Integration und Uniformierung von Methoden des tableaubasierten Theorembeweisens

Zur Erlangung des akademischen Grades eines

Doktors der Naturwissenschaften

der Fakultät für Informatik
der Universität Karlsruhe (TH)

genehmigte

Dissertation

von

Bernhard Beckert

aus Pforzheim

Tag der mündlichen Prüfung: 3. Juli 1998
Erster Gutachter Prof. Dr. Peter H. Schmitt
Zweiter Gutachter Prof. Dr. Jacques Calmet

The invention of tableau systems will continue, simply because they are easier to think of than other formulations.

Melvin Fitting (1998)

Vorwort

Veröffentlichung

Einige der in dieser Dissertation vorgestellten Resultate sind bereits in Zeitschriftenartikeln und Konferenzbeiträgen veröffentlicht:

(Beckert & Goré, 1997) – ein Tableaukalkül mit gemischten Variablen für Modallogiken (Abschnitt 4.3.7).

(Beckert & Hartmer, 1998) – Tableaukalküle für Fragmente der Mengentheorie (Abschnitt 3.8).

(Beckert & Gabbay, 1998) – Fasern von Tableaukalkülen (Kapitel 6).

Danksagung

Viele haben mir in der einen oder anderen Weise geholfen, während ich diese Dissertation geschrieben habe; ohne ihre Unterstützung wäre sie nicht entstanden.

Mein Dank gilt:

- Prof. Peter H. Schmitt für seine hervorragende Betreuung und dafür, daß er mir die Möglichkeit gegeben hat in seiner Gruppe mit ihrer freundlichen und produktiven Atmosphäre zu arbeiten;
- Prof. Jacques Calmet für fruchtbare Diskussionen und dafür, daß er sich bereit erklärt hat, Zweitgutachter bei meiner Promotion zu sein;
- meinen Kollegen und Freunden Wolfgang Ahrendt, Reiner Hähnle, Christian Pape und Joachim Posegga für all ihre Unterstützung und viele anregende Diskussionen;
- Prof. Dov Gabbay dafür, daß er mir die Möglichkeit gegeben hat, seine Gruppe am Imperial College zu besuchen, daß er mir einen Teil seines riesigen Wissens auf allen Teilgebieten der Logik vermittelt hat und daß er mich angeregt hat, die Methode des Faserns auf Tableaukalküle anzuwenden;
- Rajeev Goré, denn, mit ihm zusammenzuarbeiten, ist immer produktiv – auch wenn er auf der anderen Seite des Planeten ist;
- Ulrike Hartmer, die die Details der Kalküle für Fragmente der Mengentheorie ausgearbeitet hat, die in dieser Dissertation beschrieben sind;

– all den vielen Anderen, die mir einen Rat oder Hinweis gegeben haben, zu denen Prof. Domenico Cantone, Uwe Egly, Guido Governatori und Gernot Stenz gehören;

– der Fakultät für Informatik der Universität Karlsruhe für die guten Arbeitsbedingungen, die sie geschaffen hat;

– der Deutschen Forschungsgemeinschaft, die – als Teil des Schwerpunktprogramms *Deduktion* – die Projekte finanziert hat, an denen ich während der letzten Jahre gearbeitet habe;

– dem Department of Computing des Imperial College in London für seine Einladung – mein Besuch dort war produktiv, und ich habe sehr davon profitiert;

– der Europäischen Union für die Finanzierung meines Aufenthalts am Imperial College als Teil des Programms *Training and Mobility of Researchers* (TMR) und meiner Teilnahme an mehreren Workshops und Konferenzen als Teil der COST Aktion *Many-valued Logics*;

– und ganz besonders meinen Eltern für all ihre Unterstützung und Ermunterung über die Jahre.

Karlsruhe, im Juni 1998

Bernhard Beckert

Bemerkung. Im folgenden benutze ich das Pronomen „wir“; das ist keine *Pluralis majestatis*, sondern bezieht sich auf uns beide: mich und den Leser.

Inhaltsverzeichnis

Tabellenverzeichnis

Abbildungsverzeichnis

1 Einleitung

Logicians have but ill defined
As rational the human kind.
Logic, they say, belongs to man,
But let them prove it if they can.

— OLIVER GOLDSMITH

1.1 Motivation

Das Automatische Beweisen – und besonders das tableaubasierte Theorembeweisen – hat einen Zustand erreicht, an dem es an der Schwelle zum erfolgreichen Einsatz in realen Anwendungen steht. Jedoch ist bis jetzt der Übergang von einer Technik, die hauptsächlich in der Forschung eingesetzt wird, zu einem routinemäßig in der Praxis eingesetzten Werkzeug nicht vollzogen worden.

Ein Hinderungsgrund ist, daß sehr viel Arbeit dafür aufgewendet wurde, ausgefeilte und leistungsfähige Theorembeweiser für bestimmte Logiken zu entwickeln (besonders für die Prädikatenlogik erster Stufe), die nicht auf bestimmte Anwendungen zugeschnitten sind. Um diese Systeme verwenden zu können, müssen Probleme aus einem bestimmten Anwendungsbereich in die von dem Beweiser unterstütze Logik transformiert werden.

Ein anderer und möglicherweise erfolgreicherer Ansatz ist, ein spezialisiertes, dediziertes Beweissystem für eine gegebene Anwendung zu entwickeln, das eine besonders für die Anwendung geeignete Logik unterstützt, und dabei die besonderen Eigenschaften sowohl der Logik als auch der Anwendungen auszunutzen, um die Leistungsfähigkeit des Systems zu steigern.

Eine notwendige Voraussetzung dafür ist, daß uniforme Methoden für die Konstruktion von effizienten dedizierten Beweisprozeduren für beliebige Logiken zur Verfügung stehen, so daß diese nicht jeweils völlig neu entwickelt werden müssen. Es besteht jedoch eine Lücke zwischen den beiden Hauptteilen, in welche sich das vorhandene Wissen auf dem Gebiet des Automatischen Beweisens einteilen läßt: Einerseits sind viele verschiedene Kalküle für viele verschiedene Logiken vorgestellt worden; diese sind jedoch typischerweise nur von theoretischem Interesse und nicht für die Implementierung geeignet. Andererseits sind Beweisprozeduren für wenige wichtige Logiken entwickelt worden – vorherrschend sind die klassische Aussagenlogik und die Prädikatenlogik erster Stufe –, einschließlich einer großen Zahl besonderer Techniken und Verfeinerungen, die diese Prozeduren effizienter machen.

Das Ziel dieser Arbeit ist es, die oben beschriebene Lücke zu schließen: Um dieses Ziel zu erreichen und die Konstruktion effizienter dedizierter tableaubasierter Theorembeweiser zu ermöglichen, werden drei sich ergänzende Ansätze verfolgt:

1. die Verallgemeinerung und uniforme Beschreibung von Tableaukalkülen, ihrer Verfeinerungen und von Methoden zu ihrer Verbesserung;

2. die Entwicklung uniformer Methoden für die Konstruktion effizienter deterministischer Beweisprozeduren aus (nicht-deterministischen) Tableaukalkülen;

3. die (uniforme) Integration sowohl verschiedener Tableaukalküle als auch von Tableaukalkülen mit speziellen Methoden zur Problemlösung in bestimmten Domänen (Theorieschließen).

Die ersten beiden dieser Ansätze können bei allen Logiken angewendet werden, während der dritte im wesentlichen bei Logiken Anwendung findet, die Erweiterungen der klassischen Logik sind (z. B. mehrwertige, modale und temporale Logiken), und nur bedingt bei Einschränkungen der klassischen Logik (z. B. lineare Logik und Relevanzlogik).

Die uniformen Methoden, die im folgenden vorgestellt werden, vereinfachen den Entwurf effizienter tableaubasierter Beweisprozeduren. Da viele der jeweiligen Vorbedingungen für die Anwendbarkeit der Methoden rein syntaktisch und leicht zu überprüfen sind, kann man sich – als ein zukünftiges Ziel – ein (zumindest teilweise) automatisches Meta-Deduktionssystem vorstellen, das auf einen gegebenen Tableaukalkül uniforme Techniken zu seiner Verbesserung und zur Konstruktion einer effizienten Beweisprozedur anwendet oder das sogar einen Tableaukalkül für eine gegebene Logik entwickelt.

1.2 Resultate und Struktur dieser Arbeit

Kapitel 1 In diesem einführenden Kapitel wird die Wichtigkeit uniformer Methoden zum Entwurf effizienter Tableaukalküle diskutiert und motiviert; die Resultate und die Struktur dieser Arbeit werden zusammengefaßt; der Begriff des Tableaukalküls wird eingeführt; die wesentlichen Eigenschaften von Tableaukalkülen und ihre Unterschiede zu anderen Arten von Kalkülen werden erklärt; und schließlich wird ein kurzer Überblick über die Historie von Tableaukalkülen gegeben.

Kapitel 2 Logische Systeme (oder kurz: Logiken) werden in sehr allgemeiner Weise definiert (möglichst wenige Beschränkungen bezüglich Syntax und Semantik werden gemacht). Der Begriff des Terms und, darauf aufbauende, der Substitution und der Unifikation werden eingeführt. Als Beispiele für die Beschreibung von Logiken in diesem allgemeinen Rahmen werden die Prädikatenlogik erster Stufe, Modallogiken und zwei Fragmente der quantorenfreien Mengentheorie vorgestellt.

Kapitel 3 Eine uniforme Beschreibung von Tableaukalkülen und ihrer syntaktischen und semantischen Eigenschaften wird gegeben – wie beispielsweise analytisch, monoton, saturierend, bzw. korrekt, vollständig, beweiskonfluent zu sein – ohne Beschränkung auf bestimmte Logiken oder Normalformen. Allgemeine Kriterien um zu überprüfen, ob ein Kalkül diese Eigenschaften hat, werden vorgestellt (einschließlich Kriterien für Korrektheit und Vollständigkeit). Eine wichtige Klasse von sich „wohlverhaltenden“ Kalkülen wird identifiziert, die *gutartig* genannt werden. Diese Klasse erweist sich als sehr wichtig, weil (a) Gutartigkeit eine Vorbedingung ist für die Anwendbarkeit vieler der uniformen Methoden zur Steigerung der Effizienz von Tableaukalkülen, die im folgenden beschrieben werden, und (b) gutartige Kalküle für die meisten Logiken existieren. Als Beispiele werden gutartige Kalküle für die Prädikatenlogik erster Stufe und für Modallogiken vorgestellt. Außerdem werden Kalküle für die in Kapitel 2 eingeführten Fragmente der Mengentheorie definiert; es wird gezeigt, daß diese Kalküle effizienter sind, als die in der Literatur beschriebenen.

Kapitel 4 Techniken werden vorgestellt zur Verbesserung von Tableaukalkülen für das Automatische Beweisen, so daß darauf aufbauende Beweisprozeduren effizienter sind. Insbesondere werden Methoden verallgemeinert, die sich beim tableaubasierten Beweisen in Prädikatenlogik erster Stufe als nützlich erwiesen haben – darunter (a) das Konzept der starren Variablen, die Terme repräsentieren und „bei Bedarf“ während der Beweissuche instantiiert werden können, und (b) die Technik der universellen Variablen, bei der Variablen *alle* Terme zugleich repräsentieren, und (c) deren Kombination. Die Beziehung von Kalkülen mit starren und mit universellen Variablen zu Grundkalkülen (d. h. variablenfreien Kalkülen) wird erklärt, einschließlich *Lifting*-Methoden zur Konstruktion von Kalkülen mit starren und universellen Variablen aus Grundkalkülen. Eine verbesserte Version der Skolemisierung wird vorgestellt, bei der, statt neue Skolemsymbole einzuführen, jeder Prämisse, aus der die Existenz von Objekten mit bestimmten Eigenschaften abgeleitet werden kann, ihr eigenes Symbol zugeordnet wird. Als Beispiel für Kalküle, die diese Techniken ausnutzen, werden Kalküle mit gemischten starren und universellen Variablen für Prädikatenlogik erster Stufe und für Modallogiken vorgestellt. Andere Methoden zur Verbesserung von Tableaukalkülen, die in diesem Kapitel in verallgemeinerter Form beschrieben werden, sind die Technik der *lokalen Lemmata*, das *Prunen* redundanter Tableauäste und die Einführung zusätzlicher Tableauregelschemata.

Kapitel 5 Dieses Kapitel ergänzt Kapitel 4, in dem Methoden zur Verbesserung von Tableaukalkülen beschrieben werden, die es erlauben, kürzere Beweise zu konstruieren. Hier ist das Thema, wie effizient nach einem Beweis in dem kleiner gewordenen Suchraum gesucht werden kann. Techniken zur Umsetzung eines (nichtdeterministischen) Tableaukalküls in eine deterministische Beweisprozedur werden diskutiert und analysiert. Ein allgemeines Konzept der *Regularität* für beliebige Kalküle und der Begriff der *Gewichtsordnungen* werden eingeführt; es wird gezeigt, daß diese geeignet sind zur Konstruktion einer deterministischen Prozedur zur Tiefensuche nach Beweisen für beliebige gutartige Kalküle mit starren Variablen (obwohl bekannt war, daß solche Prozeduren existieren, war es ein bisher ungelöstes Problem, tatsächlich eine praktikable Prozedur anzugeben).

Kapitel 6 Die Technik des *Faserns* (fibring), die es erlaubt, logische Systeme zu kombinieren, indem man ihre Semantiken kombiniert, wird auf Tableaukalküle erweitert. Eine Methode zur uniformen Konstruktion eines korrekten und vollständigen Tableaukalküls für die kombinierte Logik aus Kalkülen für die Komponenten-Logiken wird vorgestellt. Da Kalküle für die meisten „Basislogiken“ schon zur Verfügung stehen, erhält man so Kalküle für viele „komplexe“ Logiken, die durch das Fasern von Basislogiken entstehen, darunter modale Prädikatenlogik, intuitionistische Temporallogik, usw. Als ein Beispiel wird ein Kalkül vorgestellt ist, der durch die Kombination eines Kalkül für Prädikatenlogik erster Stufe und eines Kalküls für eine Modallogik entsteht.

