \mathcal{M}_1^{rt}$, dann $(M, t) \models \mathit{depth} \wedge \mathit{determined} \wedge \mathit{branching}$ für alle von s in höchstens m Schritten erreichbaren Zustände t. Damit diese Eigenschaft auch bei der Kombination von K_1K_2 zu Verfügung steht, reicht ein blosses Ersetzen des Operators K durch K_1K_2 nicht aus. Gilt $(M, s) \models K_1K_2\mathit{depth} \wedge \mathit{determined} \wedge \mathit{branching}$, dann soll $(M, t) \models \mathit{depth} \wedge \mathit{determined} \wedge \mathit{branching}$ bei jenen Zuständen t gelten, die über

maximal m Alternationen von $\mathcal{K}_1$- und $\mathcal{K}_2$-Kanten erreichbar sind. Mit Axiom **4** folgt zwar leicht $(M,s) \models K_1K_2\varphi \Rightarrow K_1K_1K_2K_2\varphi$, doch ist $(M,s) \models K_1K_2\varphi \Rightarrow K_1K_2K_1K_2\varphi$ keine gültige Implikation. Es bedarf einer zusätzlichen Änderung, die neue Formel $\varphi_m^{S5_2}$ nimmt daher folgende Gestalt an:

$$\varphi_m^{S5_2} \equiv d_0 \wedge \neg d_1 \wedge (K_1K_2)^m(\mathit{depth} \wedge \mathit{determined}' \wedge \mathit{branching}')$$

Zweifelsohne ist $\varphi_m^{S5_2}$ erfüllbar, ich bediene mich wieder der einelementigen Struktur aus dem vorhergehenden Fall der Logik S4. Sei $N' = (S', \pi', \mathcal{K}_1, \mathcal{K}_2)$ mit

$$\begin{aligned} S' &\doteq \{s\} \\ \pi'_s(d_0) &\doteq \textbf{true} \\ \pi'_s(d_i) &\doteq \textbf{false} \text{ für alle } i \geq 1 \\ \pi'_s(p_i) &\doteq \text{beliebig für alle } i \\ \mathcal{K}_1 &\doteq \emptyset \\ \mathcal{K}_2 &\doteq \emptyset \end{aligned}$$

Erneut gilt $N' \notin \mathcal{M}_2^{rst}$, ein Widerspruch zu Lemma 4.2.1 entsteht nicht. Weiter gilt offenbar $|\varphi_m^{S5_2}| \in O(m)$. Es bleibt zu zeigen, dass für jede Struktur $M' = (S', \pi', \mathcal{K}_1, \mathcal{K}_2)$ aus $\mathcal{M}_2^{rst}$, die $\varphi_m^{S5_2}$ erfüllt, $|S'| \geq 2^m$ gilt. Es sei also $(M', s) \models \varphi_m^{S5_2}$. Wieder lässt sich, analog zur Formel φ_m^{S4}, induktiv über j zeigen: Falls $j \leq m$, dann gibt es zu jeder beliebigen Wahrheitsfunktion v über den Variablen $p_1, \ldots, p_j$ einen von s via $\mathcal{K}_1 \circ \mathcal{K}_2$-Kanten erreichbaren Zustand t derart, dass $(M,t) \models d_j \wedge \neg d_{j+1}$ und $(M,t) \models p_i$ gdw $v(p_i) =$ **true**,$i \in \{1, \ldots, j\}$. Dabei bezeichnet $\mathcal{K}_1 \circ \mathcal{K}_2$ das Relationenprodukt von $\mathcal{K}_1$ und $\mathcal{K}_2$, d.h.

$$\mathcal{K}_1 \circ \mathcal{K}_2 = \{(s,u) \mid \exists t \in S : s\mathcal{K}_1 t \mathcal{K}_2 u\}.$$

Zur Veranschaulichung wird die Semantik einer Formel der Form $K_1K_2\psi$ kurz erläutert:

$$\begin{aligned} (M,s) \models K_1K_2\psi \quad \text{gdw} \quad & (M,t) \models K_2\psi \text{ für alle } t \text{ mit } (s,t) \in \mathcal{K}_1 \\ \text{gdw} \quad & (M,u) \models \psi \text{ für alle } t \text{ mit } (s,t) \in \mathcal{K}_1 \\ & \text{und für alle } u \text{ mit } (t,u) \in \mathcal{K}_2 \\ \text{gdw} \quad & (M,u) \models \psi \text{ für alle } u \text{ zu denen ein } t \text{ existiert, so} \\ & \text{dass } (s,t) \in \mathcal{K}_1 \text{ und } (t,u) \in \mathcal{K}_2 \\ \text{gdw} \quad & (M,u) \models \psi \text{ für alle } u \text{ mit} (s,u) \in \mathcal{K}_1 \circ \mathcal{K}_2 \end{aligned}$$

Weitere Unterschiede zum Beweis im Falle der S4-Formel treten aber nicht auf, eine Auflistung der Beweisschritte käme bloss einer langweiligen Wiederholung gleich.

Allerdings wird nun klar, dass die Propositionen $d_0, \ldots, d_{m+1}$ nicht weiter die graphentheoretische Tiefe repräsentieren; beim Übergang von s zu seinen $\mathcal{K}_1 \circ \mathcal{K}_2$-Nachfolger steigt diese zwar um zwei Einheiten, die durch die d_i's angezeigte Tiefe nimmt aber nur um einen Schritt zu. Die anfänglich etwas seltsam anmutende Definition der Formel *determined* erweist sich nun als passend. q.e.d.

Bemerkung 4.2.2. Ein direktes Übertragen der Konstruktion des Lemmas 4.2.1 auf die Sprache S5 ist nicht möglich. Wäre φ_m^{S4} in einem Modell von S5 erfüllbar, so nach Lemma 1.7.7 in einer Struktur $M = (S, \pi, \mathcal{K})$ mit $\mathcal{K}$ als Allrelation. Aus $(M, s) \models \varphi_m^{S4}$ folgte mit *branching* die Existenz zweier Zustände t und u mit $(M, t) \models d_1 \wedge p_1$ und $(M, u) \models d_1 \wedge \neg p_1$. Zudem gälte $(M, t) \models$ *determined*, damit auch $(M, t) \models (d_1 \wedge p_1) \Rightarrow Kp_1$. Da aber $(t, u) \in \mathcal{K}$, würde der Widerspruch $(M, u) \models p_1$ folgen.

Das Lemma 4.2.1 zerschlägt die Hoffnung, das Erfüllbarkeitsproblem von $S5_2$ mit einem Verfahren aus NP entscheiden zu können. Die Arbeit von Ladner ([Lad77]) bestätigt diese Einsicht. Er zeigt dort zwar die PSPACE-Vollständigkeit des Erfüllbarkeitsproblems von S4, doch lässt sich seine Methode leicht auf den Fall $S5_2$ übertragen. So komme ich zu einem ersten Höhepunkt dieser Arbeit: Das Erfüllbarkeitsproblem der Logik $S5_2$ ist PSPACE-hart.

4.2.2 $SATS5_n$ ist PSPACE-hart

Eine Verallgemeinerung des Beweises aus [Lad77] liefert die untere PSPACE-Schranke für das Erfüllbarkeitsproblem der Logik $S5_2$.

Theorem 4.2.3. *Das Erfüllbarkeitsproblem der Logik $S5_2$ ist PSPACE-hart.*

Beweis. Das Theorem wird durch eine Reduktion des Problems QBF (siehe Beispiel 3.2.3) auf $SATS5_2$ bewiesen. Wie bereits im Beweis zum Lemma 4.2.1 betrachte ich auch hier erst den Fall der Logik S4, d.h ich reduziere QBF auf SATS4.

Sei $A = Q_1 p_1 \ldots Q_m p_m B$ eine beliebige Formel aus QBF, wobei B nur Variablen aus $\{p_1, \ldots, p_m\}$ enthält. Zu konstruieren bleibt eine Formel ψ_A^{S4}, die genau dann erfüllbar ist, wenn $A \equiv \mathbf{true}$. Als roter Faden dient das Vorgehen im Beweis zum Lemma 4.2.1, einzig mit dem Unterschied, nicht alle Wahrheitsfunktionen über den Propositionen $p_1, \ldots, p_m$ zu betrachten, sondern nur die für $A \equiv \mathbf{true}$ relevanten.

Wieder füge ich den primitiven Propositionen $p_1, \ldots, p_m$ die Variablen $d_0, \ldots, d_{m+1}$ hinzuzu, wobei die intentierte Bedeutung der Variablen d_i erneut als Mindesttiefe im „Baum der Wahrheitsfunktionen“ zu verstehen ist. Die

Formel *depth* und *determined* werden unverändert aus dem Beweis zu Lemma 4.2.1 übernommen:

$$depth \equiv \bigwedge_{i=1}^{m+1} (d_i \Rightarrow d_{i-1})$$

$$determined \equiv \bigwedge_{i=1}^{m} (d_i \Rightarrow ((p_i \Rightarrow K(d_i \Rightarrow p_i)) \wedge (\neg p_i \Rightarrow K(d_i \Rightarrow \neg p_i))))$$

Die Formel $branching_A$ resultiert aus einer kleinen Modifikation der Formel *branching*. Erzwingt letztere als Modell einen binären Baum, der vollständig alle Wahrheitsfunktionen über den Variablen $\{p_1, \ldots, p_m\}$ repräsentiert, so sind nun gewisse Wahrheitsfunktionen irrelevant. Für $Q_i = \exists$ muss lediglich *eine* Belegung existieren, die der Variablen p_i den für $A \equiv \mathbf{true}$ passenden Wert zuschreibt. Somit gilt $branching_A \equiv$

$$\bigwedge_{\{i:Q_{i+1}=\forall\}} ((d_i \wedge \neg d_{i+1}) \Rightarrow (\neg K \neg (d_{i+1} \wedge \neg d_{i+2} \wedge p_{i+1}) \wedge \neg K \neg (d_{i+1} \wedge \neg d_{i+2} \wedge \neg p_{i+1}))) \wedge$$
$$\bigwedge_{\{i:Q_{i+1}=\exists\}} ((d_i \wedge \neg d_{i+1}) \Rightarrow (\neg K \neg (d_{i+1} \wedge \neg d_{i+2} \wedge p_{i+1}) \vee \neg K \neg (d_{i+1} \wedge \neg d_{i+2} \wedge \neg p_{i+1})))$$

Mit diesen Bauteilen lässt sich die gesuchte Formel ψ_A^{S4}zusammenstellen:

$$\psi_A^{S4} \equiv d_0 \wedge \neg d_1 \wedge K(depth \wedge determined \wedge branching_A \wedge (d_m \Rightarrow B)).$$

Sei nun $A \equiv \mathbf{true}$. Man betrachte nun den Ableitungsbaum der quantifizierten Boolschen Formel $A = Q_1 p_1 \ldots Q_m p_m B$, der alle möglichen Belegungen der Variablen beschreibt:

Dieser Baum hat Tiefe m und besitzt $2^{m+1} - 1$ Knoten, welche zusammen die Menge J bilden. Jeder Endpunkt stellt eine Wahrheitsfunktion auf den Variablen $p_1, \ldots, p_m$ dar. Aus dem Ableitungsgraph lässt sich leicht eine Struktur M aus $\mathcal{M}^{rt}$ gewinnen, die ψ_A^{S4} erfüllt. Ich definiere induktiv eine Funktion $l : J \to \{0,1\}^*$, die den Knoten des Graphen Wörter über dem Alphabet $\{0,1\}$ zuordnet.

$$l(s) \doteq \epsilon \text{ (das leere Wort) falls } s \text{ die Wurzel des Baumes ist}$$
$$l(w) \doteq \begin{cases} l(s);1 & \text{falls } w \text{ linker Nachfolger von } s \text{ ist} \\ l(s);0 & \text{falls } w \text{ rechter Nachfolger von } s \text{ ist} \end{cases}$$

Dabei bezeichnet $l(s);1$ die Konkanation der Wörter $l(s)$ und 1. Weiter bezeichen $|l(s)|$ die Länge des Wortes $l(s)$ über dem Alphabet $\{0,1\}$ und

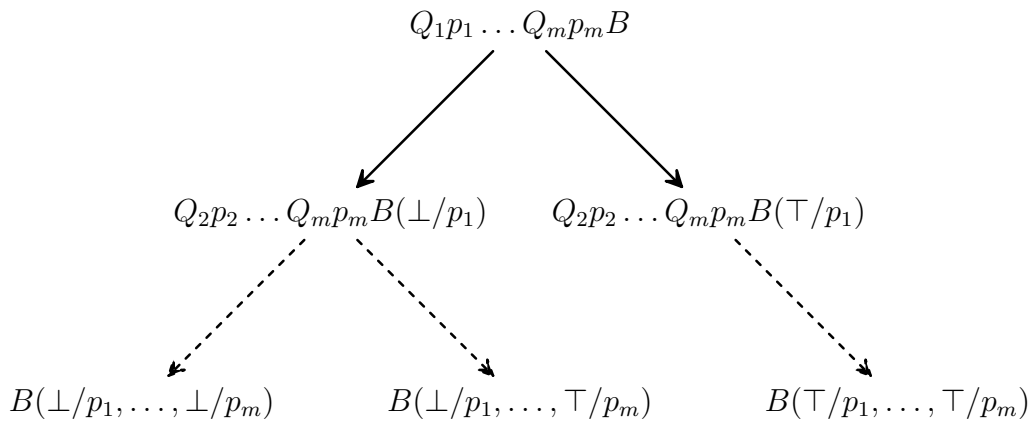


Bild 3: Ableitungsgraph für $Q_1p_1 \ldots Q_mp_mB$;
$\bot$ steht für **false**, $\top$ für **true**.

$(\alpha)_i$ die i-te Komponente des Wortes α. Die Menge der Wörter der Länge $\leq m$ heisse Λ. Jedes Wort $\alpha \in \Lambda$ entspricht somit eindeutig einem Knoten im Ableitungsgraphen. Ausgestattet mit diesen Hilfsmitteln, lässt sich die Struktur $M = (S, \pi, \mathcal{K})$ definieren:

$$\begin{aligned} S &\doteq \Lambda \\ \pi_\alpha(d_i) &\doteq \textbf{true} \text{ falls } |\alpha| \geq i \\ \pi_\alpha(p_i) &\doteq \begin{cases} \textbf{true} & \text{falls } |\alpha| \geq i \text{ und } (\alpha)_i = 1 \\ \textbf{false} & \text{sonst} \end{cases} \\ \mathcal{K} &\doteq \{(\alpha, \beta) : l^{-1}(\beta) \text{ ist Nachfolger von } l^{-1}(\alpha)\}^{rt} \end{aligned}$$

Offensichtlich ist $M \in \mathcal{M}^{rt}$. Da $A \equiv \textbf{true}$, wird B bei mindestens einem Endpunkt erfüllt. Damit und mit den Überlegungen zum Beweis von 4.2.1 ist ohne Schwierigkeit $(M, \epsilon) \models \psi_A^{S4}$ einzusehen.