Kapitel 7 Das Konzept des Theorieschließens, das es erlaubt, Tableaukalküle mit dedizierten Prozeduren zur Lösung von Problemen einer bestimmten Domäne zu kombinieren, wird verallgemeinert und mit Hilfe der in den vorangegangenen Kapiteln eingeführten Begriffe neu formuliert.

1.3 Tableaus und warum wir sie benutzen

Ein Tableau, so steht es in Wörterbüchern, ist „ein wohlgeordnetes Bild“ (Drosdowski *et al.*, 1991) oder eine „beeindruckende oder lebhafte Darstellung“ (“a striking or vivid representation”) (Hayward & Sparkes, 1968). Im Automatischen Beweisen ist ein Tableau eine besondere Form der Darstellung eines (partiellen) Beweises. Diese Darstellung ist sicherlich wohlgeordnet; ob sie auch beeindruckend oder gar lebhaft ist, ist natürlich Ansichtssache; aber von all den Beweisrepräsentationen, die sich zur Implementierung eignen, und die darum im Automatischen Beweisen Verwendung finden, ist die Repräsentation mit Hilfe von Tableaus wohl diejenige, die am einfachsten für Menschen zu verstehen ist.

Während Tableaukalküle immer für pädagogische Zwecke in Einführungen in die Logik benutzt wurden, begann das Interesse an ihnen zum Einsatz in der Deduktion erst in den achtziger Jahren zu wachsen. Ein Grund war der wachsende Bedarf an Deduktion in nicht-klassischen Logiken in verschiedenen KI-Anwendungen. Für viele nicht-klassische Logiken sind tableauartige Kalküle die einzigen, die es gibt. Außerdem macht es die Nähe der Tableau-Ableitungsregeln zur Semantik einfach, Tableaukalküle für neue Logiken zu konstruieren. Auch führte die Einführung von Unifikation und anderer Verfeinerungen zu einer Zunahme der Effizienz von Tableaukalkülen für klassische Logik; heute basieren einige der leistungsstärksten automatischen Theorembeweiser für klassische Prädikatenlogik erster Stufe auf Tableaukalkülen.

Es gibt viele verschiedene Definitionen des Begriffs *Tableau* in der Literatur; und es scheint genauso viele Charakterisierungen von Tableaukalkülen und Kriterien, die sie von anderen Kalkülarten unterscheiden, zu geben, wie es Experten auf dem Gebiet der tableaubasierten Deduktion gibt; die am häufigsten verwendeten und wichtigsten dieser Kriterien sind:

1. Ein Tableau ist eine Baumstruktur deren Knoten mit Formeln markiert sind; und jeder Kalkül, der auf solchen Strukturen operiert, ist ein Tableaukalkül.

2. Ein Kalkül ist ein Tableaukalkül, wenn er die (meisten der) folgenden Eigenschaften hat: er führt Beweise, indem er das zu beweisende Theorem analysiert (*top-down*); er beweist durch vollständige Fallunterscheidung; er führt Widerspruchsbeweise.

In dieser Arbeit wird davon ausgegangen, daß ein Kalkül ein Tableaukalkül ist, wenn er beide dieser Kriterien erfüllt. Insbesondere ist die Baumstruktur der Tableaus Bestandteil ihrer Definition (während einige Autoren, z. B. (Fitting, 1998), Bäume nur als eine mögliche Datenstruktur zur Implementierung von Tableaus ansehen). Darum werden im folgenden Kalküle wie beispielsweise die Konnektionsmethode, die auf einer Matrixrepräsentation beruht, und der Sequenzenkalkül *nicht* als Tableaukalküle angesehen – wenn es auch sicherlich eine enge Beziehung zwischen diesen Kalkülen und Tableaukalkülen gibt.[1]

Um Verwechslungen zu vermeiden, müssen die folgenden Begriffe klar unterschieden werden:

- Tableau – ein Baum, dessen Knoten mit Formeln annotiert sind;

- (partielle) Tableaubeweise – welche aus einem Tableau bestehen und der Information, wie dieses Tableau mit Hilfe der Kalkülregeln zu konstruieren ist; diese Information kann z. B. in Form einer Sequenz von Tableaus gegeben sein, wobei jedes Tableau in dieser Sequenz aus dem vorangegangenen durch eine einzelne Regelanwendung erzeugt werden kann;

[1] Jeder Sequenzenkalkül kann in einen Tableaukalkül verwandelt werden – und umgekehrt, wobei die Äste eines Tableaus in dem Tableaukalkül den Sequenzen im Sequenzenkalkül entsprechen.

- der Zustand, den die Berechnung einer Tableau-Beweisprozedur erreicht hat – die, zusätzliche zu einem partiellen Tableaubeweis, Informationen über fehlgeschlagene Beweisversuche enthalten kann;

- Tableaukalküle – die durch ihre Kalkülregeln charakterisiert sind;

- die Tableaumethode im allgemeinen.

In der englischsprachigen Literatur wird der Begriff „Tableau“ in inkonsistenter Weise mit jeder der oben genannten Bedeutungen verwendet. In dieser Arbeit steht „Tableau“ jedoch immer für eine Baumstruktur.

1.4 Historischer Überblick

Historisch gehen Tableaukalküle auf Gentzens beweistheoretische Arbeiten der dreißiger Jahre zurück, nämlich seine *Sequenzenkalküle* (Gentzen, 1935). Die Bezeichnung *Tableau* stammt von Beth (1955), der nach einer „systematischen Methode zur Konstruktion eines Gegenbeispiels“ suchte. Zu etwa der gleichen Zeit – und wie Beth durch semantische Überlegungen motiviert – entwickelten Hintikka (1955) und Schütte (1956) unabhängig voneinander ähnliche Systeme. Diese Kalküle hatten noch Nachteile bezüglich der verwendeten Notation, die umständlich war; aber Hintikka benutzte ein Argument in seinem Vollständigkeitsbeweis, daß (mit wenigen Modifikationen) noch heute Verwendung findet (siehe Abschnitt 3.5.4).

Die moderne Form von Tableaukalkülen (und besonders die Baumdarstellung) wurde, wieder unabhängig, von Lis (1960) und Smullyan eingeführt; letzterer fügte die *uniforme Notation* hinzu (*uniform notation*) und stellte seine Ergebnisse in seinem bekannten Lehrbuch (Smullyan, 1968) vor.

Die Entwicklung von tableauartigen Kalkülen für nicht-klassische Logiken begann parallel zu der von Kalkülen für klassische Logiken – der erste war ein von Gentzen (1935) beschriebener Sequenzenkalkül für intuitionistische Logik. Beth (1959) stellte einen Kalkül für intuitionistische Logik vor, der seinem Kalkül für klassische Logiken ähnelte. Etwa zur gleichen Zeit entwickelten Kanger (1957) sowie Matsumoto und Ohnishi (1957; 1959) tableauartige Kalküle für Modallogiken. In Kangers Kalkül für die Modallogik S5 waren Formeln mit ganzen Zahlen indiziert; dieser kann als der erste tableauartige Kalkül angesehen werden, der mit Markierungen versehene Formeln benutzt. In Kripkes gefeiertem Papier (Kripke, 1959) – in dem er die Mögliche-Welten-Semantik für Modallogiken vorschlug – stellte er einen tableauartigen Kalkül für Modallogiken im Stile Beths (1955) vor. Kripke benutzte Hilfs-Tableaus, wobei verschiedene Tableaus für verschiedene mögliche Welten verwendet und durch eine Erreichbarkeitsrelation zueinander in Beziehung gestellt werden.

Die ersten wirklichen Tableaukalküle (die die Baumdarstellung verwenden) für nichtklassische Logiken, nämlich für die Modallogik S4 und intuitionistische Logik, wurden von Fitting (1969) vorgestellt (später definierte Fitting (1983) Tableaukalküle für viele andere Modallogiken). Ein erster Tableaukalkül für Temporallogiken wurde in (Rescher & Urquhart, 1971) beschrieben; und der erste tableauartige Kalkül für mehrwertige Logiken war Rousseaus Sequenzenkalkül (Rousseau, 1967) (der darauf beruhte, Sequenzen mit mehr als zwei Teilen zu verwenden). Später definierte Suchon (1974) einen Tableaukalkül für Łukasiewicz-Logiken; und Surma (1984), Carnielli (1987) und Hähnle (1990) präsentierten Tableaukalkül für beliebige mehrwertige Logiken.

In den späten fünfziger Jahren begann der Entwurf von tableauartigen Kalkülen und Beweisprozeduren, die speziell auf die automatische Beweissuche und damit die Implementierung automatischer Theorembeweiser zugeschnitten sind; und dieses Forschungsgebiet fing an, sich von der Entwicklung von Tableaukalkülen für rein theoretische oder pädagogische Zwecke abzuspalten.

Die Möglichkeit, die Beweissuche zu automatisieren, wurde zuerst von Kanger (1957; 1963) betrachtet. In den Jahren 1957 und ’59 implementierten D. Prawitz, H. Prawitz und Voghera einen Sequenzenkalkül für Prädikatenlogik erster Stufe ohne Funktionssymbole (Prawitz, 1960; Prawitz *et al.*, 1960). Zur gleichen Zeit, im Jahr 1958, implementierte Wang einen Sequenzenkalkül für das $\forall\exists$-Fragment der Prädikatenlogik erster Stufe auf einer IBM 704 (Wang, 1960); dieses bemerkenswerte Programm war in der Lage, alle 220 aussagenlogischen und 139 der 158 prädikatenlogischen Theoreme in Russell und Whiteheads *Principia Mathematica* zu beweisen.

In den sechziger und siebziger Jahren dominierten resolutionsartige Kalküle das Automatische Beweisen; während dieser Zeit basierten nahezu alle Implementierungen auf Resolution; eine bemerkenswerte Ausnahme ist Popplestones Implementierung eines Tableaukalküls für die Prädikatenlogik erster Stufe im Stile Beths (Popplestone, 1967). Der Hauptvorteil der Resolution war (zu dieser Zeit), daß sie Unifikation verwendet, um sinnvolle Instantiierungen universell quantifizierter Variablen zu finden – während Tableaukalkülen diese mächtige Methode fehlte; sie beruhten darauf, alle möglichen Instantiierungen aufzuzählen. In den achtziger Jahren überwanden Tableaukalküle diesen Nachteil. Die Einführung freier Variablen, deren Instantiierungen mit Hilfe von Unifikation berechnet werden, führte zu einer drastischen Effizienzsteigerung bei tableaubasierten Theorembeweisern.

Die Idee, Unifikation in Tableaukalkülen zu verwenden, wurde zuerst in (Cohen *et al.*, 1974) erwähnt; die ersten Tableaukalküle, die Unifikation verwenden, wurden (unabhängig voneinander) in (Broda, 1980) und (Bowen, 1982) vorgestellt. Jedoch wurde in diesen Arbeiten das Problem noch nicht gelöst, die Korrektheit des Kalküls zu bewahren, wenn freie Variablen mit (Skolem-)Konstanten instantiiert werden (siehe Abschnitt 4.4). In der Modell-Elimination (Andrews, 1981) und der Matrix-Methode (Bibel, 1982), die mit Unifikation verwendenden Tableaukalkülen eng verwendet sind, wurde das Problem dadurch umgangen, daß man von einer Eingabeformel in skolemisierter Negations-Normalform ausging, so daß keine Skolem-Konstanten während eines Beweises eingeführt werden.

Die ersten Tableaukalküle für die Prädikatenlogik erster Stufe, die Skolemisierung zur Laufzeit und Unifikation verwende, wurden von Wrightson (1984) und Reeves (1987) vorgestellt; diese lösten das Korrektheits-Problem durch Beschränkung der Unifizierbarkeit von freien Variablen und Skolem-Konstanten. Schließlich wurde die Methode zur Bewahrung der Korrektheit, die heute hauptsächlich Anwendung findet, nämlich komplexe Skolem-Terme statt Skolem-Konstanten zu verwenden, in (Schmitt, 1987) und (Fitting, 1990) eingeführt.

Eine weitere wichtige Verbesserung von Tableaukalkülen für die Prädikatenlogik erster Stufe war die Einführung von *Konnektionsbedingungen* (siehe Abschnitt 5.5); der Begriff der Konnektiertheit wurde zuvor in der Modell-Elimination (Andrews, 1981) und der Matrix-Methode (Bibel, 1982) verwendet.

Heute ist die zugänglichste und kompakteste Quelle von Informationen über Tableaukalküle das *Handbook of Tableau Methods* (D'Agostino *et al.*, 1998). Seine Kapitel decken die wichtigsten Varianten von Tableaukalkülen für klassische wie auch nichtklassische Logiken ab.

2 Logische Systeme

2.1 Syntax und Semantik eines logischen Systems

Wir definieren den Begriff eines *logischen Systems* auf sehr allgemeine Weise; nur sehr grundlegende Eigenschaften seiner Syntax und seiner Semantik sind Teil dieser Definition.

Die Logik muß eine Modell-Semantik haben, die Modelle im Stile Kripkes verwendet, d. h., Modelle, die aus (möglichen) *Welten* bestehen, in denen Formeln entweder wahr oder falsch sind. Das schließt Modelle ein, die aus nur einer Welt bestehen (beispielsweise Modelle der klassischen Aussagenlogik und der klassischen Prädikatenlogik erster Stufe;[1] und es gibt keine Beschränkung dessen, was die Beziehung zwischen diesen Welten ist.

Definition 2.1.1 Ein *logisches System* $\mathbf{L}$ (eine *Logik*) ist charakterisiert durch eine Menge Sig von ($\mathbf{L}$-)Signaturen[2] von $\mathbf{L}$. Für jede Signatur $\Sigma \in Sig$ sind Syntax und Semantik der Instanz $\mathbf{L}(\Sigma)$ von $\mathbf{L}$ gegeben durch:

Syntax: Eine Menge $Form(\Sigma)$ von *(*$\mathbf{L}$*-)Formeln* und eine Menge $Atom(\Sigma)$ von *atomaren* ($\mathbf{L}$-)Formeln ([$\mathbf{L}$-]Atomen), die eine Teilmenge von $Form(\Sigma)$ ist, wobei die Mengen $Atom(\Sigma)$ und $Form(\Sigma)$ entscheidbare formale Sprachen sind (die nicht das leere Wort enthalten), d. h., Atome und Formeln sind Wörter in diesen Sprachen.