Sei andererseits $M = (S, \pi, \mathcal{K}) \in \mathcal{M}^{rt}$ und $(M, s) \models \psi_A^{S4}$. Zu jedem Zustand $t \in S$ bezeichne A_j^t die quantifizierte Boolsche Formel der Form $Q_{j+1}p_{j+1} \ldots Q_mp_mB$, wobei jedes Auftreten von $p_i, i \leq j$, durch **true** ersetzt wurde, falls $\pi_t(p_i) = \textbf{true}$, ansonsten durch **false**. Damit ist $A_0^t = A$ und A_m^t entsteht aus B durch entsprechendes Ersetzen aller p_i durch **true** oder **false**. Nun impliziert $(M, s) \models K(d_m \Rightarrow B)$, falls $(s, t) \in \mathcal{K}$ und $(M, t) \models d_m$, dass $A_m^t \equiv \textbf{true}$; $(M, t) \models B$ bedeutet ja gerade $A_m^t \equiv \textbf{true}$, nach Definition der Gültigkeit. Mit einer Induktion über j zeige ich, falls $(s, t) \in \mathcal{K}$ und $(M, t) \models d_{m-j} \wedge \neg d_{m-j+1}$, dann $A_{m-j}^t \equiv \textbf{true}$.
Sei $j = 0$. Dann $(s, t) \in \mathcal{K}$ und $(M, t) \models d_m \wedge \neg d_{m+1}$. Somit also $(M, t) \models B$, ergo $A_m^t \equiv \textbf{true}$.
Sei $j > 0$ und $(s, t) \in \mathcal{K}$, $(M, t) \models d_{m-j} \wedge \neg d_{m-j+1}$. Man beachte: $A_{m-j}^t =$

$Q_{m-j+1}p_{m-j+1}A^t_{m-j+1}$. Da $(s,t) \in \mathcal{K}$ gilt auch $(M,t) \models \mathit{branching}_A$. Ist nun $Q_{m-j+1} = \forall$, dann besitzt t zwei Nachfolger u und v mit Tiefe d_{m-j+1} und $\pi_u(p_{m-j+1}) = \mathbf{true}$, $\pi_v(p_{m-j+1}) = \mathbf{false}$. Nach der Induktionsvoraussetzung gilt $A^u_{m-j+1} \equiv \mathbf{true}$, resp. $A^v_{m-j+1} \equiv \mathbf{true}$ gilt. Damit folgt aber auch $A^t_{m-j} \equiv \mathbf{true}$; wenn A^u_{m-j+1} und A^v_{m-j+1} zu einer wahren Formel aufgedröselt werden können, dann sicher auch $\forall p_{m-j+1}A^t_{m-j+1}$. Der Fall $Q_{m-j+1} = \exists$ verläuft analog.
So folgt, mit $j = m$ und $(M,s) \models d_0$, $A^t_0 \equiv \mathbf{true}$.

Somit gilt $A \equiv \mathbf{true}$ genau dann wenn ψ^{S4}_A in einer Struktur aus $\mathcal{M}^{rt}$ erfüllbar ist. Nun gilt mit Lemma 1.6.5: Eine Formel φ ist genau dann S4-erfüllbar, wenn sie in einer Struktur M aus $\mathcal{M}^{rt}$ erfüllt werden kann. Somit ergibt sich:

$$\psi^{\mathrm{S4}}_A \text{ ist S4-erfüllbar gdw } A \equiv \mathbf{true}.$$

Weiter ist $|\psi^{S4}_A|$ linear in der Länge von A und die Reduktion von QBF auf SATS4 daher gelungen.

Die Umwandlung von ψ^{S4}_A in ψ^{S5}_A geschieht nach dem in 4.2.1 gezeigten Muster. Demnach hat ψ^{S5}_A die Gestalt

$$d_0 \wedge \neg d_1 \wedge (K_1K_2)^m(\mathit{depth} \wedge \mathit{determined} \wedge \mathit{branching}_A \wedge (d_m \Rightarrow B))$$

Sei nun $A \equiv \mathbf{true}$. Die Formel ψ^{S4}_A ist dann in der auf Seite 56 definierten Struktur $M = (S, \pi, \mathcal{K})$ erfüllbar. Die Struktur $M' \in \mathcal{M}^{rst}_2$ entsteht aus M durch Ersetzen der $\mathcal{K}$-Kanten mit $\mathcal{K}_1 \circ \mathcal{K}_2$-Kanten. Die formale Definition von $M' = (S', \pi', \mathcal{K}_1, \mathcal{K}_2)$ lautet folgendermassen:

$$\begin{aligned}
S' &\doteq S \cup \{s_{(\alpha,\beta)} : (\alpha,\beta) \in \mathcal{K}\} \\
\pi'_\alpha(p) &\doteq \pi_\alpha(p) \text{ falls } \alpha \in S \\
\pi'_{s_{(\alpha,\beta)}}(p) &\doteq \pi_\alpha(p) \text{ falls } s_{(\alpha,\beta)} \in S' \setminus S \\
\mathcal{K}_1 &\doteq \{(\alpha, s_{(\alpha,\beta)}) : (\alpha,\beta) \in \mathcal{K}\}^{rst} \\
\mathcal{K}_2 &\doteq \{(s_{(\alpha,\beta)}, \beta) : (\alpha,\beta) \in \mathcal{K}\}^{rst}
\end{aligned}$$

Damit keine Unklarheiten entstehen: $p \in \{p_1, \ldots, p_m\} \cup \{d_0, \ldots, d_{m+1}\}$.
Mit dieser Definition sind $\mathcal{K}_1$ und $\mathcal{K}_2$ sicher Äquivalenzrelationen. Problemlos einzusehen ist auch $(M', \epsilon) \models \psi^{S5}_A$.
Gilt andererseits $M = (S, \pi, \mathcal{K}_1, \mathcal{K}_2)$ und $(M,s) \models \psi^{S5}_A$. Die weitere Argumentation deckt sich mit jener der S4-Formel, mit dem Unterschied, dass nun $(s,t) \in \mathcal{K}_1 \circ \mathcal{K}_2$ gilt. q.e.d.

Bemerkung 4.2.4. Mit Theorem 4.2.3 sind natürlich auch die Logiken $\mathrm{S4}_n, n \geq 1$ und $\mathrm{S5}_n, n \geq 2$, PSPACE-hart.

4.2.3 PSPACE-Entscheidungsverfahren für $S5_n$

Mit dem Resultat des vorhergehenden Abschnittes besitzt ein Verfahren, welches das Erfüllbarkeitsproblem der Logik $S5_2$ entscheidet, mindestens die Komplexität PSPACE. In diesem Abschnitt zeige ich, dass ein solcher Algorithmus auch wirklich in der Klasse PSPACE liegt. $SATS5_2$ ist demzufolge PSPACE-vollständig.

Das präsentierte Entscheidungsverfahren orientiert sich an der Tableau-Methode, die z. B. in [Smu68] vorgestellt wird. Es ist eine Übertragung der Arbeit von [HM92] auf die Logik $S5_n$.

Definition 4.2.5 (Aussagenlogisches Tableau). Ein *aussagenlogisches Tableau* bezeichent eine Formelmenge T mit den Eigenschaften:

a) Falls $\neg\neg\psi \in T$, dann $\psi \in T$.

b) Falls $\psi \wedge \psi' \in T$, dann $\psi, \psi' \in T$.

c) Falls $\neg(\psi \wedge \psi') \in T$, dann $\neg\psi \in T$ oder $\neg\psi' \in T$.

d) Für keine Formel $\psi \in \mathcal{L}_n(\Phi)$ gilt $\psi \in T$ und $\neg\psi \in T$.

Bemerkung 4.2.6. Ist T ein aussagenlogisches Tableau mit $\varphi \in T$, so nennt man T ein „aussagenlogisches Tableau für φ“ oder spricht „φ besitzt das Tableau T“.

Lemma 4.2.7. *Eine aussagenlogische Formel φ ist genau dann erfüllbar, wenn sie ein Tableau besitzt.*

Beweis. Sei φ erfüllbar, d. h. es gibt eine Belegung v so, dass $v(\varphi) = \mathbf{true}$. Die Menge $T \doteq \{\psi \in \mathit{Sub}^+(\varphi) : v(\psi) = \mathbf{true}\}$ ist dann ein Tableau für φ, wie die nachstehenden Überlegungen zeigen:

- Da $v(\varphi) = \mathbf{true}$, gilt offenbar $\varphi \in T$.
- Ist $v(\neg\neg\psi) = \mathbf{true}$, dann auch $v(\psi) = \mathbf{true}$.
- Mit $v(\psi \wedge \psi') = \mathbf{true}$ gilt auch $v(\psi) = \mathbf{true}$ und $v(\psi') = \mathbf{true}$.
- Falls $v(\neg(\psi \wedge \psi')) = \mathbf{true}$, dann folgt $v(\neg\psi) = \mathbf{true}$ oder $v(\neg\psi') = \mathbf{true}$.
- Ist $v(\psi) = \mathbf{true}$, dann gilt $v(\neg\psi) = \mathbf{false}$. Somit gilt entweder $\psi \in T$ oder $\neg\psi \in T$.

Sei andererseits T ein Tableau für φ. Die in φ auftretenden Variablen seien in $\{x_1, \ldots, x_n\}$ enthalten. Definiere die Belegung v durch

$$v(x_i) = \begin{cases} \textbf{true} & \text{falls } x_i \in T \\ \textbf{false} & \text{falls } x_i \notin T \end{cases}$$

Für alle $\psi \in Sub^+(\varphi)$ zeige ich durch simultane Induktion über den Aufbau von ψ:

a) $\psi \in T$ impliziert $v(\psi) = \textbf{true}$

b) $\neg\psi \in T$ impliziert $v(\neg\psi) = \textbf{true}$

Die Behauptung folgt aufgrund folgender Argumente:

- Ist ψ von der Form x_i oder $\neg x_i$, so liefert die Definition von v das gewünschte Resultat.
- Sei $\psi \equiv \xi \wedge \chi$. Aus $\xi \wedge \chi \in T$ folgt $\xi \in T$ und $\chi \in T$. Mit der Induktionsvoraussetzung folgt $v(\xi) = \textbf{true}$ und $v(\chi) = \textbf{true}$, also $v(\xi \wedge \chi) = \textbf{true}$.
- Ist $\psi \equiv \neg(\xi \wedge \chi)$, dann $\neg\xi \in T$ oder $\neg\chi \in T$. Die Induktionsvoraussetzung liefert $v(\xi) = \textbf{true}$ oder $v(\chi) = \textbf{true}$, womit $v(\xi \vee \chi) = \textbf{true}$, resp. $v(\neg(\xi \wedge \chi)) = \textbf{true}$ folgt.
- Es ist $\psi \equiv \neg\xi$ und $\neg\xi \in T$. Die Induktionsvoraussetzung von (b) liefert unmittelbar $v(\neg\xi) = \textbf{true}$.
- Schliesslich bleibt noch $\psi \equiv \neg\xi$ und $\neg\neg\xi \in T$. Damit ist aber auch $\xi \in T$, mit der Induktionsvoraussetzung folgt daher $v(\xi) = \textbf{true}$ und folglich gilt $v(\neg\neg\xi) = \textbf{true}$.

q.e.d.

Das aussagenlogische Tableau soll nun auf die Modallogik $S5_2$ übertragen werden. Dazu ist der Begriff der Labelfunktion notwendig.

Definition 4.2.8. Sei S eine Menge von Zuständen. Eine Funktion $L : S \to \mathcal{P}(\mathcal{L}_n(\Phi))$ heisst *$S5_n$-Labelfunktion*, falls sie folgende Bedingungen verfüllt:

a) Für alle $s \in S$ ist $L(s)$ ein aussagenlogisches Tableau.

b) Falls $K_i\psi \in L(s)$ und $(s,t) \in \mathcal{K}_i$, dann $\psi \in L(t)$.

c) Falls $\neg K_i\psi \in L(s)$, dann existiert ein t mit $(s,t) \in \mathcal{K}_i$ und $\neg\psi \in L(t)$.

d) Falls $K_i\psi \in L(s)$, dann $\psi \in L(s)$.

e) Falls $(s,t) \in \mathcal{K}_i$, dann $K_i\psi \in L(s)$ gdw $K_i\psi \in L(t)$.

Definition 4.2.9. Ein $S5_n$-Tableau ist ein Tupel $T = (S, L, \mathcal{K}_1, \ldots, \mathcal{K}_n)$, mit

a) einer Zustandsmenge S,

b) einer $S5_n$-Labelfunktion L

c) und binären Relationen $\mathcal{K}_1, \ldots, \mathcal{K}_n$.

Bemerkung 4.2.10. Ist $T = (S, L, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ ein $S5_n$-Tableau mit $\varphi \in L(s)$ für ein beliebiges $s \in S$, so nennt man T ein „$S5_n$-Tableau für φ“, oder spricht „φ besitzt ein $S5_n$-Tableau “.