Semantik: Eine Menge $\mathcal{M}(\Sigma)$ von *(*$\mathbf{L}$*-)Modellen*, wobei jedes Modell $\mathbf{m} \in \mathcal{M}(\Sigma)$ (zumindest) aus folgendem besteht:

1. einer Menge W von *Welten*,
2. einer *initialen Welt* $w^0 \in W$, und
3. einer binären Relation $\models$ zwischen W und $Form(\Sigma)$.

[1] Tatsächlich kann jedes Modell als ein Kripke-Modell mit nur einer Welt aufgefaßt werden (nämlich dem Modell selbst). Da jedoch die Markierungen der Tableauformeln als Welten interpretiert werden, werden sie nutzlos für einen Kalkül, wenn die Interpretation aller Markierungen dieselbe ist.

[2] Was eine Signatur ist, wird nicht weiter spezifiziert; Sig kann als eine Menge von Indizes angesehen werden, die die Unterscheidung verschiedener Instanzen der Logik $\mathbf{L}$ erlaubt (welche sich *gewöhnlich* in den Symbolen unterscheiden, die sie verwenden).

Wenn $w \models F$ für eine Welt $w \in W$ und eine Formel $F \in \mathit{Form}(\Sigma)$ gilt, dann wird F als *wahr* in w bezeichnet, andernfalls als *falsch* in w.

Eine Formel $F \in \mathit{Form}(\Sigma)$ wird (genau dann) von einem Modell $\mathbf{m} \in \mathcal{M}(\Sigma)$ *erfüllt*, wenn sie in der initialen Welt w^0 von $\mathbf{m}$ wahr ist. Eine Menge $G \subset \mathit{Form}(\Sigma)$ von Formeln wird (genau dann) von $\mathbf{m}$ erfüllt, wenn ihre Elemente von $\mathbf{m}$ erfüllt werden.

Eine Formel $F \in \mathit{Form}(\Sigma)$ (eine Menge $\mathfrak{F} \subset \mathit{Form}(\Sigma)$ von Formeln) ist *erfüllbar*, wenn es ein Modell $\mathbf{m} \in \mathcal{M}$ gibt, das F (bzw. $\mathfrak{F}$) erfüllt. $\square$

Obwohl nicht-atomare Formeln gewöhnlich aus atomaren Formeln aufgebaut werden, und ihr Wahrheitswert durch die Wahrheitswerte der Atome bestimmt ist, aus denen sie bestehen, ist das *nicht* Teil der obigen Definition. Jedoch impliziert die Existenz eines „gutartigen" Tableaukalküls für eine Logik $\mathbf{L}$, daß der Wahrheitswert einer Formel F in engem Bezug zu den Wahrheitswerten gewisser Atome steht (die Teilformeln von F sein können aber nicht müssen).

Beispiel 2.1.2 Der Wahrheitswert der Formel $(\exists x)(p(x))$ in einem Modell der Prädikatenlogik erster Stufe ist nicht durch den Wahrheitswert des (einzigen) Atoms bestimmt, das sie enthält – noch kann er aus den Wahrheitswerten aller atomaren Formeln $p(t)$ berechnet werden (es sei denn, alle Elemente im Universums des Modells sind durch einen Term t repräsentiert, wie es in Herbrand-Modellen der Fall ist). $\square$

Tableaukalküle erlauben, die *Erfüllbarkeit* einer Formel zu überprüfen; wir beschäftigen uns nur mit dieser Eigenschaft. Es mag in einer bestimmten Logik möglich sein oder auch nicht, zu testen, ob eine Formel gültig in einem Modell ist (wahr in allen Welten) oder ob sie allgemeingültig ist (wahr in allen Modellen), indem man dieses Problem auf ein Erfüllbarkeits-Problem reduziert; in vielen Logiken – jedoch nicht in allen – ist eine Formel genau dann allgemeingültig, wenn ihre Negation nicht erfüllbar ist.

Oft werden Formeln in Tableaukalkülen benutzt, die nicht aus Symbolen der ursprünglichen Signatur sondern aus Symbolen einer erweiterten Signatur gebildet werden (beispielsweise Formeln, die Skolem-Symbole enthalten):

Definition 2.1.3 Sei eine Logik gegeben. Eine Signatur $\Sigma^* \in \mathit{Sig}$ ist eine *Erweiterung* einer Signatur $\Sigma \in \mathit{Sig}$ (und Σ ist eine *Einschränkung* von Σ^*), falls

$$\mathit{Form}(\Sigma) \subset \mathit{Form}(\Sigma^*) \text{ and } \mathit{Atom}(\Sigma) \subset \mathit{Atom}(\Sigma^*) \enspace .$$

Dann ist ein Modell $\mathbf{m} \in \mathcal{M}(\Sigma)$ eine *Einschränkung* eines Modells $\mathbf{m}^* \in \mathcal{M}(\Sigma^*)$ (auf die Signatur Σ), falls es eine Funktion f gibt, die jeder Welt in $\mathbf{m}$ eine Welt in $\mathbf{m}'$ zuweist, so daß: (a) der initialen Welt von $\mathbf{m}^*$ die initiale Welt von $\mathbf{m}$ zugewiesen wird; und (b) für alle Formeln $F \in \mathit{Form}(\Sigma)$ und alle Welten w von $\mathbf{m}$: $w \models F$ genau dann, wenn $f(w) \models F$. $\square$

2.2 Terme und Substitutionen

2.2.1 Logische Systeme mit Termen und freien Variablen

Es gibt eine wichtige Klasse logischer Systeme, einschließlich aller Prädikatenlogiken, in der die Formeln *Terme* enthalten. Wieder machen wir so wenige Einschränkungen wie möglich bezüglich dessen, was ein Term sein kann. Insbesondere machen wir nicht die Annahme, daß Terme von der Form $f(t_1, \ldots, t_n)$ sind; statt dessen kann die Menge von Termen jede entscheidbare Menge von Wörtern sein, die in Formeln auftreten. Die einzige Bedingung, die sogar benutzt wird, um den Begriff Term zu definieren, ist, daß die Menge der Formeln abgeschlossen ist unter der Ersetzung von Termen in einer Formel durch andere Terme.

Um flexibler zu sein, lassen wir zu, daß die Terme in verschiedene Klassen eingeteilt sind, d. h., wir versehen die Terme mit *Sorten*. Wir benutzen keine Untersorten-Hierarchie; jedoch können die meisten Begriffe und Methoden, die im folgenden eingeführt werden, leicht an ein komplexeres Sorten-Konzept angepaßt werden (ein Überblick über Tableaukalküle für Prädikatenlogik erster Stufe mit sortenbehafteten Termen findet sich in (Weidenbach, 1995)).

Definition 2.2.1 Eine formale Sprache L ist eine *Sprache mit Termen*, wenn

1. es eine nicht-leere Menge $Term$ von *(Grund-)Termen* gibt, die eine entscheidbare formale Sprache über dem gleichen Alphabet wie L ist (und nicht das leere Wort enthält);

2. es eine nicht-leere Menge S von *Sorten* gibt;

3. es eine Funktion $sort$ gibt, die jedem Term $t \in Term$ eine Sorte $s \in S$ zuweist, so daß es mindestens einen Term jeder Sorte gibt;

4. die Menge von Termen einer Sorte s abgeschlossen ist unter der Ersetzung eines Teilterms einer Sorte s' durch einen Teilterm der gleichen Sorte s', d. h., wenn $vrw \in Term$, $r, r' \in Term$ und $sort(r) = sort(r')$, dann $vr'w \in Term$ und $sort(vrw) = sort(vr'w)$;

5. die Sprache L abgeschlossen ist unter der Ersetzung von Termen durch andere Terme derselben Sorte, d. h., wenn $s, t \in Term$, $sort(s) = sort(t)$ und das Wort wsw' ein Element von L ist, dann ist wtw' ein Element von L.

Wenn t und r Terme sind und t von der Form vrw ist (v, w können leer sein), dann ist r ein *Teilterm* von t. □

Die Abgeschlossenheits-Eigenschaft erlaubt es, Terme durch Platzhalter zu ersetzen, die für beliebige Terme einer bestimmten Sorte stehen. Wir nennen diese Platzhalter *freie Variablen*. Ein Wort, das eine freie Variable X^s der Sorte s enthält, steht für ein einzelnes (unbekanntes) Element der Menge aller Wörter, die dadurch entstehen, daß X^s durch einen Term der Sorte s ersetzt wird.

Ein Nicht-Grund-Term wird konstruiert, indem man eine beliebige Zahl von Teiltermen eines Grundterms durch freie Variablen ersetzt; ein Nicht-Grund-Wort wird konstruiert, indem in einem Grundwort auftretende Grundterme durch Nicht-Grund-Terme ersetzt werden.

Definition 2.2.2 Sei L eine Sprache mit Termen; und sei Var eine unendliche Menge von *freien Variablen*, die nicht in L vorkommen. Die Funktion $sort$ wird (beliebig) auf Var erweitert, so daß es eine unendliche Zahl freier Variablen jeder Sorte $s \in S$ gibt.

Dann ist die Menge $Term^{\mathrm{fv}}$ von *(Nicht-Grund-)Termen* definiert durch:

$$\begin{aligned}
Term_0^{\mathrm{fv}} &= Term \\
Term_i^{\mathrm{fv}} &= \{vXw \mid vtw \in Term_{i-1}^{\mathrm{fv}},\ t \in Term,\ X \in Var,\ sort(t) = sort(X)\} \\
Term^{\mathrm{fv}} &= \bigcup_{i \geq 0} Term_i^{\mathrm{fv}} \ .
\end{aligned}$$

Die Funktion $sort$ wird erweitert auf (Nicht-Grund-)Terme in $Term^{\mathrm{fv}}$ durch die Definition

$$sort(vXw) = sort(vtw)$$

für alle Variablen X und Terme t mit $sort(X) = sort(t)$.

Die Sprache L^{fv} der *(Nicht-Grund-)Wörter* ist definiert durch:

$$L^{\mathrm{fv}} = \{vtw \mid vsw \in L,\ s \in Term,\ t \in Term^{\mathrm{fv}},\ sort(s) = sort(t)\} \ .$$

□

Definition 2.2.3 Ein logisches System ist eine *Logik mit Termen*, wenn für jede Signatur $\Sigma \in Sig$ die Mengen $Form(\Sigma)$ von Formeln und $Atom(\Sigma)$ von Atomen eine Sprache mit Termen mit der gleichen Menge $Term(\Sigma)$ von Termen sind. □

Im folgenden benutzen wir die Mengen $Var = \{X_i \mid i \geq 1\}$ und $UVar = \{\boldsymbol{x}_i \mid i \geq 1\}$ freier Variablen; wir gehen davon aus, daß diese Variablen verschieden sind von allen anderen auftretenden Symbolen (ohne daß dies in Definitionen explizit erwähnt wird).

Freie Variablen, welche entweder mit Großbuchstaben (X, Y, Z, X_i, X' usw.) oder mit Buchstaben in Fettdruck ($\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}, \boldsymbol{x}_i, \boldsymbol{x}'$ usw.) bezeichnet werden, sollten nicht mit Objektvariablen verwechselt werden, die in $Term(\Sigma)$ auftreten und mit x, y, z, x_i, x' usw. bezeichnet werden. Ein Term, der keine freien Variablen enthält, wird immer als *Grund*-term bezeichnet – auch wenn er Objektvariablen enthält, die nicht durch einen Quantor gebunden sind.

2.2.2 Substitutionen

Ein wichtiger Begriff ist der der *Substitution* von Variablen durch Terme. Dieses Konzept, das aus der klassischen Logik bekannt ist, kann leicht auf unseren allgemeineren Termbegriff erweitert werden.

Definition 2.2.4 Sei L eine Sprache mit Termen. Eine *Substitution* ist eine Abbildung

$$\sigma : \mathit{Var} \to \mathit{Term}^{\mathrm{fv}}$$

von freien Variablen auf (Nicht-Grund-)Terme, so daß $\mathit{sort}(X) = \mathit{sort}(\sigma(X))$ für alle $X \in \mathit{Var}$.

Die Menge $\mathit{dom}(\sigma) = \{X \in \mathit{Var} \mid \sigma(X) \neq X\}$ heißt der *Definitionsbereich* von σ; und die Menge $\mathit{ran}(\sigma) = \{\sigma(X) \mid X \in \mathit{dom}(\sigma)\}$ von Termen heißt der *Wertebereich* von σ.

Falls $\mathit{dom}(\sigma) = \{X_1, \ldots, X_n\}$ endlich ist, dann kann $\{X_1 \mapsto t_1, \ldots, X_n \mapsto t_n\}$ als Darstellung von σ verwendet werden, wobei $t_i = \sigma(X_i)$ ($1 \leq i \leq n$).

Eine Substitution $\{X_1 \mapsto Y_1, \ldots, X_n \mapsto Y_n\}$, die die Variablen X_i in paarweise verschiedene Variablen Y_j abbildet, heißt eine *Variablenumbenennung*.

Ein Wort $w' \in L$ ist eine *Variante* eines Wortes $w \in L$, wenn es eine Variablenumbenennung π gibt, so daß $w\pi = w'$.

Eine *Anwendung* $t\sigma$ einer Substitution σ auf einen Term t ist das Ergebnis der (simultanen) Ersetzung aller Vorkommen freier Variablen X in t durch $\sigma(X)$. Der Term $t\sigma$ heißt eine *Instanz* von t. Die Anwendung einer Substitution auf eine Formel und Instanzen von Formeln sind in analoger Weise definiert.

Die *Einschränkung* $\sigma_{|V}$ einer Substitution σ auf eine Menge $V \subset \mathit{Var}$ von Variablen ist die Substitution, die für alle $X \in \mathit{Var}$ definiert ist durch:

$$\sigma_{|V}(X) = \begin{cases} \sigma(X) & \text{falls } X \in V \\ X & \text{sonst} \end{cases}$$

Die *Komposition* $\sigma \circ \tau$ zweier Substitutionen σ und τ ist die Substitution, die für alle $X \in \mathit{Var}$ definiert ist durch:

$$(\sigma \circ \tau)(X) = \sigma(\tau(X)) \ .$$

Die leere Substitution, die einen leeren Definitionsbereich hat, wird mit id bezeichnet.

Eine Substitution σ ist *idempotent*, wenn $\sigma = \sigma \circ \sigma$. Die Menge aller idempotenten Substitutionen wird mit Subst bezeichnet.

Eine Substitution τ *grundiert* eine Formel F (eine Menge $\mathfrak{F}$ von Formeln), wenn $\mathit{dom}(\tau)$ die Menge der freien Variablen ist, die in F (bzw. in $\mathfrak{F}$) vorkommen, und $F\tau$ (bzw. $\mathfrak{F}\tau$) keine freien Variablen enthält. $\square$

Das Ergebnis der Anwendung einer Komposition $\sigma \circ \tau$ auf einen Term t kann berechnet werden, indem zuerst τ und dann σ angewendet wird, d. h., $t(\sigma \circ \tau) = (t\tau)\sigma$.

Wenn eine *idempotente* Substitution $\sigma = \{X_1 \mapsto t_1, \ldots, X_n \mapsto t_n\}$ auf einen Term angewendet wird, ist es nicht notwendig, die Variablen simultan zu ersetzen, d. h.,

$$\sigma = \{X_1 \mapsto t_1\} \circ \cdots \circ \{X_n \mapsto t_n\}$$

Beispiel 2.2.5 Die Substitutionen $\{X \mapsto Y,\, Z \mapsto Y\}$ und $\{X \mapsto a,\, Y \mapsto f(b)\}$ sind idempotent.

Die Substitutionen $\{X \mapsto Y,\, Y \mapsto a\}$ und $\{X \mapsto f(X)\}$ sind *nicht* idempotent. □

Definition 2.2.6 Sei W eine endliche Menge freier Variablen; und seien σ und τ Substitutionen in $Subst(\Sigma)$. Die Substitution σ ist *allgemeiner* als τ (auf W), und τ ist eine Spezialisierung von σ, in Zeichen $\sigma \leq^W \tau$, falls es eine Substitution $\rho \in Subst(\Sigma)$ gibt, so daß $\tau(X) = (\sigma(X))\rho$ für alle $X \in W$. □

Die Menge W enthält die „relevanten" freien Variablen, d. h., diejenigen freien Variablen, die in dem Kontext vorkommen, in dem die Substitutionen σ und τ verwendet werden (gewöhnlich die in einem bestimmten Tableau vorkommenden). Es ist von Vorteil, die Menge W so klein wie möglich zu halten, weil beispielsweise die Substitution $\sigma = \{X \mapsto f(Y)\}$ die Substitution $\tau = \{X \mapsto f(c)\}$ subsumiert, falls $Y \notin W$; im Falle $Y \in W$ subsumiert σ die Substitution $\tau' = \{Y \mapsto f(c), Y \mapsto c\}$ aber nicht τ.

Die leere Substitution id ist die allgemeinste aller Substitutionen, d. h., es gilt $id \leq^W \sigma$ für alle Substitutionen σ und alle Mengen W freier Variablen.

2.2.3 Unifikation

Obwohl Terme auf eine allgemeinere Weise als üblich definiert sind, ist es möglich, den Begriff der *Unifikation* einzuführen. Allerdings ist das Problem zu überprüfen, ob zwei Terme unifizierbar sind, nicht immer entscheidbar.

Definition 2.2.7 Sei K eine Sprache mit Termen. Zwei Terme $r, t \in Term^{\mathrm{fv}}$ sind *unifizierbar*, wenn $r\mu = t\mu$ für eine Substitution $\mu \in Subst$. In diesem Fall heißt μ ein *Unifikator* von r und t. □

In vielen Logiken mit Termen (z. B. Prädikatenlogik erster Stufe) ist es möglich, die Menge aller Lösungen eines Unifikationsproblems (alle Unifikatoren) durch einen einzigen allgemeinsten Unifikator (*most general unifier*, MGU) zu repräsentieren, der allgemeiner bezüglich der Subsumtionsrelation $\leq^W$ (Def. 2.2.6) ist als alle anderen Unifikatoren. Im allgemeinen Fall ist jedoch ein einzelner MGU nicht ausreichend, um alle Lösungen zu repräsentieren. Statt dessen muß eine *Menge* $\mathcal{U}$ (allgemeinster) Unifikatoren benutzt werden; $\mathcal{U}$ ist *vollständig*, wenn jede Lösung eines gegebenen Problems von einem Unifikator in $\mathcal{U}$ subsumiert wird.

Beispiel 2.2.8 Seien $s = abc$ und $t = XY$ Terme. Die Substitutionen

$$\sigma_1 = \{X \mapsto ab, Y \mapsto c\} \text{ und } \sigma_2 = \{X \mapsto a, Y \mapsto bc\}$$

bilden eine vollständige Menge von Unifikatoren von s und t; aber da sie bezüglich $\leq^W$ unvergleichbar sind, gibt es keine einzelne Substitution die alle Unifikatoren von s und t repräsentiert. □

In Tableaukalkülen mit freien Variablen ist die Kardinalität einer vollständigen Menge von Unifikatoren eng verknüpft mit der Anzahl von Wahlmöglichkeiten (*choice points*) bei der Anwendung von Tableauregeln, die Unifikation involvieren. Darum ist es wünschenswert, eine *minimale* vollständige Menge von Unifikatoren zu berechnen. Allerdings ist es nicht immer sinnvoll, die Minimalität zu erzwingen, da zwar einerseits eine minimale Menge von Unifikatoren von Vorteil ist, andererseits aber zusätzlicher Aufwand durch das Entfernen subsumierter Substitutionen entsteht.

Definition 2.2.9 Sei L eine Sprache mit Termen; sei W eine Menge freier Variablen; und seien $r, t \in \mathit{Term}^{\mathrm{fv}}$ Terme. Eine Menge $\mathcal{U} \subset \mathit{Subst}$ ist eine *vollständige Menge von Unifikatoren* von r und t bzgl. W, falls

1. jedes $\sigma \in \mathcal{U}$ ist ein Unifikator von r und t *(Korrektheit)* und
2. für jeden Unifikator τ von r und t gibt es einen Unifikator $\sigma \in \mathcal{U}$, so daß $\sigma \leq^W \tau$ *(Vollständigkeit)*.

Die Menge $\mathcal{U}$ heißt *minimale* vollständige Menge von Unifikatoren, falls zudem

3. es *keine* Unifikatoren $\sigma_1, \sigma_2 \in \mathcal{U}$, $\sigma_1 \neq \sigma_2$, gibt, so daß $\sigma_1 \leq^W \sigma_2$ *(Minimalität)*. □

Wenn zwei Terme unifizierbar sind, dann besitzen sie eine *endliche* vollständige Menge von Unifikatoren.

Die Berechnung von (allgemeinsten) Unifikatoren ist eng verknüpft mit der Berechnung (allgemeinster) gemeinsamer Spezialisierungen (idempotenter) Substitutionen, denn eine Substitution ρ ist genau dann eine gemeinsame Spezialisierung von σ und τ. wenn sie für alle $X \in dom(\sigma) \cap dom(\tau)$ ein Unifikator der Terme $\sigma(X)$ und $\tau(X)$ ist.

2.3 Prädikatenlogik erster Stufe

Als ein erste Beispiel für ein logisches System benutzen wir die Prädikatenlogik erster Stufe PL1 , die ein logisches System *mit Termen* ist. Gemäß Definition 2.1.1 muß die Menge $\mathit{Sig}_{\mathrm{PL1}}$ der Signaturen, sowie Syntax und Semantik von PL1 definiert werden.

Signaturen: Eine Signatur $\Sigma = \langle P(\Sigma), F(\Sigma), \alpha_\Sigma \rangle$ in Sig_{PL1} besteht aus einer Menge $P(\Sigma)$ von Prädikatensymbolen, einer Menge $F(\Sigma)$ von Funktionssymbolen und einer Funktion α_Σ, die jedem Prädikaten- und jedem Funktionssymbol eine Stelligkeit $n \geq 0$ zuweist. Ein Funktionssymbol mit der Stelligkeit 0 heißt eine Konstante.

Syntax: Zusätzlich zu den Prädikaten- und Funktionssymbolen in Signaturen gibt es eine unendliche Menge V von *Objektvariablen* (die disjunkt ist von den Mengen Var und $UVar$ von freien Variablen). Die logischen Operatoren sind $\vee$ (Disjunktion) , $\wedge$ (Konjunktion) , $\rightarrow$ (Implikation) und $\neg$ (Negation) sowie die Quantorensymbole $\forall$ und $\exists$. Wir sehen $\phi \leftrightarrow \psi$ (Äquivalenz) als Abkürzung für $(\phi \wedge \psi) \vee (\neg\phi \wedge \neg\psi)$ an.

Definition 2.3.1 Sei $\Sigma \in Sig_{\mathrm{PL1}}$ eine Signatur. Die Menge $Term^{nc}(\Sigma)$ von *Termen* über Σ ist definiert durch:

1. Alle Variablen $x \in V$ und alle Konstanten $c \in F(\Sigma)$ sind Terme über Σ.
2. Wenn $f \in F(\Sigma)$ ein Funktionssymbol ist und $t_1, \ldots, t_{\alpha_\Sigma(f)}$ Terme über Σ sind, dann ist auch $f(t_1, \ldots, t_{\alpha_\Sigma(f)})$ ein Term über Σ.

Mit $Term^0_{\mathrm{PL1}}(\Sigma)$ bezeichnen wir die Menge aller Terme in $Term^{nc}(\Sigma)$, die keine Objektvariablen enthalten.

Die Menge $Atom^{nc}_{\mathrm{PL1}}(\Sigma)$ von *Atomen* über Σ ist definiert durch: Ist $p \in P(\Sigma)$ und sind $t_1, \ldots, t_{\alpha_\Sigma(f)}$ Terme über Σ, dann ist $p(t_1, \ldots, t_{\alpha_\Sigma(f)})$ ein Atom über Σ.

Die Menge $Form^{nc}_{\mathrm{PL1}}(\Sigma)$ von Formeln über Σ ist definiert durch:

1. Atome über Σ sind Formeln über Σ.
2. Wenn F eine Formel über Σ ist, dann ist $\neg F$ eine Formel über Σ.
3. Wenn F und G Formeln über Σ sind, dann sind $F \wedge G$, $F \vee G$ und $F \rightarrow G$ Formeln über Σ.
4. Wenn F eine Formel über Σ ist und $x \in V$, dann sind $(\forall x)F$ und $(\exists x)F$ Formeln über Σ. $\square$

Wir definieren, daß die Menge $Form_{\mathrm{PL1}}(\Sigma)$ der Formeln des logischen Systems PL1 aus allen *Sätzen* in $Form^{nc}_{\mathrm{PL1}}(\Sigma)$ besteht, d. h., allen Elementen von $Form^{nc}_{\mathrm{PL1}}(\Sigma)$, in denen Objektvariablen nur durch Quantoren gebunden auftreten; und wir definieren, daß die Menge $Atom_{\mathrm{PL1}}(\Sigma)$ der Atome von PL1 die Menge aller atomaren Sätze in $Form^0_{\mathrm{PL1}}(\Sigma)$ ist.

Man beachte, daß dies leicht von der üblichen Terminologie abweicht. Gewöhnlich können Objektvariablen frei in Formeln der Prädikatenlogik vorkommen. Freie Objektvariablen werden häufig als Platzhalter in Tableaukalkülen verwendet. Hier haben wir jedoch die freien Variablen, die Platzhalter sein können, und die Objektvariablen klar voneinander getrennt. Man könnte trotzdem zulassen, daß Objektvariablen frei in Formeln auftreten; das erhöht aber nicht die Ausdrucksstärke und verkompliziert den Entwurf eines Tableaukalküls unnötig. Freie Variablen in einer Formel, deren Unerfüllbarkeit zu beweisen ist, wären so zu behandeln als wären sie existentiell quantifiziert. Zudem müßte man darauf achten, keine freien Variablen neu in den Bindungsbereich eines Quantors einzuführen, wenn Variablen mit Termen instantiiert werden. Um diese Schwierigkeiten zu vermeiden, haben wir formal definiert, daß die Menge der Formeln des logischen Systems PL1 nur aus Sätzen besteht; Formeln mit freien Objektvariablen sind nur Hilfsmittel, die in der rekursiven Definition der Menge aller Sätze benutzt werden. Dementsprechend ist das logische System PL1 eine Logik *mit Termen* bezüglich der Menge $Term^0_{\mathrm{PL1}}(\Sigma)$ aller Terme, die keine Objektvariablen enthalten, denn die Menge $Form_{\mathrm{PL1}}(\Sigma)$ ist abgeschlossen unter der Ersetzung von variablenfreien Termen durch andere variablenfreie Terme.

Semantik: Gemäß Definition 2.1.1 müssen alle Modelle eine Menge von Welten beinhalten. Darum definieren wir, daß $\mathcal{M}_{\mathrm{PL1}}(\Sigma)$ aus Modellen bestehe, bei denen die initiale Welt w^0 eine prädikatenlogische Struktur (Def. 2.3.2) ist und $\{w^0\}$ die Menge der Welten.

Definition 2.3.2 Sei $\Sigma \in Sig_{\mathrm{PL1}}$ eine Signatur.

Eine prädikatenlogische *Struktur* $\langle D, \mathcal{I}\rangle$ für Σ besteht aus einem Universum D und einer Interpretation $\mathcal{I}$, die den Prädikaten- und Funktionssymbolen in Σ eine Bedeutung zuweist.