Lemma 4.2.11. *Ein Formel φ ist genau dann $S5_n$-erfüllbar, wenn sie ein $S5_n$-Tableau besitzt.*

Beweis. Sei φ $S5_n$-erfüllbar. Dann gibt es eine Struktur $M = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n) \in \mathcal{M}_n^{rst}$, die φ erfüllt. Man setze $T = (S, L, \mathcal{K}_1, \ldots, \mathcal{K}_n)$, wobei $L(s) = \{\psi : (M,s) \models \psi\}$. Dann gilt:

a) $L(s) = \{\psi : (M,s) \models \psi\}$ ist aufgrund der Definition der Gültigkeit ein aussagenlogisches Tableau.

b) Gilt $K_i\xi \in L(s)$, dann $(M,s) \models K_i\xi$. Somit für alle t mit $(s,t) \in \mathcal{K}_i$: $(M,t) \models \xi$. Also $\xi \in L(t)$.

c) Ist $\neg K_i\xi \in L(s)$, so $(M,s) \models \neg K_i\xi$. Damit existiert ein t mit $(s,t) \in \mathcal{K}_i$ und $(M,t) \models \neg\xi$ und folglich $\neg\xi \in L(t)$.

d) Gilt $K_i\xi \in L(s)$, dann wiederum $(M,s) \models K_i\xi$. Mit der Reflexivität der $\mathcal{K}_i$ folgt mühelos $\xi \in L(s)$.

e) Ist $(s,t) \in \mathcal{K}_i$ und gilt $K_i\xi \in L(s)$, dann $(M,s) \models K_i\xi$. Mit der Transitivität von $\mathcal{K}_i$ folgt $(M,t) \models K_i\xi$ für alle t mit $(s,t) \in \mathcal{K}_i$. Damit auch $K_i\xi \in L(t)$. Sei andererseits $K_i\xi \notin L(s)$. Somit $(M,s) \models \neg K_i\xi$. Es gibt also einen Zustand u mit $(s,u) \in \mathcal{K}_i$ und $(M,u) \models \neg\xi$. Sei t ein beliebiger Zustand mit $(s,t) \in \mathcal{K}_i$. Aus Symmetrie und Transitivität von $\mathcal{K}_i$ folgen $(t,s) \in \mathcal{K}_i$ und $(t,u) \in \mathcal{K}_i$. Damit ergibt sich $(M,t) \models \neg K_i\xi$ und $\neg K_i\xi \notin L(t)$ für alle t mit $(s,t) \in \mathcal{K}_i$.

Damit ist $T = (S, L, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ ein S5$_n$-Tableau für φ.

Sei $T = (S, L, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ ein S5$_n$-Tableau für φ. Sei $M = (S, \pi, \mathcal{K}'_1, \ldots, \mathcal{K}'_n)$, wobei $\mathcal{K}'_i \doteq \mathcal{K}_i^{rst}$ und

$$\pi_s(p) = \begin{cases} \textbf{true} & \text{falls } p \in L(s) \\ \textbf{true} & \text{falls } p \notin L(s) \end{cases}.$$

Sei $\psi \in \mathit{Sub}^+(\varphi)$. Ich zeige induktiv über den Aufbau von ψ: $\psi \in L(s)$ impliziert $(M, s) \models \psi$ und $\neg\psi \in L(s)$ impliziert $(M, s) \models \neg\psi$.

- Die Fälle $\psi \equiv p, \psi \equiv (\xi \wedge \chi)$ und $\psi \equiv \neg\xi$ wurden bereits im Beweis zum Lemma 4.2.7 gezeigt.

- $\psi \equiv K_i\psi'$. Gilt $\neg K_i\psi' \in L(s)$, dann garantiert die Definition des S5$_n$ Tableaus die Existenz eines Zustandes t mit $(s, t) \in \mathcal{K}_i$ und $\neg\psi' \in L(t)$. Mit $\mathcal{K}_i \subseteq \mathcal{K}'_i$ und der Induktionsvoraussetzung $(M, t) \models \neg\psi'$ folgt $(M, s) \models \neg K_i\psi'$.
  Andererseits sei $K_i\psi' \in L(s)$. Für beliebige $(s, t) \in \mathcal{K}'_i$ existieren ein $k > 0$ und Zustände $s_0, \ldots, s_k$ mit $s = s_0, t = s_k$ so, dass für alle $j < k$ entweder $(s_j, s_{j+1}) \in \mathcal{K}_i$ oder $(s_{j+1}, s_j) \in \mathcal{K}_i$ gilt. Mit der Eigenschaft (e) der Definition 4.2.8 folgt induktiv über j leicht $K_i\psi' \in L(s_j)$ für alle $j \leq k$. Damit auch $K_i\psi' \in L(t)$. Mit dem Punkt (d) der erwähnten Definition ergibt sich $\psi' \in L(t)$ und die Induktionsvoraussetzung liefert $(M, t) \models \psi'$. Da t als beliebiger Zustand mit $(s, t) \in \mathcal{K}'_i$ gewählt wurde, folgt $(M, s) \models K_i\psi'$.

Weiter ist T ein S5$_n$-Tableau für φ, d. h. es gibt ein $s \in S$, so dass $\varphi \in L(s)$. Offensichtlich gilt $\varphi \in \mathit{Sub}^+(\varphi)$, mit der eben bewiesenen Behauptung folgt demnach $(M, s) \models \varphi$. Damit ist φ S5$_n$-erfüllbar. q.e.d.

Der im weiteren präsentierte Algorithmus versucht für jede Formel φ die Konstruktion eines Tableaus. Er wird genau dann erfolgreich sein, wenn φ S5$_n$-erfüllbar ist. Ein zweiter Algorithmus aus PSPACE wird zudem in der Lage sein, die Konstruktion auf ihren Erfolg hin zu testen.
Doch vorerst zwei neue Begriffe:

Definition 4.2.12.

a) Eine Formelmenge T heisst *voll aufgebläht*, wenn für jede Formel φ und jede Subformel ψ von φ entweder $\psi \in T$ oder $\neg\psi \in T$ gilt.

b) Eine Formelmenge T heisst *impertinent*, falls für eine Formel $\varphi \in T$ auch $\neg\varphi \in T$ gilt.

Der Algorithmus besteht aus vier voneinander unabhängigen Unterprogrammen. In einem ersten Schritt wird eine Formelmenge zu einem aussagenlogischen Tableau erweitert. Die zweite Subroutine konstruiert eine voll aufgeblähte Formelmenge. Eine weitere Prozedur sorgt für die Einhaltung der Bedingung (c) der Definition 4.2.8. Schliesslich wird das resultierende Tableau auf erfüllbare Labels untersucht.

An dieser Stelle wird eine Präzisierung notwendig: Der Algorithmus liefert als Ergebnis ein Prä-Tableau, welches das gesuchte Tableau umfasst. Nur jene Knoten werden von Interesse sein, deren Labels voll aufgeblähte, aussagenlogische Tableaus darstellen. Alle andern Knoten des Prä-Tableaus werden ignoriert. Auch besitzt das Prä-Tableau nicht markierte Kanten, sie werden später ebenfalls von der Bildfläche verschwinden. Ein wichtiges Resultat wird aber bereits im Prä-Tableau sichtbar: Das Prä-Tableau bildet einen Baum, dessen Tiefe bloss polynomial mit $|\varphi|$ wächst.

Bildet eine Formelmenge T kein aussagenlogisches Tableau, so gibt es einen *Zeugen* ψ, der eine der Bedingungen (a)-(c) der Definition des aussagenlogischen Tableaus verletzt. Analog, ist T nicht voll aufgebläht, so existiert eine Subformel ψ eines Elementes φ aus T, wobei weder $\psi \in T$ noch $\neg\psi \in T$. Weiter seien die Formeln aus $\mathcal{L}_n(\Phi)$ einer beliebigen Ordnung unterworfen (z. B. der lexikographischen Ordnung), so dass, falls Zeugen existieren, der kleinste gewählt werden kann.
Nun aber zur Definition des Algorithmus:

Algorithmus 4.2.13. Das $S5_n$-Tableau für φ_0 wird mit folgenden Schritten konstruiert:

1) Setze einen beliebigen Startknoten s_0 als Wurzel des Baumes und sei $L(s_0) = \{\varphi\}$.

2) Wiederhole bis keine der Bedingungen (a)-(d) zutreffen:

   (a) *Herstellen eines aussagenlogischen Tableaus:* Ist s ein Blatt des Baumes, $L(s)$ nicht impertinent, $L(s)$ ist kein aussagenlogisches Tableau und ψ der kleinste Zeuge dieser Tatsache, dann:

      i. Ist ψ von der Form $\neg\neg\psi'$, dann füge den neuen Knoten s' und die Kante (s, s') zum Baum hinzu (Dieses Vorgehen soll im weiteren als Bilden eines Nachfolgers s' von s bezeichnet werden). Setze $L(s') = L(s) \cup \{\psi'\}$.

      ii. Ist ψ von der Form $\psi_1 \wedge \psi_2$, dann bilde einen Nachfolger s' von s und setze $L(s') = L(s) \cup \{\psi_1, \psi_2\}$.

      iii. Ist ψ von der Form $\neg(\psi_1 \wedge \psi_2)$, dann bilde die Nachfolger s_1 und s_2 von s und setze $L(s_i) = L(s) \cup \{\neg\psi_i\}$, für $i = 1, 2$.

(b) *Herstellen eines voll aufgeblähten aussagenlogischen Tableaus:* Ist s ein Blatt des Baumes, $L(s)$ ist nicht impertinent, $L(s)$ ist kein voll aufgeblähtes aussagenlogisches Tableau und ψ ist der kleinste Zeuge dieser Tatsache, dann bilde die Nachfolger s_1 und s_2 und setze $L(s_1) = L(s) \cup \{\neg\psi\}, L(s_2) = L(s) \cup \{\psi\}$.

(c) *Bilden von $\mathcal{K}_i$-Nachfolgeknoten:* Ist s ein Blatt des Baumes, $L(s)$ ein voll aufgeblähtes aussagenlogisches Tableau, nicht impertinent und enthält $L(s)$ Formeln der Form $\neg K_i\psi$, dann bilde für jedes i die Menge $L'(s,i) = \{K_i\psi' : K_i\psi' \in L(s)\} \cup \{\neg K_i\psi' : \neg K_i\psi' \in L(s)\} \cup \{\neg\psi : \neg K_i\psi \in L(s)\}$. Gibt es keinen Vorgänger s'' von s mit $L(s'') = L'(s,i)$, dann bilde den i-Nachfolger s' von s (d. h. füge den Knoten s' und die mit i markierte Kante (s,s') zum Baum hinzu) und setze $L(s') = L'(s,i)$.

(d) *Markieren der Knoten mit „satisfiable“:*[3] Ist s nicht mit „satisfiable“ angeschrieben und gilt $\{K_i\psi, \neg\psi\} \not\subseteq L(s)$ für eine Formel ψ und einen Agenten i, dann markiere s mit „satisfiable“ falls eine der folgenden Bedingungen zutrifft:

i. $L(s)$ ist kein voll aufgeblähtes aussagenlogisches Tableau und es gibt einen mit „satisfiable“ markierten direkten Nachfolger s' von s.

ii. $L(s)$ ist ein voll aufgeblähtes, nicht impertinentes aussagenlogisches Tableau, das keine Formeln der Form $\neg K_i\psi$ enthält.

iii. $L(s)$ ist ein voll aufgeblähtes, nicht impertinentes aussagenlogisches Tableau, $L(s)$ enthält Formeln der Form $\neg K_i\psi$, aber es gilt $L'(s,\psi) = L(s'')$ für einen Vorgänger s'' von s.

iv. $L(s)$ ist ein voll aufgeblähtes aussagenlogisches Tableau und s besitzt direkte Nachfolger, die alle mit „satisfiable“ markiert sind.

3) Ist die Wurzel des Baumes mit „satisfiable“ markiert, dann melde „φ_0 ist erfüllbar“; ansonsten melde „φ_0 ist nicht erfüllbar“.

Diese doch ein wenig umfangreiche Definition schreit nach einem klärenden Beispiel. Bild 4 soll die Vorgänge grafisch veranschaulichen und so den nachfolgenden Worten illuminativ zur Seite stehen. Sei

$$\varphi_0 = (p \wedge \neg(p \wedge \neg q)) \wedge (K_1 p \wedge \neg K_1 K_2 q).$$

[3]Die Verwendung des englischen Begriffes ist Absicht: „satisfiable“ hat a priori nichts mit dem im Kapitel 1 definierten Ausdruck „erfüllbar“ gemeinsam.

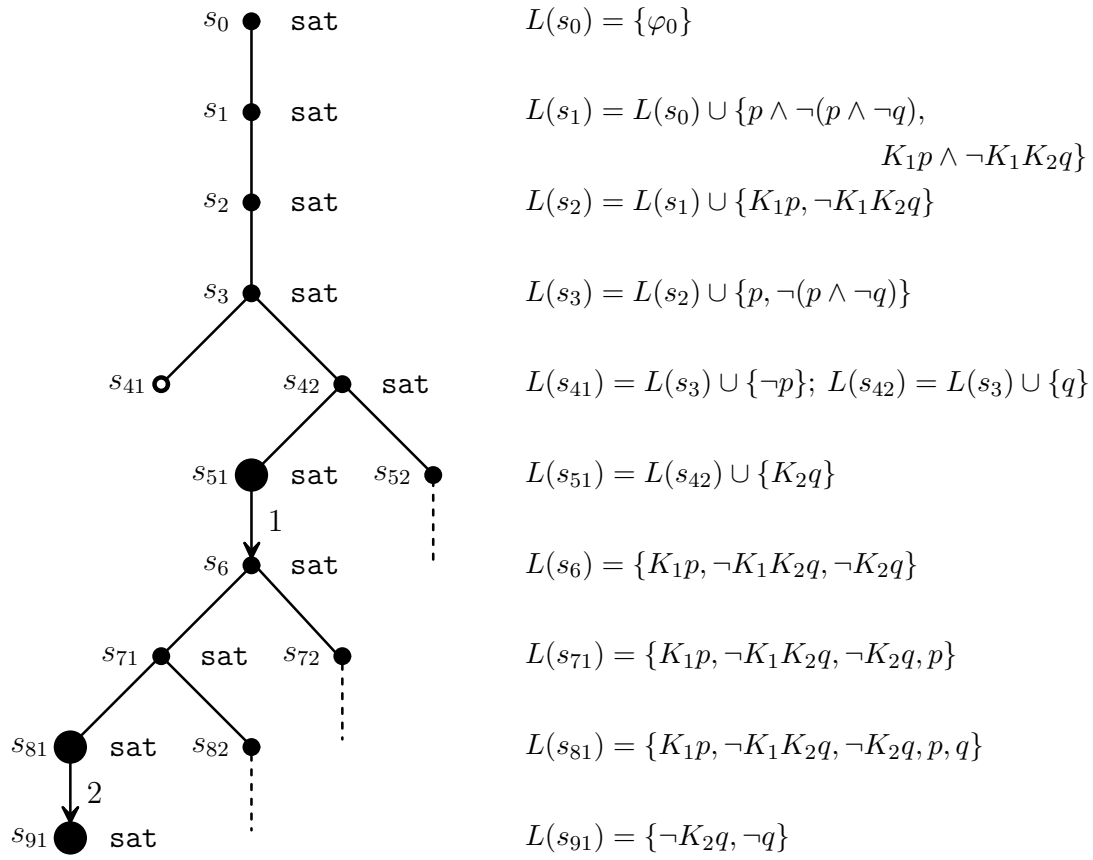


Bild 4: Beispiel eines Prä-Tableaus.