Eine *Variablenbelegung* sei eine Abbildung $\mu : V \to D$ der Menge der Objektvariablen in das Universum D.

Die *Auswertungsfunktion* val sei wie üblich definiert; das heißt, gegeben eine Struktur $\langle D, \mathcal{I}\rangle$ und eine Variablenbelegung μ, weise sie jeder Formel $F \in Form^{nc}_{\mathrm{PL1}}(\Sigma)$ einen Wahrheitswert $val_{\mathcal{I},\mu}(F) \in \{true, false\}$ zu. □

Die Relation $\models_{\mathrm{PL1}}$ sei für alle Variablenbelegungen μ definiert durch:

$$w^0 \models_{\mathrm{PL1}} F \text{ genau dann, wenn } val_{\mathcal{I},\mu}(F) = true \quad .$$

2.4 Modallogiken

Als ein zweites Beispiel für logische Systeme benutzen wir die bekannten Modallogiken K, KT, KB, K4, K5, K45, KD, KDB, KD4, KD5, KD45, KB4, B, S4 und S5 (einen Überblick gibt beispielsweise (Goré, 1998)).

Signaturen: Die Menge $Sig_{\mathbf{L}}$ ist dieselbe für alle Modallogiken **L**; eine Signatur Σ ist eine aufzählbare, nicht-leere Menge aussagenlogischer Variablen.

Syntax: Formeln werden mit Hilfe der klassischen Operatoren $\wedge$ (Konjunktion), $\vee$ (Disjunktion), $\neg$ (Negation) und den nicht-klassischen, modalen Operatoren $\Box$ („immer") und $\Diamond$ („manchmal") gebildet.

Die Menge $Form_{\mathbf{L}}(\Sigma) = Form_{\mathrm{mod}}(\Sigma)$ der Formeln für eine gegebene Signatur Σ ist dieselbe für alle Modallogiken **L**; Formeln werden wie üblich aus den aussagenlogischen Variablen und den logischen Operatoren aufgebaut. Die Menge

$$Atom_{\mathbf{L}}(\Sigma) = Atom_{\mathrm{mod}}(\Sigma) = \Sigma$$

besteht aus den aussagenlogischen Variablen. Die Modallogiken sind logische Systeme ohne Terme.

Semantik: Wir benutzen eine Mögliche-Welten-Semantik im Stile Kripkes, d. h., die Modelle der Modallogiken sind Kripke-Rahmen.

Definition 2.4.1 Eine *Kripke-Rahmen* sei ein Paar $\langle W, R\rangle$, wobei W eine nicht-leere Menge (möglicher Welten) und R eine binäre Relation auf W ist.

Ein *Kripke-Modell* ist ein Tripel $\langle W, R, V\rangle$, wobei die Belegung V eine Abbildung der aussagenlogischen Variablen in die Menge der Welten ist, d. h., $V(p)$ ist die Menge aller Welten, in denen p unter der Belegung V *wahr* ist.

Falls wRw' (d. h., falls $\langle w, w'\rangle \in R$), dann ist die Welt w' *erreichbar* von der Welt w, und w' ist eine *Nachfolgewelt* von w.

Der Begriff der Wahrheit einer Aussage in einer Welt wird auf komplexe Formeln $F \in Form_{\mathbf{L}}(\Sigma)$ wie folgt erweitert: F sei genau dann *wahr* in einer Welt w, wenn

- G nicht wahr ist in w, im Falle $F = \neg G$,
- G_1 und G_2 wahr sind in w, im Falle $F = G_1 \wedge G_2$,
- G_1 oder G_2 wahr ist in w, im Falle $F = G_1 \vee G_2$,
- G wahr ist in allen Welten, die von w erreichbar sind, im Falle $F = \Box G$,
- G wahr ist in einer Welt, die von w erreichbar ist, im Falle $F = \Diamond G$.

$\Box$

Die ersten zwei Spalten in Tabelle 2.2 zeigen die Axiomatisierungen der 15 einfachen Modallogiken, die mit Hilfe der Axiome in Tabelle 2.1 definiert werden können.

Name	Axiom	Eigenschaft
(K)	$\Box(A \to B) \to (\Box A \to \Box B)$	—
(T)	$\Box A \to A$	reflexiv
(D)	$\Box A \to \Diamond A$	seriell
(4)	$\Box A \to \Box\Box A$	transitiv
(5)	$\Diamond A \to \Box\Diamond A$	euklidisch
(B)	$A \to \Box\Diamond A$	symmetrisch

Tabelle 2.1: Einfache Modallogiken und die entsprechenden Einschränkungen der Erreichbarkeitsrelation.

Logik	Axiome	Logik	Axiome
K	(K)	KT	(K), (T)
KB	(K), (B)	K4	(K), (4)
K5	(K), (5)	K45	(K), (4), (5)
KD	(K), (D)	KDB	(K), (D), (B)
KD4	(K), (D), (4)	KD5	(K), (D), (5)
KD45	(K), (D), (4), (5)	KB4	(K), (B), (4)
B	(K), (T), (B)	S4	(K), (T), (4)
S5	(K), (T), (5)		

Tabelle 2.2: Axiomatische Charakterisierung der einfachen Modallogiken.

Definition 2.4.2 Sei **L** eine der Modallogiken aus Tabelle 2.2. Ein Kripke-Rahmen $\langle W, R\rangle$ heiße ein **L**-*Rahmen*, wenn jede Instanz jedes Axioms von **L** in allen Welten von $\langle W, R\rangle$ wahr ist.

Ein Kripke-Modell $\langle W, R, V\rangle$ heiße ein **L**-*Modell*, wenn $\langle W, R\rangle$ ein **L**-Rahmen ist. $\Box$

Es ist bekannt, daß die in Tabelle 2.1 aufgeführten Axiome die daneben stehenden Eigenschaften von R charakterisieren. Folglich haben alle KT-Rahmen eine reflexive Erreichbarkeitsrelation R, und wenn ein Rahmen eine reflexive Erreichbarkeitsrelation hat, dann erfüllt er das Axiom (T). Darum assoziieren wir die Eigenschaften auch mit den Logiken selbst und sagen beispielsweise, daß eine Logik **L** seriell sei, wenn **L**-Rahmen eine serielle Erreichbarkeitsrelation haben. Einige Vorsicht ist hier jedoch geboten: so ist beispielsweise das Axiom (D) kein Axiom von KT, aber dennoch ist jeder KT-Rahmen seriell, denn (D) wird von (T) impliziert.

Wir können nun fortfahren, die Semantik der einfachen Modallogiken formal zu definieren. Sei **L** eine der einfachen Modallogiken, die in Tabelle 2.2 aufgeführt sind, und sei Σ eine Signatur in $Sig_{\mathbf{L}}$. Die Modelle in $\mathcal{M}_{\mathbf{L}}(\Sigma)$ seien die Kripke **L**-Modelle. Eine der Welten $w^0 \in W$ jedes Modells $\mathbf{m} = \langle W, R, V\rangle$ wird zur initialen Welt w^0 von $\mathbf{m}$ bestimmt. Die Relation $\models_{\mathbf{L}}$ ist für alle Welten $w \in W$ und Formeln $F \in Form_{\mathbf{L}}(\Sigma)$ definiert durch: $w \models_{\mathbf{L}} F$ genau dann, wenn F wahr ist in w.

2.5 Modallogiken ohne binäre Konnektive

Als ein weiteres Beispiel für logische Systeme führen wir die Modallogiken $\widehat{\mathbf{L}}$ ohne binäre Konnektive ein. Alle Formeln sind also modale Literale, d. h., sie sind von der Form $\circ_1 \cdots \circ_n\, p$ ($n \geq 0$), wobei p eine aussagenlogische Variable und $\circ_i$ eine der Modalitäten $\Box$, $\Diamond$ oder das Negationssymbol $\neg$ ist; die Semantik von $\widehat{\mathbf{L}}$ ist die gleiche wie die der entsprechenden vollen Modallogik $\mathbf{L}$ – formaler:

Signaturen: Die Menge $Sig_{\widehat{\mathbf{L}}}$ der Signaturen von $\widehat{\mathbf{L}}$ ist die gleiche wie die der Modallogik $\mathbf{L}$, d. h., ein Signatur ist eine aufzählbare, nicht-leere Menge aussagenlogischer Variablen.

Syntax: Sei Σ eine Signatur in $Sig_{\widehat{\mathbf{L}}}$. Dann sei $Form_{\widehat{\mathbf{L}}}(\Sigma)$ die Menge derjenigen Formeln von $Form_{\mathbf{L}}$, die aus einem einzelnen Element von Σ mit einer vorangestellten Folge der logischen Operatoren $\Box$, $\Diamond$ und $\neg$ besteht. Die Menge $Atom_{\widehat{\mathbf{L}}}(\Sigma)$ ist identisch mit Σ.

Semantik: Die Menge $\mathcal{M}_{\widehat{\mathbf{L}}}(\Sigma)$ der Modelle von $\widehat{\mathbf{L}}$ ist identisch mit der Menge $\mathcal{M}_{\mathbf{L}}$ der Modelle von $\mathbf{L}$; und die Relation $\models_{\widehat{\mathbf{L}}}$ ist die Einschränkung von $\models_{\mathbf{L}}$ auf die Formeln in $Form_{\mathbf{L}}(\Sigma)$, die auch Formeln in $Form_{\widehat{\mathbf{L}}}(\Sigma)$ sind.

Jede Formel in $Form_{\widehat{\mathbf{L}}}(\Sigma)$ ist erfüllbar. Nichtsdestotrotz ist die Logik nicht trivial, denn wir sind an der Erfüllbarkeit von *Mengen* von Formeln interessiert, die implizit konjunktiv verknüpft sind und unerfüllbar sein können.

Die Logiken $\widehat{\mathbf{L}}$ werden in Kapitel 6 benutzt, um die Vorteile des Faserns zu demonstrieren, d. h., der Kombination von Logiken und ihrer Kalküle. Die fehlenden Konnektive können durch Fasern von $\widehat{\mathbf{L}}$ mit der Prädikatenlogik erster Stufe PL1 wieder eingeführt werden; und ein Kalkül für die resultierende modale Prädikatenlogik kann konstruiert werden, indem Kalküle für $\widehat{\mathbf{L}}$ und PL1 gefasert werden.

2.6 Die Fragmente MLSS und MLSSF der Mengentheorie

Als weitere Beispiele für logische Systeme stellen wir zwei Fragmente der quantorenfreien Mengentheorie vor; ein neuer und verbesserter Tableaukalkül für diese Logiken ist in Abschnitt 3.8 definiert.

Die Mengentheorie ist die Sprache der Mathematik. Darum spielt sie eine wichtige Rolle in vielen Anwendungsbereichen des Automatischen Beweisens. Zum Beispiel basieren zwei der am meisten verbreiteten Spezifikationssprachen, nämlich die Sprachen Z und B, ausschließlich auf Mengentheorie. In anderen Sprachen sind Mengen zumindest ein wichtiges Konstrukt, das häufig für Spezifikationen verwendet wird, sei es auf der Meta-Ebene oder als Datenstruktur des spezifierten Programms. Mengentheoretische Beweisverpflichtungen treten sowohl dann auf, wenn eine Implementierung als korrekt bezüglich einer Spezifikation nachzuweisen ist, wie auch beim Testen der Konsistenz der Spezifikation oder der Überprüfung von Vor- oder Nachbedingungen.

Mengentheoretisches Schließen, d. h., der Einsatz spezieller Techniken statt der Verwendung der Axiome der Mengentheorie, ist unverzichtbar für das Automatische Beweisen in vielen Domänen. Automatische Deduktionswerkzeuge können beispielsweise in interaktive Sofware-Verifikationssysteme integriert werden und nehmen dann dem Benutzer die interaktive Lösung einfacher mengentheoretischer Beweisaufgaben ab, die nicht seine Intuition erfordern, sondern durch kombinatorische Suche gelöst werden können.

Der mehrstufige Syllogismus (*multi-level syllogistic*, MLS) besteht aus quantorenfreien Formeln, die mit den mengentheoretischen Prädikaten *Element-von*, *Gleichheit*, *Teilmenge-von*, den binären Funktionen *Vereinigung*, *Durchschnitt*, *Mengen-Differenz* und einer die leere Menge darstellenden Konstante aufgebaut sind. In der Erweiterung von MLS um einelementige Mengen (*MLS with singleton*, MLSS) können n-stellige Funktionen $\{\cdot\}_n$ benutzt werden, um Mengen der Mächtigkeit n darzustellen. Das Fragment MLSSF is die Anreicherung von MLSS um freie (uninterpretierte) Funktionssymbole.

Die Ausdrucksstärke von MLSS und MLSSF ist für viele Anwendungen ausreichend. MLSS-Formeln können Variablen enthalten, die implizit universell quantifiziert sind. Die wesentliche Einschränkung ist, daß keine existentielle Quantifizierung möglich ist; darum können Aussagen wie „es gibt eine unendliche Menge“ nicht in MLSS formalisiert werden.

In der Literatur sind Entscheidungs- und Semi-Entscheidungsverfahren für verschiedene Erweiterungen von MLS beschrieben; diese basieren jedoch nicht auf Tableaus, sondern sind hochgradig nicht-deterministische Suchverfahren und sind nicht für eine Implementierung geeignet. Ein Überblick findet sich in (Cantone & Ferro, 1995; Cantone *et al.*, 1989). Zu den Erweiterungen von MLS, deren Entscheidbarkeit bekannt ist, gehören: MLS mit Potenzmengenoperator und einelementigen Mengen (Cantone, 1991; Cantone *et al.*, 1985), mit relationalen Konstrukten (Cantone & Schwartz, 1991), mit einstelligem Vereinigungsoperator, der alle Teilmengen vereinigt (Cantone *et al.*, 1987), und mit einem Auswahloperator (Ferro & Omodeo, 1987).