Der Algorithmus konstruiert zu Beginn die Wurzel s_0 mit $L(s_0) = \{\varphi_0\}$. Anschliessend wird die Prozedur 2(a)(ii) dreimal ausgeführt, bis $L(s_3)$ sämtliche Konjutionsglieder enthält. Nun ist $L(s_3)$ kein aussagenlogisches Tableau, bezeugt wird dies durch die Formel $\neg(p \wedge \neg q)$. Mit Schritt 2(a)(iii) werden demnach zwei Nachfolger s_{41} und s_{42} kreiert, mit $\neg p \in L(s_{41})$ und $q \in L(s_{42})$. Damit enthält $L(s_{41})$ aber die Formeln p und $\neg p$, $L(s_{41})$ ist also impertinent. Die Prozeduren 2(a)-(c) werden folglich nicht weiter auf $L(s_{41})$ angewandt, der Knoten s_{41} erhält keine Nachfolger. Ebenso wird das Unterprogramm 2(d) diesen Knoten nie als „satisfiable“ kennzeichnen. Der Algorithmus bearbeitet in der Folge den Knoten s_{42}. Nun ist $L(s_{42})$ zwar ein aussagenlogisches, doch kein voll aufgeblähtes Tableau. Als Zeuge dieser Tatsache dient die Formel $K_2 q$. Mit Schritt 2(c) entstehen folglich die Nachfolger s_{51} und s_{52}, mit $L(s_{51}) = L(s_{42}) \cup \{K_2 q\}$ und $L(s_{52}) = L(s_{42}) \cup \{\neg K_2 q\}$. Damit wird $L(s_{51})$ zu einem voll aufgeblähten, aussagenlogischen Tableau, das zudem nicht impertinent ist. Nun lässt sich die Prozedur 2(c) auf s_{51} anwen-

den. Die einzige Formel der Form $\neg K_i\psi$ in $L(s_{51})$ ist $\neg K_1K_2q$. Somit wird der neue Knoten s_6 über eine 1-Kante zum Baum hinzugefügt, mit $L(s_6) = \{\neg K_2q, K_1p, \neg K_1K_2q\}$. Diese Menge wird durch zweimalige Anwendung der Prozedur 2(b) erneut zu einer voll aufgeblähten erweitert. Zeugen sind jeweils die Formeln p und q, es entstehen so z. B. die Mengen $L(s_{71}), L(s_{81})$ und $L(s_{82})$. Da nun $L(s_{81})$ voll aufgebläht ist, kommt wieder das Unterprogramm 2(c) zum Zuge. Die Formeln der Form $\neg K_i\psi$ sind $\neg K_1K_2q$ und $\neg K_2q$. Somit werden die Mengen $L'(s_{81}, 1) = \{K_1p, \neg K_1K_2q, \neg K_2q\}$ und $L'(s_{81}, 2) = \{\neg q, \neg K_2q\}$ berechnet. Ein kurzer Blick auf die Grafik zeigt, dass $L(s_6) = L'(s_{81}, 1) = \{K_1p, \neg K_1K_2q, \neg K_2q\}$. Daher wird kein neuer 1-Nachfolger von s_{81} hinzugefügt. Anders sieht es mit der Menge $L'(s_{81}, 2)$ aus. Es gibt keinen Vorgänger s_i mit $L(s_i) = L'(s_{81}, 2)$, also entsteht der Knoten s_{91} mit $L(s_{91}) = \{\neg q, \neg K_2q\}$. Die Prozeduren 2(a)-(d) lassen sich nun nicht mehr auf s_{91} anwenden. $L(s_{91})$ ist zwar ein voll aufgeblähtes, aussagenlogisches Tableau und enthält die Formel $\neg K_2q$, doch gilt $L'(s_{91}, 2) = L(s_{91})$, nach 2(c) entsteht kein neuer Knoten. Zudem ist $L(s_{91})$ nicht impertinent, 2(d)(iii) wird s_{91} demnach mit „satisfiable“ markieren (in der Grafik mit `sat` bezeichnet). In der Folge wird 2(d) nacheinander die Knoten $s_{81}, s_{71}, s_6, s_{51}, s_{42}, s_3, s_2, s_1$ und schliesslich s_0 mit „satisfiable“ kennzeichnen.

Natürlich würde der Algorithmus nun an den losen Enden s_{82}, s_{72} und s_{52} weiterarbeiten und eventuell weitere Knoten mit „satisfiable“ markieren. Diese Fortsetzung sei dem interessierten Leser als Übungsaufgabe überlassen. Mit der Meldung „φ_0 ist erfüllbar“ erfüllt der Algorithmus aber die an ihn gestellten Anforderungen.

Einige wichtige Eigenschaften des Algorithmuses sollen nun bewiesen werden. Dazu sind erneut ein paar Begriffe einzuführen.

Definition 4.2.14.

a) Ein Knoten s des Prä-Tableaus heisst *interner Knoten*, falls $L(s)$ kein voll aufgeblähtes, aussagenlogisches Tableau darstellt.

b) Ist ein Knoten s des Prä-Tabelau kein interner Knoten, so wird er *Welt* genannt.

c) Eine Welt s' heisst genau dann $\mathcal{K}_i$-Nachfolger der Welt s, wenn im Prä-Tableau ein Pfad $t_0, \ldots, t_k$ so existiert, dass $s = t_0, s' = t_k$, t_1 ein i-Nachfolger von t_0 und t_j ein interner Knoten mit direkten Nachfolger t_{j+1} ist, für alle j mit $0 < j < k$.

Theorem 4.2.15. *Die Konstruktion des $S5_n$-Tableau terminiert für alle Formeln $\varphi \in \mathcal{L}_n(\Phi)$.*

Beweis. Sei $|\varphi| = m$. Aus der Konstruktion folgt $L(s) \subseteq Sub(\varphi)^+$, für jeden Knoten s im Baum. Daraus ergibt sich $|L(s)| \leq 2m$. Somit wird nach höchstens m Schritten aus 2(a)-(c) entweder eine Welt erreicht oder $L(s)$ wird impertinent. Ist nun s eine Welt und s' ein beliebiger Nachfolgeknoten, so gilt nicht notwendigerweise $dep(L(s')) < dep(L(s))$. Ist s' aber ein i-Nachfolger, so sinkt die modale Tiefe aller Formeln in $L(s')$, die nicht die Form $K_i\psi$ aufweisen. Anders formuliert, einzig die modale Tiefe der Formel der Form $K_i\psi$ wird beim Übergang von $L(s)$ zu $L(s')$ nicht sinken. Man betrachte nun die Welten s, s' und s'', wobei s' ein $\mathcal{K}_i$ Nachfolger von s und s'' ein $\mathcal{K}_j$ Nachfolger von s' ist, mit $i \neq j$. Mit den eben angestellten Überlegungen folgt $dep(L(s'')) < dep(L(s))$.
Zwischen den Welten s und s' liegen maximal m Schritte des Types 2(a) oder 2(b). Die Anwendung der Prozedur 2(c) auf $L(s')$ kann nur dann einen neuen Knoten generieren, falls die Menge $L'(s', j)$ für ein $j \neq i$ geblidet wird. Ansonsten gilt $L'(s', i) = L'(s, i)$. Nach zusätzlichen m Aufrufen der Unterprogramme 2(a) oder 2(b) entsteht die Welt s'' mit $dep(L(s'')) < dep(L(s))$. Diese Gedankengänge zeigen: Nach maximal $2m$ Arbeitsschritten wird die modale Tiefe streng monoton fallen. Nun gilt $dep(L(s_0)) \leq m$, da die modale Tiefe die Länge der Formel sicher nicht übertrifft. Der vom Algorithmus konstruierte Baum besitzt demnach die maximale Länge $2m^2$. Die Konstruktion wird infolgedessen terminieren. q.e.d.

Theorem 4.2.16. *Ein Formel φ ist genau dann $S5_n$-erfüllbar, wenn die $S5_n$-Tableau-Konstruktion die Meldung „φ ist erfüllbar“ ausgibt.*

Beweis. Die $S5_n$ Tableau-Konstruktion liefere die Meldung „φ ist erfüllbar“. Das $S5_n$ Tableau wird wie folgt generiert:

$$\begin{aligned} S &\doteq \{s : s \text{ ist eine mit „satisfiable“ markierte Welt im Prä-Tableau}\} \\ \tilde{L} &\doteq L|_S \text{ (die Einschränkung von } L \text{ auf } S) \end{aligned}$$

Die Relationen $\mathcal{K}_i$ sind etwas umständlicher zu definieren:

$(s,t) \in \mathcal{K}_i$ gdw t ist ein $\mathcal{K}_i$-Nachfolger von s oder
t ist die erste Welt eines Pfades im Prä-Tableau beginnend mit einem Vorgänger s' von s nach s und $L(s') = L'(s, \psi)$ für eine Formel $\neg K_i\psi \in L(s)$.

Ich überprüfe nun Schritt für Schritt die Bedingungen der Definition 4.2.8. Vorerst wird aber erst ein kleiner Hilfssatz bewiesen:

Hilfssatz 4.2.17. *Gilt $(s,t) \in \mathcal{K}_i$, dann stimmen die Mengen $L(s)$ und $L(t)$ in allen Formeln der Form $K_i\psi$ und $\neg K_i\psi$ überein.*

Beweis. Sei s' ein i-Nachfolger von s auf einem Pfad von s nach t. Die Prozedur 2(c) garantiert die Übereinstimmung von $L(s)$ und $L(s')$ auf den Formeln der Form $K_i\psi$ und $\neg K_i\psi$. Weiter gilt $L(s') \subseteq L(t)$, der Algorithmus entfernt nie Formeln innerhalb eines Pfades. Mit $K_i\psi \in L(s)$ folgt also unmittelbar $K_i\psi \in L(t)$. Analoge Überlegungen gelten für $\neg K_i\psi$. Ist andererseits $K_i\psi \in L(t)$, dann folgt aus der Definition der Algorithmus, dass $K_i\psi$ Subformel einer Formel ψ' aus $L(s)$ sein muss. Nun ist $L(s)$ voll aufgebläht, d. h. entweder $K_i\psi \in L(s)$ oder $\neg K_i\psi \in L(s)$. Aus $\neg K_i\psi \in L(s)$ folgt bekanntlich $\neg K_i\psi \in L(t)$, $L(t)$ wäre impertinent und damit keine Welt, im Widerspruch zur Annahme. Also $K_i\psi \in L(s)$, wie gewünscht. Die Argumentation im Falle der Formel $\neg K_i\psi$ verläuft ganz analog. q.e.d.

Nun zum eigentlichen Beweis:

a) Die Mengen $L\tilde{(}s)$ sind nach Konstruktion aussagenlogische Tableaus. Zudem gilt $\varphi \in L(s)$ für ein s, ansonsten die Wurzel des Baumes nicht mit „satisfiable“ gekennzeichnet wäre.

b) Gilt $K_i\psi \in L(s)$ und $(s,t) \in \mathcal{K}_i$, dann mit obigem Hilfssatz $K_i\psi \in L(t)$. Weiter ist $L(t)$ voll aufgebläht, d. h. es gilt entweder $\psi \in L(t)$ oder $\neg\psi \in L(t)$. Mit letzterem wäre t aber nicht mit „satisfiable“ markiert und demnach nicht in S. Also $\psi \in L(t)$.

c) Ist $\neg K_i\psi \in L(s)$, dann existiert offensichtlich ein t mit $\neg\psi \in L(t)$, nach Schritt 2(c).

d) Gilt $K_i\psi \in L(s)$, dann folgt $\neg\psi \notin L(s)$, ansonsten s nicht mit „satisfiable“ markiert wäre. Dann aber $\psi \in L(s)$, da auch $L(s)$ voll aufgebläht ist.

e) Ist $(s,t) \in \mathcal{K}_i$, dann gilt mit dem Hilfssatz: $K_i\psi \in L(s)$ gdw $K_i\psi \in L(t)$.

Die Formel φ besitzt somit ein $S5_n$-Tableau und ist mit Lemma 4.2.11 tatsächlich auch $S5_n$-erfüllbar.

Nun zur Umkehrung. Als roter Faden soll folgende Anleitung dienen: Zu jedem Knoten im Prä-Tableau sei ψ_s die Konjunktion aller Elemente aus $L(s)$. Falls nun s nicht mit „satisfiable“ markiert ist, so soll $\neg\psi_s$ beweisbar sein (d. h. ψ_s ist $S5_n$-inkonsistent). Mit der Vollständigkeit folgte somit ψ_s nicht erfüllbar. Insbesondere, falls die Wurzel nicht mit „satisfiable“ gekennzeichnet ist, so ist $\neg\varphi$ beweisbar und φ nicht erfüllbar. Unter der Höhe h von s verstehe ich die Länge eines Pfades im Prä-Tableau, beginnend in einem Blatt und endend in s. Die Behauptung „s nicht mit ‚satisfiable’ markiert

impliziert $\neg\psi_s$ beweisbar“ wird induktiv über h bewiesen. Sei $h = 0$. In diesem Fall ist s ein Blatt des Baumes. Die Prozedur 2(d) markiert s dann nicht mit „satisfiable“, wenn $L(s)$ impertinent ist oder $\{K_i\psi, \neg\psi\} \subseteq L(s)$. Sind ξ und $\neg\xi$ in $L(s)$ für $\xi \in Sub(\varphi)^+$, so ist ψ_s offenbar $S5_n$-inkonsistent. Mit $\{K_i\psi, \neg\psi\} \subseteq L(s)$ und Axiom **T** folgt ebenso die $S5_n$-Inkonsistenz im zweiten Falle.
Sei $h > 0$ und s ein interner Knoten. Mit 2(d) gilt entweder $\{K_i\psi, \neg\psi\} \subseteq L(s)$ oder kein direkter Nachfolger von s ist mit „satisfiable“ angeschrieben. Der ersten Variante wurde bereits in Rechnung gezogen. Sind andererseits alle direkten Nachfolger nicht mit „satisfiable“ markiert, so folgt mit der Induktionsvoraussetzung die $S5_n$-Inkonsistenz der Formeln $\psi_{s'}$ für alle direkten Nachfolger s' von s. Eine Fallunterscheidung nach der Entstehung der Nachfolger s' zeigt auch die $S5_n$-Inkonsistenz von ψ_s:

- Der Knoten s' entstand aufgrund des Zeugen $\neg\neg\xi$. Damit ist $\psi_{s'} = \psi_s \wedge \xi$. Mit der Induktionsvoraussetzung folgt $S5_n \vdash \neg(\psi_{s'})$ und damit $S5_n \vdash \neg(\psi_s \wedge \xi)$. Bekanntlich ist $\neg\neg\xi \equiv \xi$ eine Tautologie, also $S5_n \vdash \neg(\psi_s \wedge \neg\neg\xi)$ und $S5_n \vdash \neg\psi_s$, da $\neg\neg\xi$ bereits in $L(s)$.

- Der Zeuge hat die Form $\xi_1 \wedge \xi_2$. Somit gilt $\psi_{s'} = \psi_s \wedge \xi_1 \wedge \xi_2$. Aus $S5_n \vdash \neg(\psi_s \wedge \xi_1 \wedge \xi_2)$ folgt unmittelbar $S5_n \vdash \neg\psi_s$, da wiederum $(\xi_1 \wedge \xi_2) \in L(s)$.

- Der Zeuge hat die Form $\neg(\xi_1 \wedge \xi_2)$. Dann gilt $\psi_{s_1} = \psi_s \wedge \neg\xi_1$ und $\psi_{s_2} = \psi_s \wedge \neg\xi_2$. Die Induktionsvoraussetzung liefert $S5_n \vdash \neg(\psi_s \wedge \neg\psi_1)$ und $S5_n \vdash \neg(\psi_s \wedge \neg\psi_2)$. Damit folgt aber $S5_n \vdash \neg(\psi_s \wedge \neg(\psi_1 \wedge \psi_2))$, folglich $S5_n \vdash \neg\psi_s$.

- Der Zeuge ξ ist Subformel eines Elementes aus $L(s)$, aber weder $\xi \in L(s)$, noch $\neg\xi \in L(s)$. Sei $\psi_{s_1} = \psi_s \wedge \xi$ und $\psi_{s_2} = \psi_s \wedge \neg\xi$. Dann gilt mit der Induktionsvoraussetzung $S5_n \vdash \neg(\psi_s \wedge \xi)$ und $S5_n \vdash \neg(\psi_s \wedge \neg\xi)$. Mit etwas aussagenlogischem Räsonieren folgt erst $S5_n \vdash \psi_s \Rightarrow \xi$, $S5_n \vdash \psi_s \Rightarrow \neg\xi$ und dann $S5_n \vdash (\xi \vee \neg\xi) \Rightarrow \neg\psi_s$. Offenbar liefert dies $S5_n \vdash \neg\psi_s$.

Sei s nun eine nicht mit „satisfiable“ markierte Welt. Somit gilt entweder $\{K_i\xi, \neg\xi\} \subseteq L(s)$ oder es existiert ein Nachfolger s' im Prä-Tableau, der nicht mit „satisfiable“ gekennzeichnet ist. Erstgenannte Möglichkeit wurde bereits besprochen, es treffe daher die zweite zu. Nach der Konstruktion des Algorithmus' gibt es eine Formel $\neg K_i\psi \in L(s)$ so, dass $L'(s,i) = L(s')$ für ein i. Mit der Induktionsvoraussetzung ist $\psi_{s'}$ $S5_n$-inkonsistent. Es sei $L(s') = \{K_i\alpha_1, \ldots, K_i\alpha_n, \neg K_i\beta_1, \ldots, \neg K_i\beta_m, \neg\psi\}$. Dann gilt

$$S5_n \vdash \neg(K_i\alpha_1 \wedge \cdots \wedge K_i\alpha_n \wedge \neg K_i\beta_1 \wedge \cdots \wedge \neg K_i\beta_m \wedge \neg\psi).$$

Mit derselben Argumentation wie auf Seite 19 folgt

$$\mathrm{S5}_n \vdash K_i\alpha_1 \Rightarrow (\ldots(K_i\alpha_n \Rightarrow (\neg K_i\beta_1 \Rightarrow (\ldots(\neg K_i\beta_m \Rightarrow K_i\psi)))))$$

und

$$\mathrm{S5}_n \vdash (K_i\alpha_1 \wedge \cdots \wedge K_i\alpha_n \wedge \neg K_i\beta_1 \wedge \cdots \wedge K_i\beta_m) \Rightarrow K_i\psi,$$

folglich

$$\mathrm{S5}_n \vdash \neg(K_i\alpha_1 \wedge \cdots \wedge K_i\alpha_n \wedge \neg K_i\beta_1 \wedge \cdots \wedge K_i\beta_m \wedge \neg K_i\psi)$$

Mit $\{K_i\alpha_1, \ldots, K_i\alpha_n, \neg K_i\beta_1, \ldots, \neg K_i\beta_m, \neg K_i\psi\} \subseteq L(s)$ folgt unmittelbar die $\mathrm{S5}_n$-Inkonsistenz von ψ_s. q.e.d.

Angewandt auf das beschriebene Beispiel ergeben obige Resultate folgendes Tableau:

$$\begin{aligned} S &= \{s_{51}, s_{81}, s_{91}\} \\ \tilde{L} &= L|_S \text{ (die Einschränkung von } L \text{ auf } S) \\ \mathcal{K}_1 &= \{(s_{51}, s_{81}), (s_{81}, s_{81})\} \\ \mathcal{K}_2 &= \{(s_{81}, s_{91}), (s_{91}, s_{91})\} \end{aligned}$$

Nebenbei sei erwähnt, dass der Algorithmus 1.7.3 eine alternative Variante des Vollständigkeitsbeweises von $\mathrm{S5}_n$ ermöglicht. Sei dazu φ gültig. Man füttert nun die $\mathrm{S5}_n$-Tableau-Konstruktion mit der Formel $\neg\varphi$. Man erhält die Antwort „$\neg\varphi$ ist nicht erfüllbar“, ansonsten wäre $\neg\varphi$ erfüllbar und φ nicht gültig. Nach dem Beweis zum Theorem 4.2.16 ist demnach $\neg\neg\varphi$ beweisbar, und damit auch φ.

Theorem 4.2.18. *Die $S5_n$-Erfüllbarkeit einer Formel φ ist in PSPACE entscheidbar.*

Beweis. Das Verfahren prüft, ob die Wurzel des Baumes der $\mathrm{S5}_n$-Tableau-Konstruktion für φ mit „satisfiable“ markiert ist. Der Algorithmus beginnt bei der Wurzel und sucht sich nach dem Prinzip des „depth-first search“ den Weg zu einem Blatt. Die Labels der besuchten Knoten werden in einer Liste X abgelegt. Werden die maximal $2m$ Elemente, mit $m = |\varphi|$, von $Sub(\varphi)^+$ geordnet und durchnummeriert, so kann $L(s)$ für jeden Knoten s durch einen Bit-String der Länge $2m$ repräsentiert werden: Die i-te Formel der Aufzählung ist genau dann in $L(s)$, wenn das i-te Bit im String 1 ist. Die Längen der Pfade im Prä-Tableau sind durch $2m^2$ beschränkt, somit $|X| \leq 4m^3$.

Die Markierung eines Knotens s ist nur von seinen direkten Nachfolgern abhängig, deren er höchstens m Stück besitzt. Das Prüfen der Nachfolger auf die Markierung verlangt die Überprüfung der Nachfolger der Nachfolger usw.. Ein exponentieller Speicherbedarf wird dadurch verhindert, dass die Überprüfung von unten nach oben ausgeführt und nach jeder Berechnung der benötigte Speicherplatz wieder freigegeben wird. Einzig die Information, welche Nachfolger noch zu überprüfen sind, wird in weiteren m Bits abgelegt.

Bezeichne h erneut die Länge eines Pfades vom Blatt zur Wurzel. Die Markierung eines Knotens s der Höhe h kann nun mit einem Speicherbedarf von $(3h+2)m+|X|$ Bits erruiert werden. Der Beweis verläuft induktiv über h.

Sei $h = 0$. Der Knoten s ist daher ein Blatt des Baumes. Ist $L(s)$ impertinent, so wird s nicht im „satisfiable“ markiert. Dasselbe gilt für den Fall, dass s eine Welt mit $\{K_i\psi, \neg\psi\} \subseteq L(s)$ ist, für eine Formel ψ und einen Agenten i. Andererseits, erfüllt $L(s)$ eine der Bedingungen 2(d)(ii)-(iii), so wird s mit „satisfiable“ gekennzeichnet. All diese Anforderungen können mit Kenntnis von $L(s)$ überprüft werden, der Speicherbedarf beträgt daher $2m$.

Sei $h > 0$ und s ein interner Knoten. Die $S5_n$-Tableau-Konstruktion kreiert ein bis zwei Nachfolger von s. Nun wird s dann mit „satisfiable“ markiert, wenn $\{K_i\psi, \neg\psi\} \not\subseteq L(s)$ und einer der beiden Nachfolger bereits mit „satisfiable“ gekennzeichnet wurde. Nach der Induktionsvorausetzung lässt sich letzterer Test mit $(3h+2)m+|X|$ Bits Speicherplatz ausführen, wobei, wie erwähnt, der zur Berechnung benötigte Speicher nachträglich freigegeben und erneut verwendet wird. $L(s)$ seinerseits benötigt wiederum $3m$ Bits, der gesamte Speicherbedarf beträgt zusammen also $(3(h+1)+2)m+|X|$ Bits. Ist s eine Welt, so kann diese bis zu m direkte Nachfolger besitzen. Somit müssen nun m Knoten nach obigem Muster überprüft werden, weitere Unterschiede zum internen Knoten ergeben sich nicht.

Da $h \leq 2m^2$ und $|X| \leq 4m^3$, liegt der Platzbedarf des Entscheidungsverfahren demnach in $\mathcal{O}(m^4)$. q.e.d.

4.3 Low cost $S5_n$

4.3.1 Beschränkte Verschachtelungstiefe

Nach den Ergebnissen der beiden vorhergehenden Abschnitte ist das Erfüllbarkeitsproblem von S5 in NP, dasjenige von $S5_n$, für ≥ 2, jedoch in PSPACE. Die Beweise trugen der unbegrenzten Verschachtelungstiefe der K_i Operatoren Rechnung und nutzten diese auch aus. Was geschieht hingegen bei beschränkter modaler Tiefe der Formeln aus $\mathcal{L}_n(\Phi)$? Sei zu diesem

Zwecke $\mathcal{L}_n^k(\Phi)$ die Menge aller Formel aus $\mathcal{L}_n(\Phi)$ mit $dep(\varphi) \leq k$ für eine Konstante k. Die Überlegungen zum Beweis von Theorem 4.2.15 erzwingen in diesem Fall eine hübsche Konsequenz:

Korollar 4.3.1. *Das Erfüllbarkeitsproblem der Logik $S5_n$ bezüglich der Sprache $\mathcal{L}_n^k(\Phi)$ ist NP-vollständig.*

Beweis. Das Erfüllbarkeitsproblem der Aussagenlogik liefert unmittelbar die untere Grenze der Komplexität. Wie im Beweis von 4.2.15 dargelegt wurde, sind zur Senkung der modalen Tiefe um eine Einheit $2m$ Arbeitsschritte und 3 Welten notwendig, mit $|\varphi| = m$. Wird bei unbeschränkter Verschachtelung aus dem konstruierten Tableau wie beschrieben ein $S5_n$-Modell generiert, so hat dieses die Form eines Baumes mit maximaler graphentheoretischer Tiefe $3m$ und Verzweigungsgrad m, die Grösse des Modells wächst somit exponentiell mit $|\varphi|$. Beschränkt k nun aber die modale Tiefe der Formel φ, so besitzen die Pfade im Baum nunmehr die Höchstlänge $3k$. Bei unveränderten Verzweigungsgrad ergibt dies eine Strukturgrösse von $|\varphi|^{3k}$, ein Polynom in der Länge von φ. Damit ist φ genau dann erfüllbar, wenn φ in einer Struktur der Grösse $|\varphi|^{3k}$ erfüllbar ist. Wie im Fall von S5 wird nun eine beliebige Struktur M aus $\mathcal{M}_n^{rst}$ der Gösse $|\varphi|^{3k}$ geraten und in polynomialer Zeit überprüft, ob M tatsächlich φ erfüllt. q.e.d.

Kapitel 5

Die Komplexität von $\mathbf{S5}_n^C$

Der Operator C erhöht die Ausdrucksstärke der bisher betrachteten Sprachen ungemein, sind doch nun Aussagen über alle in einer Struktur erreichbaren Zustände möglich. Dieser Gewinn an Ausdruckskraft hat aber auch seine Kehrseite: die Komplexität des Erfüllbarkeitsproblems steigt auf EXPTIME.

5.1 Gegenbeispiel Nr.2

Im Kapitel 4 konnte ich für die Logik $\mathrm{S5}_n$ ohne Allgemeinwissen einen Algorithmus aus PSPACE zur Lösung des Erfüllbarkeitsproblems angeben. Die Prozedur konstruierte baumartige Strukturen, deren graphentheoretische Tiefe nur polynomial mit $|\varphi|$ anwuchs. War φ eine erfüllbare Formel aus $\mathcal{L}_n(\Phi)$, so musste φ in einer dieser Strukturen erfüllbar sein. Dieser Zusammenhang ist in Sprachen mit Allgemeinwissen nicht länger gültig. Nach einer Arbeit von Fischer und Immerman [FL79] gibt es Formeln aus $\mathcal{L}_n^C(\Phi)$, die sich nur in Strukturen erfüllen lassen, deren Pfade exponentiell in der Länge der Formel anwachsen.

Ich beginne meine Betrachtungen zu diesem Problem mit der Logik K_n^C. Anschliessend wird das Resultat schrittweise auf $\mathrm{S5}_n^C$ übertragen, wobei $n \geq 2$ gilt.