Signaturen Eine MLSSF Signatur Σ ist eine PL1-Signatur, so daß

1. ihre Menge $P(\Sigma)$ von Prädikatensymbolen aus den binären Symbolen $\sqsubset$ („Element von“) , $\approx$ (Gleichheit) und $\sqsubseteq$ („Teilmenge von“) besteht,

2. ihre Menge $F(\Sigma)$ von Funktionssymbolen aus

(a) den binären Symbolen $\sqcap$ (Durchschnitt) , $\sqcup$ (Vereinigung) , $\setminus$ (Mengen-Differenz) , den Konstruktoren $\{\cdot\}_n$ mit der Stelligkeit $n \geq 1$ und der mengentheoretischen Konstante $\emptyset$ (die leere Menge),

(b) Funktionssymbolen, die keine besondere mengentheortische Interpretation haben (diese heißen *freie Funktionssymbole*),

besteht.

Eine MLSSF-Signatur ist eine MLSS-Signatur, wenn alle ihre freien Funktionssymbole Konstanten sind, d. h., von der Stelligkeit 0 sind.

Syntax Die Formeln von MLSS und MLSSF sind gemäß den Regeln der Prädikatenlogik erster Stufe aufgebaut, wobei die logischen Operatoren $\vee$ (Disjunktion), $\wedge$ (Konjunktion), $\neg$ (Negation) und $\rightarrow$ (Implikation) aber *keine* Quantoren verwendet werden.

Definition 2.6.1 Sei Σ eine MLSS-Signatur (bzw. eine MLSSF-Signatur).

Die Menge $Atom_{\text{MLSS}}(\Sigma)$ aller *MLSS-Atome* (bzw. die Menge $Atom_{\text{MLSSF}}(\Sigma)$ aller *MLSSF-Atome*) ist die Menge $Atom_{\text{PL1}}(\Sigma)$ aller PL1-Atome über Σ; und die Menge $Form_{\text{MLSS}}(\Sigma)$ der *MLSS-Formeln* (bzw. die Menge $Form_{\text{MLSSF}}(\Sigma)$ der *MLSSF*-Forlmeln) besteht aus allen PL1-Formeln über Σ, in denen die Quantoren $\forall$ und $\exists$ nicht vorkommen (und also auch keine Objektvariablen). $\square$

Notation 2.6.2 Um Verwechselungen zu vermeiden, verwenden wir die Nicht-Standard-Symbole $\sqsubset\!\!\!\!\in, \approx, \sqsubseteq, \sqcap, \sqcup$ auf der Objektebene und die Symbole $\in, =, \subset, \cap, \cup$ auf der Metaebene.

Wie gewöhnlich werden die binären mengentheoretischen Funktions- und Prädikatensymbole in Infix-Notation und $\{\cdot\}_n$ in Circumfix-Notation geschrieben. $\square$

Definition 2.6.3 Ein Term $t \in Term_{\text{PL1}}(\Sigma)$ über einer MLSSF-Signatur Σ heiße ein *Mengenterm*. Er heiße ein *reiner* Mengenterm, wenn er keine freien Funktionssymbole f mit Stelligkeit $\alpha_\Sigma(f) > 0$ enthält. Ein Mengenterm heiße *funktional*, wenn er von der Form $f(t_1, \ldots, t_n)$ ist, wobei f ein freies Funktionssymbol ist. $\square$

Man beachte, daß funktionale Mengenterme nicht-funktionale Mengenterme (die nicht notwendig reine Mengenterme sind) als Teilterme enthalten können – und umgekehrt.

Semantik Wir verwenden die Semantik der Mengentheorie (und damit ihrer Fragmente MLSS und MLSSF), wie sie durch ZF-Axiomensystem oder, in äquivalenter Weise, durch die von-Neumann-Hierarchie von Mengen gegeben ist (siehe z. B. (Jech, 1978) für eine ausführliche Diskussion der Semantik der Mengentheorie).

Definition 2.6.4 Es sei Ord die Klasse aller Ordinalzahlen. Die *von-Neumann-Hierarchie* $\mathfrak{V}$ ist durch

$$\mathfrak{V} = \bigcup_{\alpha \in Ord} \mathfrak{V}_\alpha$$

definiert, wobei

1. $\mathfrak{V}_0 = \emptyset$,
2. $\mathfrak{V}_\alpha = \bigcup_{\beta<\alpha} \mathfrak{V}_\beta$ für jede Limit-Ordinalzahl α und
3. $\mathfrak{V}_{\alpha+1}$ die Potenzmenge von $\mathfrak{V}_\alpha$ für jede Ordinalzahl α bezeichnet. □

Definition 2.6.5 Sei Σ eine MLSS- oder MLSSF-Signatur. Eine prädikatenlogische Struktur $\mathbf{m} = \langle D, \mathcal{I} \rangle \in \mathcal{M}_{\mathrm{PL1}}(\Sigma)$ heiße eine *Mengenstruktur*, wenn sie die folgenden Eigenschaften hat:

1. das Universum D besteht aus den Mengen der von-Neumann-Hierarchie $\mathfrak{V}$;
2. D ist abgeschlossen unter den Mengenoperationen $\cap$, $\cup$, $\setminus$ und $\{\cdot\}_n$ $(n \geq 1)$, und D enthält die leere Menge;
3. $\mathcal{I}$ interpretiert
   (a) die Konstante $\emptyset$ durch die leere Menge,
   (b) die Prädikatensymbole durch ihre kanonische Interpretation, d. h., $\approx$ durch die Identitätsrelation, $\sqsubset\!\!\!-$ durch $\in$ und $\sqsubseteq$ durch $\subseteq$,
   (c) die mengentheoretischen Funktionssymbole durch ihre kanonische Interpretation, d. h., $\sqcup$ durch $\cup$, $\sqcap$ durch $\cap$, $\setminus$ durch $\setminus$ und $\{\cdot\}_n$ durch $\{\cdot\}_n$ $(n \geq 1)$. □

Da Modelle logischer Systeme eine Menge von Welten beinhalten müssen, definieren wir, daß Modelle von MLSS und MLSSF aus einer einzelnen (initialen) Welt w^0 bestehen, die eine Mengenstruktur ist.

Die Relationen $\models_{\mathrm{MLSS}}$ und $\models_{\mathrm{MLSSF}}$ sind in gleicher Weise definiert wie die Relation $\models_{\mathrm{PL1}}$ der Logik PL1: eine MLSS-Formel oder MLSSF-Formel F ist genau dann wahr in der Welt w^0, die eine Mengenstruktur ist, wenn $val_{\mathcal{I}}(F) = true$ für alle Variablenbelegungen μ.

Man könnte zulassen, daß freie Objektvariablen in MLSS und MLSSF-Formeln auftreten; aber das würde die Ausdrucksstärke nicht erhöhen. Da freie Objektvariavlen in quantorenfreien Formeln implizit universell quantifiziert sind, ist eine Formel $G(x)$ genau dann in MLSS oder MLSSF gültig, wenn eine Skolemisierung $\neg G(c)$ ihrer Negation unerfüllbar ist. Darum können freie Objektvariablen eliminiert werden, und ein Tableaukalkül für Formeln ohne freie Objektvariablen ist ausreichend.

3 Tableaukalküle

3.1 Eine uniforme Sicht

Es ist wichtig, die beiden Phasen zu unterscheiden, in die die Entwicklung einer effizienten tableaubasierten Beweisprozedur aufgeteilt werden kann: dem Entwurf eines Tableaukalküls und der Konstruktion einer auf diesem Kalkül aufbauenden Beweisprozedur. Ein Tableaukalkül ist im wesentlichen durch Ableitungsregeln charakterisiert, die nicht-deterministisch angewendet werden können, um einen Tableaubeweis zu konstruieren; eine Beweisprozedur ist eine Beschreibung, wie die Suche nach einem Beweis unter Verwendung eines bestimmten Kalküls zu organisieren ist.

In der Literatur über Tableaukalküle werden diese beiden Phasen oft vermischt; Verfeinerungen, die weder die Korrektheit noch die Vollständigkeit eines Kalküls beeinflussen, sondern den Zweck haben, die Effizienz zu steigern, werden zu einem Teil der Definition des Kalküls. Das ist schädlich, denn welche günstigen Eigenschaften des Kalküls man dann auch immer beweist, sind tatsächlich nur für den verfeinerten Kalkül bewiesen. Außerdem sind verfeinerte Kalküle oft weniger „gutartig" als die ursprüngliche Version, was es schwieriger macht, die uniformen Methoden zur Konstruktion einer effizienten Beweisprozedur anzuwenden, die in den folgenden Kapiteln beschrieben werden.

Eine Verfeinerung, die typischerweise nicht Teil der Definition eines Kalküls sein sollte, ist die meist nützliche Heuristik, daß nicht-verzweigende Regelanwendungen solchen vorzuziehen sind, die mehrere neue Tableauäste einführen. Wenn diese Heuristik Teil der Definition des Kalküls ist, dann ist es beispielsweise unmöglich, eine ausgefeiltere Technik zur Auswahl der nächsten Regelanwendung einzusetzen, die auf einem Komplexitätsmaß für neu hinzukommende Formeln beruht.

Daraus folgt, daß die Vermeidung von Redundanz durch eine Beschränkung des Suchraums, die den Kalkül deterministischer macht, nicht das Thema dieses Kapitels ist; sie wird in Kapitel 5 behandelt, in dem der Entwurf effizienter Beweisprozeduren diskutiert wird. Nichtsdestotrotz spielt die Effizienz beim Entwurf eines Kalküls eine Rolle; so sollten stark indeterministische Regeln, wie beispielsweise die Schnittregel, vermieden werden (jedoch nur dann, wenn es deterministischere Regeln gibt, die gleichermaßen „gutartig" sind).

So wenige Einschränkungen wie möglich werden bezüglich der Art und Form von Tableaus und Tableaukalkülen gemacht. Aber, wie schon in der Einleitung erwähnt

(Abschnitt 1.3), ist ein Tableau immer ein Baum, dessen Knoten mit Formeln markiert sind. Ohne weitere Einschränkungen weiß man natürlich nichts über das Verhalten eines Tableaukalküls – außer daß Tableaus den Zustand der Beweissuche repräsentieren und daß Tableauregelanwendungen Zustandsübergängen entsprechen. Insbesondere ist nichts über die Art und Weise bekannt, wie Zustände repräsentiert werden, und was die Beziehung zwischen den Tableaus ist.

Um in der Lage zu sein, Aussagen über das Verhalten eines Tableaukalküls zu machen und uniforme Methoden anzuwenden, müssen zusätzliche Annahmen gemacht werden. Die erste dieser Annahmen wird sein, daß Tableauäste verschiedene Fälle eines Beweises repräsentieren und daß sie also implizit disjunktiv verknüpft sind; einen Ast abzuschließen bedeutet, daß der entsprechende Fall erfolgreich abgehandelt worden ist. Da Äste verschiedene Fälle darstellen sollen, führt dies auch dazu, daß die Auswirkungen einer Tableauregelanwendungen lokal auf einen Ast beschränkt sind.

Der nächste Schritt wird sein, die Annahme zu machen, daß die Formeln auf einem Ast implizit konjunktiv verknüpft sind und das Wissen repräsentieren, das über den entsprechenden Fall des Beweises abgeleitet worden ist. Das erfordert, daß die Tableauerweiterungsregel monoton ist und auf *Mengen* von Formeln operiert; die Reihenfolge der Formeln auf einem Ast spielt dabei keine Rolle mehr.

Der letzte Schritt sind semantische Annahmen. Man geht davon aus, daß ein Ast (partiell) ein Modell definiert. Die Tableaukonstruktion entspricht dann der Konstruktion eines Modells der Formeln des initialen Tableaus. Ein Ast ist geschlossen, wenn ein Widerspruch in der partiellen Modelldefinition, die er repräsentiert, gefunden wurde. Ein geschlossenes Tableau beweist die Tatsache, daß die Formeln des initialen Tableaus unerfüllbar sind.

Um so viele verschiedenen Kalküle wie möglich in unsere allgemeine Definition dessen, was ein Tableaukalkül sei, einzuschließen, werden die Formeln in Tableaus mit Vorzeichen (*signs*), die den intendierten Wahrheitswert einer Formel repräsentieren, und Markierungen (*labels*) versehen. Markierungen können viele verschiedene Zwecke haben. Sie erlauben es, Informationen über Formeln und die Beziehungen zwischen Formeln explizit darzustellen, und sind besonders in Kalkülen für nicht-klassische Logiken von Nutzen (z. B. mehrwertigen, modalen und intuitionistischen Logiken); viele tableauartige Kalküle, die markierte Formeln verwenden, sind in (Gabbay, 1996b) beschrieben. Tableaukalküle mit markierten Formeln sind sehr viel mächtiger als Kalküle, die Informationen statt sie durch Markierungen darzustellen, in der Struktur eines Tableauastes kodieren. Wenn Information implizit durch die Struktur eines Tableauastes repräsentiert ist, können alle Veränderungen, die die Struktur von Ästen beeinflussen, Korrektheit bzw. Vollständigkeit des Kalküls zerstören; darum bezeichnen wir solche Kalküle als *nicht gutartig* (die Eigenschaft der Gutartigkeit wird formal in Abschnitt 3.3.7 definiert).

Daß nur die zwei Vorzeichen T und F verwendet werden, bedeutet keine Beschränkung auf zweiwertige Logik; diese Vorzeichen repräsentieren die Tatsache, daß eine Formel wahr bzw. nicht wahr ist in einem Modell. Die Wahrheitswerte einer mehrwertigen Logik kann man in Markierungen kodieren, mit denen man die Formeln versieht.

3.2 Syntax von Tableaukalkülen

Wie schon zuvor gesagt, sehen wir die Baumstruktur von Tableaus als eine essentielle Eigenschaft an; Bäume sind *nicht* nur eine Datenstruktur zur Implementierung von Tableaus. Kalküle, wie beispielsweise Sequenzenkalküle und die Konnektionsmethode, die auf anderen Datenstrukturen operieren, sind nur „tableauartig".