Theorem 5.1.1. *Sei $\Phi = \{p_0, \ldots, p_{m-1}\}$ eine beliebige Menge von primitiven Propositionen. Es gibt dann eine Formel σ_m^K mit $|\sigma_m^K| \in \mathcal{O}(m^2)$, die K^C-erfüllbar ist, aber jede Struktur, welche σ_m^K erfüllt, besitzt einen Pfad der Länge $2^m - 1$.*

Beweis. Die primitiven Propositionen seien mit $\{p_0, \ldots, p_{m-1}\}$ bezeichnet. Die Idee ist nun, mit diesen Propositionen einen binären m-Bit Zähler zu formen. Dabei bezeichne p_0 das Bit mit der tiefsten, p_{m-1} dasjenige mit der

höchsten Ordnung. Gilt $p_i \equiv \mathbf{true}$, so ist das i-te Bit des Zählers gleich 1, gilt $\neg p_i \equiv \mathbf{true}$, so ist das i-te Bit gleich 0. Die zu konstruierende Formel σ_m^K soll den Zähler zwingen, in einer Folge von Zuständen sämtliche Werte von 0 bis 2^{m-1} anzunehmen.

Tat der Leser es mir gleich und durchlitt in früheren Jahren eine Grundausbildung in handfester Informatik, so wird ihm die folgende Eigenschaft des binären Zählens vertraut erscheinen: Sind $\mathbf{c} = c_{m-1} \dots c_0$ und $\mathbf{d} = d_{m-1} \dots d_0$ zwei Zeichenketten des Zählers, so gilt $\mathbf{d} = \mathbf{c} + 1$ genau dann, wenn ein $k \leq m-1$ so existiert, dass für alle $i < k$ $c_i = 1$ und $d_i = 0$, $c_k = 0$ und $d_k = 1$, sowie $c_j = d_j$ für alle $k+1 \leq j \leq m-1$ gilt.
Ein Beispiel:

$$\begin{array}{cccccccccccccc} & & & \overset{m-1}{\downarrow} & & & & \overset{k}{\downarrow} & & & & & & \overset{0}{\downarrow} \\ \mathbf{c} & = & 1 & 0 & 1 & 0 & 0 & 1 & 1 & \cdots & 1 & 1 & 1 \\ \mathbf{d} & = & 1 & 0 & 1 & 0 & 1 & 0 & 0 & \cdots & 0 & 0 & 0 \end{array}$$

Obiger Sachverhalt führt nun zur Definition der Formel $\sigma_m^K \equiv \sigma_{m1}^K \wedge \sigma_{m2}^K \wedge \sigma_{m3}^K \wedge \sigma_{m4}^K$, wobei gilt:

$$\begin{aligned} \sigma_{m1}^K &\doteq C(\neg K \neg \mathbf{true}) \\ \sigma_{m2}^K &\doteq (\neg p_0 \wedge \dots \wedge \neg p_{m-1}) \\ \sigma_{m3}^K &\doteq \bigwedge_{i=0}^{m-1} C((\bigwedge_{j=0}^{i-1} p_j) \Rightarrow ((p_i \Rightarrow K \neg p_i) \wedge (\neg p_i \Rightarrow K p_i))) \\ \sigma_{m4}^K &\doteq \bigwedge_{i=0}^{m-1} C((\bigvee_{j=0}^{i-1} \neg p_j) \Rightarrow ((p_i \Rightarrow K p_i) \wedge (\neg p_i \Rightarrow K \neg p_i))) \end{aligned}$$

Wie üblich ist im Fall $i = 0$ die Konjunktion $\bigwedge_{j=0}^{i-1} p_j$ in σ_{m3}^K äquivalent zu **true**, die Disjunktion $\bigvee_{j=0}^{i-1} \neg p_j$ in σ_{m4}^K äquivalent zu **false**.

Die Intentionen der Formeln $\sigma_{m1}^K, \dots \sigma_{m4}^K$ wird im Verlauf des Beweises ersichtlich. Angenommen, es gilt $(M, s_0) \models \sigma_m^K$. Dank σ_{m1}^K existiert eine Folge von nicht notwendigerweise verschiedenen Zuständen $s_0, \dots, s_l$ beliebiger Länge mit $(s_i, s_{i+1}) \in \mathcal{K}$. Ich werde nun zeigen, dass diese Folge aus $2^m - 1$ verschiedenen Zuständen besteht, wobei die Propositionen $p_0, \dots, p_{m-1}$ im Zustand s_i den Index i kodieren. Mit Formel σ_{m2}^K steht der Zähler bei s_0 tatsächlich auf 0. Ist nun s von s_0 erreichbar und gilt $(s, t) \in \mathcal{K}$, dann hat p_i aufgrund der Formeln σ_{m3}^K und σ_{m4}^K genau dann denselben Wahrheitswert in s und t, wenn $(M, s) \models \neg p_j$ für ein $j < i$. Mit den erwähnten Eigenschaften des binären m-Bit Zählers beschreiben s und t somit aufeinanderfolgende Zeichenketten. Mit einer Induktion über i zeige ich nun, dass s_i den Wert i kodiert.

- Ist $i = 0$, so folgt die Behauptung aus $(M, s_0) \models \sigma^K_{m2}$.
- Sei nun $i > 0$. Der Zustand s_{i-1} kodiert nach Induktionsvoraussetzung den Index $i - 1$. Gilt $(M, s_{i-1}) \models \neg p_0$, so folgt mit σ^K_{m3} demnach $(M, s_i) \models p_0$. Die Formel σ^K_{m4} erzwingt weiter ein Übereinstimmen der Wahrheitswerte aller p_i mit $i > 0$. Somit kodiert s_i den Zählerstand $(i - 1) + 1 = i$. Gilt andererseits $(M, s_{i-1}) \models p_0$, so ändert sich der Wahrheitswert von p_0 erneut, d. h. $(M, s_i) \models \neg p_0$. Die Wahrheitswerte der $p_i, i > 0$, erfahren nun, beginnend mit p_1, ebenfalls eine Änderung (Bildlich: der Übertrag wird weitergereicht) und zwar bis ein $p_j, j > 0$, auftritt mit $(M, s_{i-1}) \models \neg p_j$. Dessen Wert wird in s_i zwar noch auf **true** gesetzt, die Werte aller p_k mit $k > j$ bleiben aber bei s_i unverändert, bedingt durch σ^K_{m4}. Dies bedeutet nichts weiter, als dass s_i den Wert $(i - 1) + 1 = i$ kodiert.

Die Zustände $s_0, \ldots, s_{2^m-1}$ sind folglich paarweise verschieden, da sie sich durch die Belegung der Propositionen p_i unterscheiden. Eine Struktur, die σ^K_m erfüllt, hat demnach folgende Form:

$$
\begin{aligned}
S &\doteq \{s_0, \ldots, s_{2^m-1}\} \\
\pi_{s_i}(p_j) &\doteq \begin{cases} \textbf{true} & : \text{das } j\text{-te Bit der Binärdarstellung von } i \text{ ist gleich 1.} \\ \textbf{false} & : \text{sonst} \end{cases} \\
\mathcal{K} &\doteq \{(s_i, s_{i+1}) : i < 2^m - 1\} \cup \{(s_{2^m-1}, s_0)\}
\end{aligned}
$$

q.e.d.

Soviel zur Logik K^C. Der direkte Übergang zu $\mathrm{S5}^C_n$ wäre etwas gar gewagt, ich betrachte erst die Logik T^C.

Die Formel σ^K_m ist leider in reflexiven Strukturen unerfüllbar. Ausdrücke der Form $p \Rightarrow K\neg p$ schaffen in Strukturen aus $\mathcal{M}^r$ Probleme: Gilt $(M, s) \models p \wedge (p \Rightarrow K\neg p)$, so folgt der Widerspruch $(M, s) \models \neg p$, da bekanntlich $(s, s) \in \mathcal{K}$. Die Formeln $p \Rightarrow K\neg p$ müssen daher diesem Sachverhalt angepasst werden. Zu diesem Zwecke wird eine neue primitive Proposition p_Δ eingeführt, die einen Wechsel des Zustandes anzeigen soll: Gilt $(M, s_i) \models p_\Delta$, so soll $(M, s_{i+1}) \models \neg p_\Delta$ gelten, falls $s_0, \ldots, s_l$ eine Folge von Zuständen darstellt und $i < l$.

Mit einem nunmehr geschärften Auge kann die Behauptung für die Logik T^C formuliert werden:

Theorem 5.1.2. *Sei $\Phi = \{p_0, \ldots, p_{m-1}\}$ eine beliebige Menge von primitiven Propositionen. Es gibt dann eine Formel σ^T_m mit $|\sigma^T_m| \in \mathcal{O}(m^2)$, die T^C-erfüllbar ist, aber jede Struktur aus $\mathcal{M}$, welche σ^T_m erfüllt, besitzt einen Pfad der Länge $2^m - 1$.*

Beweis. Sei $\sigma_m^T \equiv \sigma_{m1}^T \wedge \sigma_{m2}^T \wedge \sigma_{m3}^T \wedge \sigma_{m4}^T$ mit

$$\sigma_{m1}^T \doteq C((p_\Delta \Rightarrow \neg K \neg p_\Delta))$$
$$\sigma_{m2}^T \doteq (\neg p_0 \wedge \cdots \wedge \neg p_{m-1})$$
$$\sigma_{m3}^T \doteq \bigwedge_{i=0}^{m-1} C\Big[(\bigwedge_{j=0}^{i-1} p_j) \Rightarrow \Big(\big((p_\Delta \wedge p_i) \Rightarrow K(\neg p_\Delta \Rightarrow \neg p_i)\big) \wedge \big((p_\Delta \wedge \neg p_i) \Rightarrow K(\neg p_\Delta \Rightarrow p_i)\big) \wedge \big((\neg p_\Delta \wedge p_i) \Rightarrow K(p_\Delta \Rightarrow \neg p_i)\big) \wedge \big((\neg p_\Delta \wedge \neg p_i) \Rightarrow K(p_\Delta \Rightarrow p_i)\big)\Big)\Big]$$
$$\sigma_{m4}^T \doteq \bigwedge_{i=0}^{m-1} C\Big[(\bigvee_{j=0}^{i-1} \neg p_j) \Rightarrow \Big(\big((p_\Delta \wedge p_i) \Rightarrow K(\neg p_\Delta \Rightarrow p_i)\big) \wedge \big((p_\Delta \wedge \neg p_i) \Rightarrow K(\neg p_\Delta \Rightarrow \neg p_i)\big) \wedge \big((\neg p_\Delta \wedge p_i) \Rightarrow K(p_\Delta \Rightarrow p_i)\big) \wedge \big((\neg p_\Delta \wedge \neg p_i) \Rightarrow K(p_\Delta \Rightarrow \neg p_i)\big)\Big)\Big]$$

Angenommen, es gilt $(M, s_0) \models \sigma_m^T$. Aufgrund der Formel σ_{m1}^T existiert eine Zustandsfolge $s_0, \ldots, s_l$ beliebiger Länge mit $(s_i, s_{i+1}) \in \mathcal{K}$ und alternierendem Wahrheitswert für p_Δ. Somit gilt z. B. $(M, s_i) \models p_\Delta$ für alle geraden und $(M, s_i) \models \neg p_\Delta$ für alle ungeraden i. Ist s von s_0 erreichbar, gilt $(s, t) \in \mathcal{K}$ und besitzt p_Δ auf s und t unterschiedliche Wahrheitswerte, so folgt mit den Formeln σ_{m3}^T und σ_{m4}^T, dass s und t aufeinanderfolgende Zählerzustände repräsentieren. Wieder kodiert s_i den Index i, wie eine Induktion über i zeigt.

- Der Fall $i = 0$ ist klar.
- Sei $i > 0$. Mit der Induktionsvoraussetzung kodiert der Zustand s_{i-1} den Index $i - 1$. Gilt nun $(M, s_{i-1}) \models \neg p_0$, so gibt es unabhängig vom Wert der Proposition $p_\Delta$bei s_{i-1} einen Zustand s_i mit $(M, s_i) \models p_0$, einem gegenüber s_{i-1} geändertem Wahrheitswert für p_Δ und unveränderten Wahrheitswerten für alle p_j mit $j > 0$. Der Zustand s_i kodiert demnach den Index i. Auch im Fall $(M, s_{i-1}) \models p_0$ verläuft die Argumentation analog zum Theorem 5.1.1. Der Unterschied besteht einzig im Wechsel des Wertes der Proposition p_Δ.

Damit existiert erneut eine Zustandsfolge $s_0, \ldots, s_{2^m-1}$ mit paarweise verschiedenen Gliedern. Die Formel σ_m^T wird somit in einer Struktur $M =$

$(S, \pi, \mathcal{K})$ erfüllt, wobei:

$$\begin{aligned}
S &\doteq \{s_0, \ldots, s_{2^m-1}\} \\
\pi_{s_i}(p_j) &\doteq \begin{cases} \textbf{true} & : \text{das } j\text{-te Bit der Binärdarstellung von } i \text{ ist gleich 1.} \\ \textbf{false} & : \text{sonst} \end{cases} \\
\pi_{s_i}(p_\Delta) &\doteq \begin{cases} \textbf{true} & : i \text{ gerade} \\ \textbf{false} & : i \text{ ungerade} \end{cases} \\
\mathcal{K} &\doteq \{(s_i, s_{i+1}) : i < 2^m - 1\} \cup \{(s_{2^m-1}, s_0)\}
\end{aligned}$$

Die Definition von $\pi_{s_i}(p_\Delta)$ ist willkürlich, sie kann selbstverständlich auch andersrum erklärt werden. q.e.d.

Bevor ich nun zu $\mathrm{S5}_n^C$-Strukturen übergehen kann, muss ich den Leser auf eine weitere Schwierigkeit aufmerksam machen. Man stelle sich $(M, s_0) \models \sigma_m^T$, für ein $M \in \mathcal{M}^{rt}$, und eine Sequenz $s_0, s_2, s_3, \ldots$ mit $(s_i, s_{i+1}) \in \mathcal{K}$ vor. Nach obigen Schlussfolgerungen kodiert s_0 den Wert 0, während s_2 den Wert 2 repräsentiert. Mit der Transitivität von $\mathcal{K}$ folgt (s_0, s_2). Gemäss den Formeln σ_{m3}^T und σ_{m4}^T kodiert s_2 nun den Index 1, ein Widerspruch.