Definition 3.2.1 Sei **L** eine Logik; sei $\Sigma \in Sig$ eine Signatur von **L**; und sei Lab eine Menge von Markierungen.

Eine *Tableauformel* $\mathsf{S}{:}\sigma{:}F$ bestehe aus einem Vorzeichen $\mathsf{S} \in \{\mathsf{T}, \mathsf{F}\}$, einer Markierung $\sigma \in Lab$ und einer Formel $F \in Form(\Sigma)$; sie heiße *atomar*, wenn $F \in Atom(\Sigma)$. Außerdem ist das Symbol $\bot$ eine Tableauformel (die den Astabschluß anzeigt). Die Menge aller Tableauformeln sei mit $TabForm(\Sigma)$ bezeichnet.

Das Komplement $\overline{\phi}$ einer Tableauformel ϕ ist definiert durch: $\overline{\phi} = \mathsf{F}{:}\sigma{:}F$, falls ϕ von der Form $\mathsf{T}{:}\sigma{:}F$ ist, und $\overline{\phi} = \mathsf{T}{:}\sigma{:}F$, falls ϕ von der Form $\mathsf{F}{:}\sigma{:}F$ ist (das Komplement von $\bot$ bleibt undefiniert).

Ein *Tableau* (über der Signatur Σ) ist ein endlich verzweigender Baum, dessen Knoten mit Tableauformeln aus $TabForm(\Sigma)$ markiert sind.

Ein *Ast* eines Tableaus T ist ein maximaler Pfad in T. Die Menge der Formeln auf einem Ast B sei mit $Form(B)$ bezeichnet. □

Im folgenden identifizieren wir oft Knoten in einem Tableau mit den Formeln, mit denen sie markiert sind.

Um den Begriff des Tableaukalküls so allgemein wie möglich zu halten, wird jede Funktion, die einem Tableau eine Menge möglicher Nachfolgetableaus zuweist, als *Tableauregel* angesehen. Eine Tableauregel kann ein Tableau, auf das sie angewendet wird, in beliebiger Weise verändern. Tableau*erweiterungs*regeln sind ein Spezialfall von Tableauregeln; sie werden in Abschnitt 3.3.3 behandelt.

Ein Tableaukalkül $\mathcal{C}$ für eine Logik **L** hat (verschiedene) „Instanzen" $\mathcal{C}(\Sigma)$ für jede Signatur $\Sigma \in Sig$. Wir lassen es zu, daß Formeln einer erweiterten Signatur Σ^* in einem Tableaubeweis verwendet werden. Nur die ursprünglichen Formeln, deren Erfüllbarkeit überprüft werden soll, müssen aus der Sprache der nicht-erweiterten Signatur Σ stammen, die eine Einschränkung von Σ^* ist; sie werden auf das initiale Tableau gesetzt. Es ist unverzichtbar für viele Logiken, zu erlauben, daß eine erweiterte Signatur in einem Beweis verwendet wird, oder, was das gleiche bedeutet, zu verlangen, daß die Formeln deren Erfüllbarkeit getestet wird, aus der Sprache einer eingeschränkten

Signatur stammen; dies gestattet es beispielsweise, Skolem-Symbole einzuführen, die mit Sicherheit im initialen Tableau nicht auftreten.

Definition 3.2.2 Ein Tableaukalkül $\mathcal{C}$ für eine Logik **L** ist für jede Signatur $\Sigma \in Sig$ definiert durch:

- eine Erweiterung $\Sigma^* \in Sig$ der Signatur Σ (Def. 2.1.3);
- eine Menge Lab von Markierungen und eine initiale Markierung $\sigma^0 \in Lab$;
- eine Tableauregel $\mathcal{R}(\Sigma)$, die jedem Tableau T über der Signatur Σ^* eine Menge von Tableaus über Σ^* zuweist, die die (möglichen) *Nachfolgetableaus* von T sind. Die Menge $\mathcal{R}(\Sigma)(T)$ kann unendlich sein, sie muß aber aufzählbar sein.

□

Nun haben wir alles zur Verfügung, was nötig ist, zu definieren, welches die Tableaus für eine Menge $\mathfrak{F}$ von Formeln sind und wann ein Tableau geschlossen ist. Die Konstruktion von Tableaus für $\mathfrak{F}$ ist im allgemeinen ein nicht-deterministischer Prozeß, da ein Tableau jede – sogar eine unendliche – Zahl von möglichen Nachfolgetableaus haben kann.

Definition 3.2.3 Sei $\mathcal{C}$ ein Tableaukalkül für eine Logik **L**; und sei $\Sigma \in Sig$ eine Signatur von **L**. Die Menge der Tableaus für eine endliche Menge $\Gamma \subset TabForm(\Sigma^*)$ von Tableauformeln sei wie folgt induktiv definiert:

1. Ein lineares Tableau, dessen Knoten mit den (allen) Formeln aus Γ markiert ist, ist ein Tableau für Γ (ein *initiales Tableau*).
2. Wenn T ein Tableau für Γ ist und $T' \in \mathcal{R}(\Sigma^*)(T)$ (d. h., wenn T' ein Nachfolgetableau von T ist), dann ist T' ein Tableau für Γ.

Ein Tableau T ist ein Tableau für eine endliche Menge $\mathfrak{F} \subset Form(\Sigma)$ von Formeln, wenn es ein Tableau für die Menge $\{\mathsf{T}{:}\sigma^0{:}F \mid F \in \mathfrak{F}\}$ von Tableauformeln ist. □

Einige nützliche Tableaukalküle beginnen (per Definition) mit einem *leeren* initialen Tableau und erlauben, Äste später mit Formeln aus der Menge $\mathfrak{F}$ zu erweitern, deren Erfüllbarkeit zu überprüfen ist. Dies kann in unserem allgemeinen Rahmen leicht modelliert werden, indem eine spezielle initiale Markierung $\circ$ mit der Bedeutung „ist noch nicht auf dem Ast" eingeführt und die Tableauregel so erweitert wird, daß $\mathsf{S}{:}\sigma^0{:}F$ aus $\mathsf{S}{:}\circ{:}F$ abgeleitet werden kann (wobei σ^0 die ursprüngliche initiale Markierung ist). Diese Ableitung entspricht dann dem Hinzufügen der Formel $\mathsf{S}{:}\sigma^0{:}F$ zu dem Tableau. Eine andere mögliche Vorgehensweise, die auch dann angewendet werden kann, wenn die Menge $\mathfrak{F}$ unendlich ist, ist die Tableauregel so zu erweitern, daß $\mathsf{S}{:}\sigma^0{:}F$ aus der leeren Prämisse abgeleitet werden kann.

Definition 3.2.4 Ein Tableauast B ist *geschlossen*, wenn einer seiner Knoten mit $\perp$ markiert ist.

Ein Tableau ist *geschlossen*, wenn alle seine Äste geschlossen sind. □

Intuitiv ist ein Tableaubeweis für eine Menge $\mathfrak{F}$ von Formeln ein Nachweis der Unerfüllbarkeit von $\mathfrak{F}$ (vorausgesetzt, der Kalkül ist korrekt).

Definition 3.2.5 Sei $\mathcal{C}$ ein Tableaukalkül für eine Logik **L**; und sei $\Sigma \in Sig_{\mathbf{L}}$. Ein *Tableaubeweis* für eine Menge $\mathfrak{F} \subset Form(\Sigma)$ von Formeln ist eine endliche Folge $T_0, \ldots, T_n$ ($n \geq 0$) von Tableaus für $\mathfrak{F}$, so daß

- T_0 ein initiales Tableau für $\mathfrak{F}$ ist,
- T_i ein Nachfolgetableau von T_{i-1} ist ($1 \leq i \leq n$),
- T_n geschlossen ist. □

Lemma 3.2.6 *Seien ein Tableaukalkül $\mathcal{C}$ für eine Logik* **L** *und eine Signatur $\Sigma \in Sig_{\mathbf{L}}$ gegeben. Es gibt genau dann einen Tableaubeweis für eine Menge $\mathfrak{F} \subset Form(\Sigma)$ von Formeln, wenn es ein geschlossenes Tableau für $\mathfrak{F}$ gibt.*

Beweis: Dies folgt sofort aus den Definitionen der Begriffe des Tableaus für eine Menge von Formeln und des Tableaubeweises. □

3.3 Syntaktische Eigenschaften

3.3.1 Nicht-destruktive Tableaukalküle

Ein Tableaukalkül ist nicht-destruktiv, wenn Anwendungen seiner Tableauregel die Formeln, die sich schon auf einem Tableau befinden, nicht verändern oder gar entfernen, sondern nur neue Formeln hinzufügt.

Definition 3.3.1 Ein Tableaukalkül heiße *nicht-destruktiv*, wenn alle möglichen Nachfolgetableaus eines beliebigen Tableaus dieses als initialen Teilbaum enthalten; andernfalls heiße der Kalkül *destruktiv*. □

Beispiel 3.3.2 Ein typisches Beispiel für destruktive Kalküle sind solche, die freie Variablen in Tableauformeln verwenden und erlauben, diese freien Variablen mit Termen zu instantiieren, wenn die Tableauregel angewendet wird. □

3.3.2 Beweiskonfluenz

Ein Tableaukalkül ist *beweiskonfluent*, wenn es keine „Sackgassen“ in der Beweissuche gibt. Eine bestimmte Tableauregelanwendung kann nutzlos für die Konstruktion eines Beweises für eine Formelmenge $\mathfrak{F}$ sein, aber in einem beweiskonfluenten Kalkül kann dies niemals verhindern, daß schließlich doch ein Beweis erzeugt werden kann. Eine deterministische Beweisprozedur für einen beweiskonfluenten Kalkül kann dadurch konstruiert werden, daß man die *Fairneß* der Regelanwendungen sicherstellt (Kapitel 5); Backtracking ist dann niemals notwendig.

Definition 3.3.3 Ein Tableaukalkül $\mathcal{C}$ für eine Logik $\mathbf{L}$ ist *beweiskonfluent*, wenn jede Folge $T_0, \dots, T_k$ ($k \geq 0$) von Tableaus für eine Menge $\mathfrak{F} \subset Form(\Sigma)$ von Formeln über einer Signatur Σ von $\mathbf{L}$, für die ein Tableaubeweis existiert, zu einem Tableaubeweis $T_0, \dots, T_k, \dots, T_n$ ($n \geq k$) für $\mathfrak{F}$ erweitert werden kann. □

3.3.3 Tableaukalküle mit Erweiterungsregel

Die wichtigste syntaktische Eigenschaften, die die „Gutartigkeit“ von Tableaukalkülen sicherstellt, ist, daß Tableauregelanwendungen nur *lokale* Auswirkungen haben. Das heißt, die Tableauregel ist nicht-destruktiv und sie erweitert nur *einen einzelnen* Ast eines Tableaus. Außerdem darf es nur von den Formeln auf einem Ast abhängen, in welcher Weise er erweitert werden kann; keine anderen Vorbedingungen sind zulässig, wie beispielsweise das Vorhandensein von Formeln auf anderen Ästen. Ein Kalkül hat diese Lokalitätseigenschaft, wenn seine Tableauregel die Form einer *Tableauerweiterungsregel* hat.

Definition 3.3.4 Sei $\mathcal{C}$ ein Tableaukalkül für eine Logik $\mathbf{L}$; und sei $\Sigma \in Sig$ eine Signatur von $\mathbf{L}$.

Eine *Extension* ist eine endliche Menge von Tableauformeln über Σ^*.

Eine *Konklusion* einer Tableauregelanwendung ist eine endliche Menge von Extensionen.

Eine *Erweiterungsregel* $\mathcal{E}(\Sigma)$ ist eine Funktion, die jedem (endlichen) Tableauast, dessen Knoten mit Formeln aus $TabForm(\Sigma^*)$ markiert sind, eine Menge $\mathcal{E}(\Sigma)(B)$ (möglicher) Konklusionen zuweist, die unendlich sein kann, aber aufzählbar sein muß. □

Man beachte, daß keine speziellen *Abschlußregeln* benötigt werden. Denn ein Ast ist geschlossen, wenn er mit der speziellen Tableauformel $\perp$ erweitert ist. Darum kann der Astabschluß als ein Spezialfall der Asterweiterung aufgefaßt werden.

Definition 3.3.5 Sei $\mathcal{C}$ ein Tableaukalkül für eine Logik **L**; und sei $\Sigma \in Sig$ eine Signatur.

Eine Erweiterungsregel $\mathcal{E}(\Sigma)$ *charakterisiert* die Tableauregel $\mathcal{R}(\Sigma)$ von $\mathcal{C}$, wenn für alle Tableaus T über Σ^* folgendes gilt: ein Tableau T' ist genau dann ein Nachfolgetableau von T (d. h., $T' \in \mathcal{R}(\Sigma)(T)$), wenn es einen Ast B in T und eine Konklusion C in $\mathcal{E}(\Sigma)(B)$ gibt, so daß das Tableau T' aus T konstruiert werden kann, indem der Ast B mit einem neuen Teilast für jede Extension E in C erweitert wird, wobei die Knoten des Teilastes mit den Elementen von E markiert sind.

Wenn die Erweiterungsregel $\mathcal{E}(\Sigma)$ die Tableauregel $\mathcal{R}(\Sigma)$ des Kalküls $\mathcal{C}$ für alle Signaturen Σ charakterisiert, dann heiße $\mathcal{E}$ *die* Erweiterungsregel von $\mathcal{C}$; und $\mathcal{C}$ heiße ein Kalkül mit Erweiterungsregel $\mathcal{E}$. □

Satz 3.3.6 *Ein Tableaukalkül mit Erweiterungsregel ist nicht-destruktiv.*

Beweis: Das Theorem folgt sofort aus den Definitionen 3.3.1 und 3.3.4. □

3.3.4 Analytische Tableaukalküle

Zwei verschiedene Definitionen dessen, was ein *analytischer* Kalkül sei, finden sich in der Literatur. Eine davon ist, daß ein Kalkül analytisch ist, wenn er die zu beweisende Formel „analysiert“, d. h., wenn er ein Top-down-Beweisverfahren ist, – im Gegensatz zu Bottom-up-Beweisverfahren, die versuchen, die zu beweisende Formel aus den Axiomen abzuleiten. In diesem Sinne sind Tableaukalküle immer analytisch; eine etwas stärkere Variante dieser Eigenschaft ist in Abschnitt 3.5 beschrieben.