Die Hinzunahme eines weiteren Agenten löst das Problem. Jedes Auftreten des Operators K in der Formel σ_m^T wird durch die Kombination K_1K_2 ersetzt. Die Formel $\sigma_m^{S5_2}$ ist somit die Konjunktion der Formeln $\sigma_{m1}^{S5_2}, \sigma_{m2}^{S5_2}, \sigma_{m3}^{S5_2}$ und $\sigma_{m4}^{S5_2}$ wobei

$$\sigma_{m1}^{S5_2} \doteq C((p_\Delta \Rightarrow \neg K_1K_2\neg p_\Delta))$$

$$\sigma_{m2}^{S5_2} \doteq (\neg p_0 \wedge \cdots \wedge \neg p_{m-1})$$

$$\sigma_{m3}^{S5_2} \doteq \bigwedge_{i=0}^{m-1} C\Big[(\bigwedge_{j=0}^{i-1} p_j) \Rightarrow$$

$$\Big(\big((p_\Delta \wedge p_i) \Rightarrow K_1K_2(\neg p_\Delta \Rightarrow \neg p_i)\big) \wedge \big((p_\Delta \wedge \neg p_i) \Rightarrow K_1K_2(\neg p_\Delta \Rightarrow p_i)\big) \wedge$$

$$\big((\neg p_\Delta \wedge p_i) \Rightarrow K_1K_2(p_\Delta \Rightarrow \neg p_i)\big) \wedge \big((\neg p_\Delta \wedge \neg p_i) \Rightarrow K_1K_2(p_\Delta \Rightarrow p_i)\big)\Big)\Big]$$

$$\sigma_{m4}^{S5_2} \doteq \bigwedge_{i=0}^{m-1} C\Big[(\bigvee_{j=0}^{i-1} \neg p_j) \Rightarrow$$

$$\Big(\big((p_\Delta \wedge p_i) \Rightarrow K_1K_2(\neg p_\Delta \Rightarrow p_i)\big) \wedge \big((p_\Delta \wedge \neg p_i) \Rightarrow K_1K_2(\neg p_\Delta \Rightarrow \neg p_i)\big) \wedge$$

$$\big((\neg p_\Delta \wedge p_i) \Rightarrow K_1K_2(p_\Delta \Rightarrow p_i)\big) \wedge \big((\neg p_\Delta \wedge \neg p_i) \Rightarrow K_1K_2(p_\Delta \Rightarrow \neg p_i)\big)\Big)\Big]$$

Die weitere Argumentation folgt den Beweisen der Theoreme 5.1.1 und 5.1.2. Gilt demnach $(M, s_0) \models \sigma_m^{S5_2}$, so kodiert der Zustand s_0 wieder den Wert 0. Ist weiter s von s_0 erreichbar und gilt $(s,t) \in \mathcal{K}_1 \circ \mathcal{K}_2$, dann repräsentieren s und t aufeinanderfolgende Zählerwerte. Der Unterschied zum Theorem 5.1.2 liegt darin, nunmehr Kanten aus dem Relationenprodukt $\mathcal{K}_1 \circ \mathcal{K}_2$ zu betrachten. Nach den Ausführungen zu dessen Eigenschaften in 4.2.1 glaube ich, hier auf weitere Erklärungen verzichten zu können, zumal noch folgende Tatsache zutrifft: Selbst wenn $\mathcal{K}_1$ und $\mathcal{K}_2$ reflexiv, symmetrisch und transitiv sind, so ist das Produkt $\mathcal{K}_1 \circ \mathcal{K}_2$ im Allgemeinen bloss noch reflexiv. Das Problem der Reflexivität wurde in Theorem 5.1.2 behandelt und bereitet damit keine weiteren Bauchschmerzen. Sicher kann die $S5_2^C$-Struktur M, die $\sigma_m^{S5_2}$ erfüllt, neben der Menge $\{s_0, \ldots, s_{2^m-1}\}$ weitere Zustände besitzen: diejenigen zwischen s_i und s_{i+1}, für alle $i < 2^m - 1$. Genauer: die Zustände s_i' mit $s_i \mathcal{K}_1 s_i' \mathcal{K}_2 s_{i+1}$. Der Zustand s_i' muss aber nicht notwendigerweise von s_i oder s_{i+1} verschieden sein. Selbstverständlich sind aber nach den Erläuterungen zum Theorem 5.1.2 die Zustände s_i und s_{i+1} weiterhin nicht identisch. Eine $\sigma_m^{S5_2}$ erfüllende Struktur M hat somit folgende Form:

$$
\begin{aligned}
S &\doteq \{s_0, s_0', s_1, s_1', \ldots, s_{2^m-1}, s_{2^m-1}'\} \\
\pi_{s_i}(p_j) &\doteq \begin{cases} \textbf{true} & : \text{das } j\text{-te Bit der Binärdarstellung von } i \text{ ist gleich 1.} \\ \textbf{false} & : \text{sonst} \end{cases} \\
\pi_{s_i}(p_\Delta) &\doteq \begin{cases} \textbf{true} & : i \text{ gerade} \\ \textbf{false} & : i \text{ ungerade} \end{cases} \\
\pi_{s_i'}(p) &\doteq \pi_{s_i}(p_j), p \in \{p_0, \ldots, p_{m-1}, p_\Delta\} \\
\mathcal{K}_1 &\doteq (\{(s_i, s_i') : i \leq 2^m - 1\})^{rst} \\
\mathcal{K}_2 &\doteq (\{(s_i', s_{i+1}) : i < 2^m - 1\} \cup \{(s_{2^m-1}', s_0)\})^{rst}
\end{aligned}
$$

Damit ist das nachstehende Korrolar bewiesen:

Korollar 5.1.3. *Sei $\Phi = \{p_1, \ldots, p_m\}$ eine beliebige Menge von primitiven Propositionen. Es gibt dann eine Formel $\sigma_m^{S5_2}$ mit $|\sigma_m^{S5_2}| \in \mathcal{O}(m^2)$, die $S5_2^C$-erfüllbar ist, aber jede Struktur aus $\mathcal{M}$, welche $\sigma_m^{S5_2}$ erfüllt, besitzt einen Pfad der Mindestlänge $2^m - 1$.*

5.2 Die Sonderrolle von $S5^C$

Mit dem Gegenbeispiel 5.1.3 kann das Erfüllbarkeitsproblem der Logik $S5_n^C$, für $n \geq 2$, nicht mit einem Algorithmus aus PSPACE gelöst werden. Für

den Fall $n = 1$ bleibt die Situation ungeklärt. Die Antwort ist jedoch leicht anzugeben. Im Anschluss an Lemma 2.2.9 wurde dem Operator C die Möglichkeitsrelation $\mathcal{C} = (\mathcal{K}_1 \cup \cdots \cup \mathcal{K}_n)^t$ zugewiesen. Ist $n = 1$, so ergibt sich entsprechend $\mathcal{C} = (\mathcal{K})^t$. Da $\mathcal{K}$ aber bereits transitiv ist, folgt $\mathcal{K} = (\mathcal{K})^t = \mathcal{C}$. Demnach wird die Ausdrucksstärke der Sprache $S5^C$ gegenüber S5 nicht erhöht; alle über $\mathcal{C}$-Kanten erreichbaren Zustände sind auch über $\mathcal{K}$-Kanten erreichbar. Das Erfüllbarkeitsproblem von $S5^C$ bleibt damit in NP.

Korollar 5.2.1. *Die $S5^C$-Erfüllbarkeit einer Formel φ ist in NP entscheidbar.*

Auch für das Erfüllbarkeitsproblem der Logik $S4^C$ treffen obige Bemerkungen zu, da die Möglichkeitsrelation $\mathcal{K}$ des Operators K in dieser Situation ebenfalls transitiv ist.

Korollar 5.2.2. *Die $S4^C$-Erfüllbarkeit einer Formel φ ist in NP entscheidbar.*

5.3 Untere EXPTIME Schranke für $S5_2^C$

Nach dem Theorem 3.2.14 ist das Erfüllbarkeitsproblem der Logik K^C hart in EXPTIME. Dazu wurde jede von einer alternierenden Turingmaschine AT akzeptierte Zeichenkette x mit einer Funktion f so nach K^C übertragen, dass galt: $x \in L(AT)$ genau dann, wenn $f(x)$ erfüllbar. Die Formel $f(x)$ hatte die Gestalt

$$f(x) = h \wedge C(g_1 \wedge g_2 \wedge g_3 \wedge g_4 \wedge g_5 \wedge g_6).$$

Ich nenne $f(x)$ fortan $\varphi^K_{\mathbf{AT},x}$.

Die Formel $\varphi^K_{\mathbf{AT},x}$ durchlebt nun dieselben Abänderungen wie σ^K_m aus Theorem 5.1.1. Erst wird die neue primitive Proposition p_Δ derart eingewoben, dass ihr Wert entlang eines C-Pfades alterniert. Dieser Schachzug behebt die mit der Reflexivität angeschwemmten Probleme. Anschliessend ersetzen die Operatoren K_1K_2 jedes Auftreten von K in $\varphi^K_{\mathbf{AT},x}$. Nach diesen Transformationen resultiert die Formel

$$\varphi^{S5}_{\mathbf{AT},x} = h' \wedge C \bigwedge_{i=1}^{7} g'$$

mit $h' = h, g_1' = g_1, g_2' = g_2$ und

$$g_3' : \bigwedge_{i=0}^{m} \bigwedge_{\sigma\in\Gamma} \Big(\big(\neg H_i \wedge P_{i,\sigma} \wedge p_\Delta \Rightarrow K_1K_2(\neg p_\Delta \Rightarrow P_{i,\sigma})\big) \\ \wedge \big(\neg H_i \wedge P_{i,\sigma} \wedge \neg p_\Delta \Rightarrow K_1K_2(p_\Delta \Rightarrow P_{i,\sigma})\big)\Big)$$

$$g_4' : \\ \bigwedge_{i=1}^{m-1} \Big(\Big(H_i \wedge p_\Delta \Rightarrow K_1K_2\big((\neg p_\Delta \Rightarrow H_{i-1} \wedge \neg H_{i+1}) \vee (\neg p_\Delta \Rightarrow \neg H_{i-1} \wedge H_{i+1})\big)\Big) \\ \wedge \Big(H_i \wedge \neg p_\Delta \Rightarrow K_1K_2\big((p_\Delta \Rightarrow H_{i-1} \wedge \neg H_{i+1}) \vee (p_\Delta \Rightarrow \neg H_{i-1} \wedge H_{i+1})\big)\Big) \\ \wedge \Big(\neg H_{i-1} \wedge \neg H_{i+1} \wedge \neg p_\Delta \Rightarrow K_1K_2(p_\Delta \Rightarrow \neg H_i)\Big) \\ \wedge \Big(\neg H_{i-1} \wedge \neg H_{i+1} \wedge p_\Delta \Rightarrow K_1K_2(\neg p_\Delta \Rightarrow \neg H_i)\Big)\Big) \wedge \neg H_0 \wedge \neg H_m$$

$$g_5' : \\ \bigwedge_{i=1}^{m-1} \bigwedge_{\sigma\in\Gamma} \bigwedge_{q\in U} \Big(\Big(H_i \wedge P_{i,\sigma} \wedge Q_q \wedge p_\Delta \Rightarrow \\ \Big(\bigwedge_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',R)\in\delta}} \neg K_1K_2\neg(\neg p_\Delta \Rightarrow H_{i+1} \wedge P_{i,\sigma'} \wedge Q_{q'}) \\ \wedge \bigwedge_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',L)\in\delta}} \neg K_1K_2\neg(\neg p_\Delta \Rightarrow H_{i-1} \wedge P_{i,\sigma'} \wedge Q_{q'})\Big)\Big) \\ \Big(H_i \wedge P_{i,\sigma} \wedge Q_q \wedge \neg p_\Delta \Rightarrow \\ \Big(\bigwedge_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',R)\in\delta}} \neg K_1K_2\neg(p_\Delta \Rightarrow H_{i+1} \wedge P_{i,\sigma'} \wedge Q_{q'}) \\ \wedge \bigwedge_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',L)\in\delta}} \neg K_1K_2\neg(p_\Delta \Rightarrow H_{i-1} \wedge P_{i,\sigma'} \wedge Q_{q'})\Big)\Big)$$

g_6' :

$$\bigwedge_{i=1}^{m-1} \bigwedge_{\sigma\in\Gamma} \bigwedge_{q\in Q\setminus U} \Big(\big(H_i \wedge P_{i,\sigma} \wedge Q_q \wedge p_\Delta \Rightarrow \bigvee_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',R)\in\delta}} \neg K_1 K_2 \neg (\neg p_\Delta \Rightarrow H_{i+1} \wedge P_{i,\sigma'} \wedge Q_{q'}) \vee \bigvee_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',L)\in\delta}} \neg K_1 K_2 \neg (\neg p_\Delta \Rightarrow H_{i-1} \wedge P_{i,\sigma'} \wedge Q_{q'}) \big)$$

$$\wedge \big(H_i \wedge P_{i,\sigma} \wedge Q_q \wedge \neg p_\Delta \Rightarrow \bigvee_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',R)\in\delta}} \neg K_1 K_2 \neg (p_\Delta \Rightarrow H_{i+1} \wedge P_{i,\sigma'} \wedge Q_{q'}) \vee \bigvee_{\substack{q',\sigma':\\ (q,\sigma,q',\sigma',L)\in\delta}} \neg K_1 K_2 \neg (p_\Delta \Rightarrow H_{i-1} \wedge P_{i,\sigma'} \wedge Q_{q'}) \big) \Big)$$

und zusätzlich

$$g_7' : (p_\Delta \Rightarrow \neg K_1 K_2 p_\Delta) \wedge (\neg p_\Delta \Rightarrow \neg K_1 K_2 \neg p_\Delta)$$

Wird nun die Zeichenkette x von der Turingmaschine AT akzeptiert, so gilt $(M,s) \models \varphi^K_{\mathbf{AT},x}$ für ein K^C-Modell $M = (S,\pi,\mathcal{K})$. Die Struktur M' entsteht aus M durch folgende Konstruktion:

$$\begin{aligned}
S' &\doteq S \cup \{z_{(s,t)} : (s,t) \in \mathcal{K}\} \\
\left.\begin{aligned} \pi'_s(P_{i,\sigma}) &\doteq \pi_s(P_{i,\sigma}) \\ \pi'_s(H_i) &\doteq \pi_s(H_i) \\ \pi'_s(Q_q) &\doteq \pi_s(Q_q) \end{aligned}\right\} &\text{ für } s \in S, 0 \le i \le m, \sigma \in \Gamma, q \in Q \\
\pi'_s(p_\Delta) &\doteq \begin{cases} \mathbf{true} & : \exists t \text{ mit } (s,t) \in \mathcal{K} \\ \mathbf{false} & : \exists s \text{ mit } (s,t) \in \mathcal{K} \end{cases} \\
\left.\begin{aligned} \pi'_{z_{(s,t)}}(P_{i,\sigma}) &\doteq \pi_s(P_{i,\sigma}) \\ \pi'_{z_{(s,t)}}(H_i) &\doteq \pi_s(H_i) \\ \pi'_{z_{(s,t)}}(Q_q) &\doteq \pi_s(Q_q) \\ \pi'_{z_{(s,t)}}(p_\Delta) &\doteq \pi'_s(p_\Delta) \end{aligned}\right\} &\text{ für } z_{(s,t)} \in S' \setminus S, 0 \le i \le m, \sigma \in \Gamma, q \in Q \\
\mathcal{K}'_1 &\doteq (\{(s, z_{(s,t)}) : (s,t) \in \mathcal{K}\})^{rst} \\
\mathcal{K}'_2 &\doteq (\{(z_{(s,t)}, t) : (s,t) \in \mathcal{K}\})^{rst}
\end{aligned}$$

Nach den Ausführungen zum Gegenbeispiel 5.1.3 ist klar, dass $(M', s) \models \varphi^{S5}_{\mathbf{AT},x}$.

Gilt nun $(M', s_0) \models \varphi^{S5}_{\mathbf{AT},x}$ für eine Struktur $M' \in \mathcal{M}_2^{rst}$. Mit Theorem 2.2.12 darf M' als endlich betrachtet werden. Aus $(M', s_0) \models g'_7$ folgt die Existenz einer endlichen Folge von Zuständen $s_0, \ldots, s_n$ mit $(s_i, s_{i+1}) \in \mathcal{K}_1 \circ \mathcal{K}_2$ und $(M', s_i) \models p_\Delta$ genau dann wenn $(M', s_{i+1}) \models \neg p_\Delta$, für $0 \leq i < n$. Auf diese Folge lässt sich nun die Argumentation aus Theorem 3.2.14 übertragen. Für jeden Zustand $s_i, 0 \leq i \leq n$, beschreiben somit die bei ihm wahren primitiven Propositionen eine Konfiguration der Turingmaschine AT. Wiederum lässt sich aus diesen eine vereinfachte, x akzeptierende Spur extrahieren. Ich verzichte auf weitere Details, da sich keine Unterschiede zum Beweis aus Theorem 3.2.14 ergeben.

Die obigen Ausführungen beweisen demnach das folgende Theorem:

Theorem 5.3.1. *Das Erfüllbarkeitsproblem der Logik $S5_n^C$,$n \geq 2$, ist EXPTIME-hart.*

5.4 Obere EXPTIME Schranke für $\mathbf{S5}_n^C$

Pratt stellt in [Pra79] einen Algorithmus zur Lösung des Erfüllbarkeitsproblems der Logik PDL vor. Dieses Verfahren kann mit leichten Abänderungen auch für den Fall der Logik $\mathrm{S5}_n^C$ verwendet werden.

Algorithmus 5.4.1 ($\mathbf{S5}_n^C$-Modellkonstruktion für φ).

1) Bilde alle Teilmengen von $Sub(\varphi)_C^+$ und setze $\mathcal{K}_i = \emptyset$ für alle $i \in \{1, \ldots, n\}$.

2) Konstruiere $\mathcal{W} \subseteq \mathcal{P}(Sub(\varphi)_C^+)$, wobei genau dann $W \in \mathcal{W}$ wenn für alle $\psi, \psi' \in Sub(\varphi)_C^+$ und $i \in \{1, \ldots, n\}$ gilt

   (a) $\psi \in W$ impliziert $\neg\psi \notin W$,

   (b) $\neg\psi \in W$ impliziert $\psi \notin W$,

   (c) $\psi \wedge \psi' \in W$ impliziert $\psi, \psi' \in W$,

   (d) $\psi \vee \psi' \in W$ impliziert $\psi \in W$ oder $\psi' \in W$,

   (e) $K_i\psi \in W$ impliziert $\psi \in W$,

   (f) $(K_i(\psi \Rightarrow \psi') \wedge K_i\psi) \in W$ impliziert $K_i\psi' \in W$,

   (g) $E\psi \in W$ impliziert $K_i\psi \in W$, für alle $i \in \{1, \ldots, n\}$,

   (h) $C\psi \in W$ impliziert $E(\psi \wedge C\psi) \in W$.

3) Für alle $W \in \mathcal{W}$ seien $K_i(W) \doteq \{K_i\varphi : K_i\varphi \in W\}$ und $\overline{K_i(W)} \doteq \{\underline{\neg K_i\varphi} : \underline{\neg K_i\varphi} \in W\}$. Setze $\mathcal{K}_i \doteq \{(W,V) : K_i(W) = K_i(V)$ und $\overline{K_i(W)} = \overline{K_i(V)}\}$

4) Falls $W \in \mathcal{W}$, dann suche zu jeder Formel $\neg K_i\varphi \in W$ ein V mit $(W,V) \in \mathcal{K}_i$ und $\neg\varphi \in V$. Existiert für eine Formel $\neg K_i\psi$ kein solcher Zustand V, dann setze $\mathcal{W} \leftarrow \mathcal{W} \setminus \{W\}$ und $\mathcal{K}_i \leftarrow \mathcal{K}_i \setminus \{(U,U') : U = W$ oder $U' = W\}$.

5) Führe für alle $W \in \mathcal{W}$ und $\neg C\psi \in W$ folgende Prozedur aus:

   - Setze $m = 1$ und *path* = **false**
   - Für $m = 1$ bis $2^{(n+3)|\varphi|}$ wiederhole
     - Suche eine von W via einen Pfad der Länge m erreichbare Menge S mit $\neg\psi \in S$ und $S \in \mathcal{W}$.
     - Falls ein Weg gefunden wurde, setze *path* = **true** und verlasse die Schlaufe.
   - Falls *path* = **false**, setze $\mathcal{W} \leftarrow \mathcal{W} \setminus \{W\}$ und $\mathcal{K}_i \leftarrow \mathcal{K}_i \setminus \{(U,U') : U = W$ oder $U' = W\}$, für alle $i \in \{1,\ldots,n\}$.

6) Gilt $\varphi \in W$ für ein $W \in \mathcal{W}$, so melde „φ ist erfüllbar“.

Theorem 5.4.2. *Eine Formel φ ist genau dann $S5_n^C$ erfüllbar, wenn die Modellkonstruktion „φ ist erfüllbar“ meldet.*

Beweis. Sei φ erfüllbar. Nach dem Beweis zu Theorem 2.2.12 besitzt φ ein endliches Modell M mit maximal $2^{(n+3)|\varphi|}$ Zuständen. Die Mengen $W_s \doteq \{\psi : \psi \in Sub(\varphi)_C^+$ und $(M,s) \models \psi\}$ sind Teilmengen von $Sub(\varphi)_C^+$ und liegen nach Schritt (2) in $\mathcal{W}$. Offenbar gilt $\varphi \in W_t$ für ein $W_t \in \mathcal{W}$, da φ erfüllbar ist. Weiter besitzt die nach (3) vorhandene Struktur N die maximale Anzahl von Kanten, M ist demnach Substruktur von N. Zudem ist M Modell von φ und verletzt daher keine der Bedingungen in den Schritten (4) und (5). Die vom Algorithmus in diesen Arbeitsabläufen entfernten Kanten und Zustände liegen somit nicht in M. Damit ist M aber auch in der resultierenden Struktur N' eingebettet und es gilt $\varphi \in W$ für ein $W \in \mathcal{W}$. Der Algorithmus wird demnach „φ ist erfüllbar“ melden.

Melde andererseits der Algorithmus „φ ist erfüllbar“. Aus der konstruierten Struktur N' lässt sich leicht ein Modell $M = (S, \pi, \mathcal{K}_1, \ldots, \mathcal{K}_n)$ für φ

gewinnen:

$$\begin{aligned} S &= \mathcal{W} \\ \pi_W(p) &= \begin{cases} \textbf{true} & \text{falls } p \in W \\ \textbf{false} & \text{sonst} \end{cases} \\ \mathcal{K}_i' &= \mathcal{K}_i \end{aligned}$$

Damit ergibt sich für alle $\psi \in Sub(\psi)_C^+$, erzwungen durch die Konstruktion des Algorithmus': $(M, W) \models \psi$ gdw $\psi \in W$. Mit $\varphi \in V$ für ein $V \in \mathcal{W}$, ist φ erfüllbar.

Die $\mathcal{K}_i$ sind nach Schritt (3) sicher Äquivalenzrelationen. Offenbar liegt der präsentierte Algorithmus in EXPTIME, variert doch bereits die zum Aufschreiben aller Teilmengen von $Sub(\psi)_C^+$ benötigte Zeit exponentiell mit $|\varphi|$. q.e.d.

5.5 Konsequenzen

Mit den Ergebnissen der beiden vorhergehenden Abschnitte folgt :

Korollar 5.5.1. *Das Erfüllbarkeitsproblem der Logik $S5_n^C, n \geq 2$, ist EXPTIME-vollständig.*

Weiter ist nach dem Beweis zum Theorem 5.3.1 bereits ein einfach verschachtelter C Operator zur Konstruktion der Formel $\varphi_{\mathbf{AT},x}^{S5}$ ausreichend. Ein zu Theorem 4.3.1 analoges Resultat ist folglich auszuschliessen. Allerdings präsentierte Luca Alberucci anlässlich einer Konferenz im Rahmen des Nationalfondsprojektes „Deduction and Inference“ ein Fragment $\mathcal{LA}$ von $\mathcal{L}_n^C(\Phi)$, dessen Beweisbarkeitsproblem in PSPACE lag. Ich verzichte aus Rücksicht gegenüber aktuellen Arbeiten von Herrn Alberucci auf einer ausführliche Darstellung dieses bisher unveröffentlichten Resultates. Ein kurzer Abriss sei mir, so hoffe ich, indes gestattet.

Es sei $\mathcal{LA} \doteq \mathcal{L}_n(\Phi) \cup \{C\varphi : \varphi \in \mathcal{L}_n(\Phi)\}$. Der Kalkül LAC umfasst für alle endlichen Formelmengen Γ, alle Formeln A,B und alle primitiven

Propositionen $P \in \Phi$ folgende Schlussregeln:

$$(Ax)\quad \frac{}{\Gamma, \top} \qquad (id)\quad \frac{\Gamma, \top}{\Gamma, P, \neg P}$$

$$(\wedge)\quad \frac{\Gamma, A \quad \Gamma, B}{\Gamma, A \wedge B} \qquad (\vee)\quad \frac{\Gamma, A, B}{\Gamma, A \vee B}$$

$$(\neg E)\quad \frac{\Gamma, \neg EA, \neg K_i A}{\Gamma, \neg EA} \qquad (E)\quad \frac{\Gamma, K_i A \quad (\forall i \in \{1, \ldots, n\})}{\Gamma, EA}$$

$$(K_i)\quad \frac{\neg\Gamma, A}{\neg K_i\Gamma, K_i A, \Sigma} \qquad (C)\quad \frac{A}{CA, \Sigma}$$

Das Beweisbarkeitsproblem des Fragmentes $\mathcal{LA}$ liegt in PSPACE, wie bereits aus der Definition des Kalküls sichtbar wird. Die Vollständigkeit von LAC wurde von Alberucci nachgewiesen [Al99].

Ist die Beweisbarkeit von $\varphi \in \mathcal{LA}$ in PSPACE entscheidbar, so auch die Frage nach der Erfüllbarkeit von $\neg\varphi$. Das Erfüllbarkeitsproblem des Fragmentes $\mathcal{LA}' \doteq \mathcal{L}_n(\Phi) \cup \{\neg C\psi : \psi \in \mathcal{L}_n(\Phi)\}$ liegt somit in PSPACE.

Literaturverzeichnis

[Al99] L.Alberucci. *Vortragsnotizen*, Kontakt: albe@iam.unibe.ch

[BC94] D.P.Bovet, P.Crescenzi. *Introduction to the theory of complexity*, Prentice Hall, 1994

[CKS81] A.K.Chandra, D.Kozen, L.J.Stockmeyer. *Alternation*, Journal of the ACM, 28:114-133, 1981

[Co71] S.A.Cook. *The complexity of theorem proving procedures*, Proc. ACM Symposium on Theory of Computing, 151-158, 1971

[FHMV95] R.Fagin, J.Y.Halpern, Y.Moses, M.Y.Vardi. *Reasoning about knowlegde*, MIT Press Cambridge, Massachusetts, 1995.

[Fi99] M.Fitting. *A simple propositional S5 tableau system*, Annals of pure and applied logic 96: 107-115, 1999

[FIm87] M.J.Fischer, N.Immerman. *Interpreting logics of knowledge in propositional dynamic logic with converse*, Information processing letters 25:175-181, 1987

[FL79] M.J.Fischer, R.E.Ladner. *Propositional dynamic logic of regular programms*, Journal of computer and system sciences, 18:194-211, 1979

[GarJo79] M.R.Garey, D.S.Johnson. *Computers und intractability*, Freeman, 1979

[Ge94] J.Geanakoplos. *Common knowledge*, Handbook of game theory, Volume 2, Elsevier science B.V., 1994

[HM92] J.Y.Halpern, Y.Moses. *A guide to completness and complexity for modal logics of knowledge and belief*, Artificial Intelligence 54, 1992, 311-379.

[Kri63] S.Kripke. *Semantical analysis of modal logic I: Normal propositional calculi*, Zeitschrift für Mathematische Logik und Grundlagen der Mathematik, 9:67-96, 1963.

[Lad77] R.E.Ladner. *The computational complexity of provability in systems of modal propositional logic*, SIAM Journal on Computing, 6(3):467-480,1977

[Pra79] V.R.Pratt. *Models of program logic*, Proc. 20th IEEE Conference on Foundations of Computer Science, 1979

[Pra81] V.R.Pratt. *A decidable μ-calculus: Preliminary report*, 22nd IEEE Symp. on foundations of computer science, 421-427, 1981

[Pre92] A.Prestel. *Einführung in die Mathematische Logik und Modelltheorie*, Vieweg Studium, 1992.

[Smu68] R. M. Smullyan. *First-Order Logic*, Springer Verlag, 1968

[SM73] L.J.Stockmeyer, A.R.Meyer. *Word problems requiring exponential time*, Proc. ACM Symposium on Theory of Computing, 1-9, 1973