Hier jedoch verwenden wir die Bezeichnung *analytisch* in ihrer traditionellen, eingeschränkteren Bedeutung: Ein Kalkül ist analytisch, wenn alle Formeln in einem Nachfolgetableau eines Tableaus T schon in T als (Teil-)Formeln vorkommen.

Definition 3.3.7 Ein Tableaukalkül ist *analytisch*, wenn für jedes Tableau T über einer Signatur Σ^* und alle Nachfolgetableaus T' von T folgendes gilt: Wenn $\mathsf{S}'{:}\sigma'{:}F'$ eine Tableauformel in T' ist, dann gibt es eine Tableauformel $\mathsf{S}{:}\sigma{:}F$ in T, so daß F' eine Teilformel von F ist. □

Analytische Kalküle haben einen kleineren Suchraum als nicht-analytische Kalküle, weil es weniger verschiedene Formeln gibt, die durch Tableauregelanwendungen neu eingeführt werden können.

Es ist möglich, analytische Kalküle für die klassische und die meisten nicht-klassischen Aussagenlogiken zu definieren. Kalküle für Prädikatenlogiken sind gewöhnlich nicht analytisch im engen Sinne, weil sie erlauben, beispielsweise das Atom $p(t)$ für alle Terme t aus der Formel $(\forall x)(p(x))$ abzuleiten. Solche Kalküle sind aber dennoch im weiteren Sinne analytisch, wenn nämlich nicht nur Teilformeln aus T sondern auch Instanzen von Teilformeln aus T in Nachfolgetableaus von T vorkommen dürfen.

3.3.5 Monotone Tableaukalküle

Ein Tableaukalkül mit Erweiterungsregel ist *monoton*, wenn die Menge der möglichen Konklusionen eines Ast B', der eine Erweiterung eines Astes B ist, alle möglichen Konklusionen von B enthält.

Definition 3.3.8 Sei $\mathcal{C}$ ein Tableaukalkül mit Erweiterungsregel $\mathcal{E}$ für eine Logik **L**.

Der Kalkül $\mathcal{C}$ ist *monoton*, wenn $\mathcal{E}(\Sigma)(B_1) \subset \mathcal{E}(\Sigma)(B_2)$ für alle Äste B_1 und B_2 über Σ^*, so daß B_1 ein initialer Teilpfad von B_2 ist. □

Wenn ein Kalkül mit Erweiterungsregel monoton ist, dann sind in ihm Regelanwendungen permutierbar. Das heißt, angenommen B ist ein Ast eines Tableaus über Σ und C_1 und C_2 sind Konklusionen in $\mathcal{E}(\Sigma)(B)$, dann können alle Tableauäste, die aus B dadurch konstruiert werden können, daß zuerst eine Extension E_1 aus C_1 und dann eine Extension E_2 aus C_2 angefügt wird, auch durch Permutation der entsprechenden Regelanwendungen gewonnen werden, das heißt, indem B zuerst mit E_2 erweitert wird und dann mit E_1.

3.3.6 Nicht-strukturelle Tableaukalküle

Ein Tableaukalkül ist *nicht-strukturell*, wenn die Reihenfolge der Formeln auf einem Tableauast B weder eine Bedeutung für die Anwendbarkeit der Erweiterungsregel des Kalküls hat, noch für das Ergebnis der Anwendung.

Definition 3.3.9 Sei $\mathcal{C}$ ein Tableaukalkül mit Erweiterungsregel $\mathcal{E}$ für eine Logik **L**.

Der Kalkül $\mathcal{C}$ ist *nicht-strukturell*, wenn für alle Äste B_1 und B_2 über Σ^*, für die $\mathit{Form}(B_1) = \mathit{Form}(B_2)$ gilt,

$$\mathcal{E}(\Sigma)(B_1) = \mathcal{E}(\Sigma)(B_2) \ .$$

□

3.3.7 Gutartige Tableaukalküle

Wir nennen einen Tableaukalkül *gutartig*, wenn er (a) ein Kalkül mit Erweiterungsregel ist, (b) monoton ist und (c) nicht-strukturell ist. Gutartige Kalküle zeigen ein gewisses – zumindest syntaktisches – Wohlverhalten. Die Eigenschaft der Gutartigkeit wird sich in den nachfolgenden Kapiteln als sehr wichtig erweisen.

Definition 3.3.10 Ein Tableaukalkül mit Erweiterungsregel, der nicht-strukturell und monoton ist, heiße *gutartig*. □

Wenn ein Tableaukalkül gutartig ist, dann kann seine Erweiterungsregel als Funktion auf endlichen Mengen von Tableauformeln dargestellt werden (eine Funktion auf *Prämissen*).

Lemma 3.3.11 *Sei $\mathcal{C}$ ein gutartiger Tableaukalkül mit Erweiterungsregel $\mathcal{E}$ für eine Logik* **L**. *Dann gibt es für alle Signaturen Σ von* **L** *eine (eindeutig bestimmte) Funktion $\tilde{\mathcal{E}}(\Sigma)$, die jeder endlichen Menge $\Pi \subset \mathit{TabForm}(\Sigma^*)$ von Tableauformeln (jeder Prämisse) eine Menge $\tilde{\mathcal{E}}(\Sigma)(\Pi)$ (möglicher) Konklusionen zuweist, so daß*

$$\mathcal{E}(\Sigma)(B) = \tilde{\mathcal{E}}(\Sigma)(\mathit{Form}(B))$$

für alle Tableauäste B über Σ^*.

Lemma 3.3.11 impliziert, daß die Funktion $\tilde{\mathcal{E}}(\Sigma)$ auf Mengen von Tableauformeln (Prämissen) als eine alternative Charakterisierung der Erweiterungsregel $\mathcal{E}$ angesehen werden kann – vorausgesetzt, der Kalkül ist gutartig. Wir identifizieren darum im Falle gutartiger Kalküle die beiden Funktionen, bezeichnen sie beide mit $\mathcal{E}$ und nennen sie „Erweiterungsregel".

Eine weitere wichtige Eigenschaft gutartiger Kalküle (Def. 3.3.8) ist, daß sie alle beweiskonfluent sind (Def. 3.3.3).

Satz 3.3.12 *Wenn ein Tableaukalkül gutartig ist, dann ist er beweiskonfluent.*

Beweis: Sei $\mathcal{C}$ ein gutartiger Kalkül für eine Logik **L**; sei $\mathfrak{F} \subset \mathit{Form}(\Sigma)$ eine Menge von Formeln über einer Signatur Σ von **L**, für die ein Tableaubeweis $T'_0, \dots, T'_m$ ($m \geq 0$) existiert; und sei $T_0, \dots, T_k$ ($k \geq 0$) eine Folge von Tableaus für $\mathfrak{F}$.

Ein Tableaubeweis, der eine Fortsetzung von $T_0, \dots, T_k$ ist, kann wie folgt konstruiert werden: für jeden Ast B_i ($1 \leq i \leq r$) von T_k werden die gleichen m Tableauregelanwendungen, die ausgeführt wurden, um T'_m aus T'_0 zu konstruieren, angewendet, um B_i so zu erweitern, daß T'_m als ein Teiltableau an das Ende von B_i angefügt wird. Das ist möglich, weil die initialen Tableaus T_0 und T'_0 die gleichen Formeln enthalten (wenn auch möglicherweise in anderer Reihenfolge) und weil der Kalkül nicht-strukturell und monoton ist. Der resultierende Tableaubeweis ist von der Länge $1 + k + mr$. $\square$

Erweiterungsregeln gutartiger Kalküle werden häufig mit Hilfe von Regelschemata beschrieben (siehe Abschnitte 3.6 und 3.7 für Beispiele). In diesen Schemata sind die Elemente der minimalen Prämisse, die vorhanden sein müssen, um die Ableitung einer gewissen Konklusion zu gestatten, sowie diese Konklusion durch eine horizontale Linie getrennt, während vertikale Linien in der Konklusion deren verschiedene Extensionen trennen. Die Benutzung von Schemata zur Definition von Erweiterungsregeln paßt in unseren allgemeinen Rahmen, jedoch werden verschiedene Schemata sonst üblicherweise als Definitionen verschiedener Regeln angesehen, während wir sie hier nur als Definition verschiedener Teilfälle der (einzelnen) Erweiterungsregel eines Kalküls ansehen.

Definition 3.3.13 Sei $\mathcal{C}$ ein gutartiger Kalkül für eine Logik $\mathbf{L}$; und sei Σ eine Signatur von $\mathbf{L}$.

Eine Menge $\Pi \subset \mathit{TabForm}(\Sigma)$ ist eine *minimale Prämisse* einer Konklusion C, wenn sie eine Prämisse von C ist, d. h., $C \in \mathcal{E}(\Sigma)(\Pi)$, und es keine echte Teilmenge $\Pi' \subset \Pi$, $\Pi \neq \Pi'$, gibt, so daß $C \in \mathcal{E}(\Sigma)(\Pi')$. □

Intuitiv schließt Gutartigkeit aus, daß sich Kalküle „seltsam“ verhalten. Wie sich in den folgenden Kapiteln erweisen wird, ist Gutartigkeit eine sehr wichtige Eigenschaft von Tableaukalkülen. Sie ist eine Vorbedingung für die Anwendbarkeit vieler der uniformen Methoden, die im folgenden beschrieben werden. Auch schon „leichte“ Nicht-Gutartigkeit sollte darum als schädlich angesehen werden. Unglücklicherweise sind viele der Kalküle, die in der Literatur beschrieben und in der Praxis verwendet werden, „ein wenig“ nicht-gutartig. Solche Kalküle können aber oft mit kleineren Änderungen „repariert“ werden. Ein typisches Beispiel sind Kalküle, die Erweiterungsregeln benutzen, die *neue* Symbole einführen, d. h., Symbole, die nicht schon auf dem Ast vorkommen dürfen oder nicht einmal im ganzen Tableau. Wie das folgende Beispiel zeigt, kann dieser Typ von Erweiterungsregeln häufig durch eine ähnliche Regel ersetzt werden, die die Monotonie *nicht* verletzt.

Beispiel 3.3.14 In Kalkülen für die Prädikatenlogik erster Stufe wird häufig eine Erweiterungsregel verwendet, die es erlaubt, aus einer Formel der Form $(\exists x)(F(x))$ die Formel $F(c)$ abzuleiten, wobei c eine *neue* Konstante ist, die noch nicht im Tableau (oder noch nicht auf dem Ast) vorkommt. Ein Kalkül, der eine solche Regel verwendet ist nicht monoton (und also nicht gutartig), weil die Regel die *Abwesenheit* von Formeln fordert, die c enthalten.

Führt man statt dessen ein spezielles Konstantensymbol c_F ein, das nicht notwendigerweise neu ist, dann ist der Kalkül monoton. Die Korrektheit des Kalküls bleibt erhalten, wenn c_F nicht auf andere Weise in Tableaus eingeführt werden kann als durch Skolemisierung von $(\exists x)(F(x))$ (insbesondere dürfen die Skolem-Konstanten c_F nicht im initialen Tableau auftreten); siehe Abschnitt 4.4. □

Gutartige Kalküle existieren für die meisten Logiken. Es gibt jedoch einige nicht-klassische Logiken, deren inhärente Eigenschaften es schwierig oder gar unmöglich machen, einen gutartigen Tableaukalkül zu definieren; dies sind nicht-monotone Logiken (da ihre Kalkül in der Regel nicht-monoton sind) und Logiken mit Ressourcen-Beschränkungen, wie beispielsweise Relevanzlogik (da bei diesen Kalkülen Regelanwendungen nicht-lokale Auswirkungen haben).

Die Gutartigkeitseigenschaft und besonders Monotonie wird oft fallen gelassen und verletzt, um Redundanzen zu beseitigen, wenn ein Kalkül in eine Beweisprozedur umgesetzt wird (siehe Kapitel 5). Das ist unschädlich, solange es die zusätzlichen Suchraumbeschränkungen sind, die die Gutartigkeit verletzen, und diese klar von der Definition des ursprünglichen Kalküls getrennt bleiben.

3.3.8 Stetigkeit

Ein gutartiger Kalkül ist insbesondere monoton. Darum können alle Konklusionen, die aus zwei (separaten) Prämissen abgeleitet werden können, auch aus deren Vereinigung abgeleitet werden, d. h.,

$$\mathcal{E}(\Pi) \cup \mathcal{E}(\Pi') \subset \mathcal{E}(\Pi \cup \Pi') \ .$$

Gilt außerdem

$$\mathcal{E}(\Pi) \cup \mathcal{E}(\Pi') \supset \mathcal{E}(\Pi \cup \Pi')$$

und also

$$\mathcal{E}(\Pi) \cup \mathcal{E}(\Pi') = \mathcal{E}(\Pi \cup \Pi') \ ,$$

dann nennen wir den Kalkül *stetig* (für Π, Π').

Praktisch kein Kalkül ist für alle Prämissen stetig. Hat eine Konklusion C eine minimale Prämisse Π, die aus mehr als einer Tableauformel besteht, ist also $\Pi = \Pi' \cup \Pi''$ wobei Π' und Π'' beide nicht-leer sind, dann ist der Kalkül nicht stetig für die Teilmengen Π' und Π'' (denn $C \notin \mathcal{E}(\Pi') \cup \mathcal{E}(\Pi'')$, da die Prämisse Π minimal ist). Darum wird der Begriff der Stetigkeit bezüglich einer bestimmten Prämisse definiert.

Definition 3.3.15 Sei $\mathcal{C}$ ein gutartiger Kalkül für eine Logik **L**; sei $\Sigma \in Sig$ eine Signatur; und sei $\Pi \subset TabForm(\Sigma^*)$ eine Prämisse.

Die Tableauregel von $\mathcal{C}$ ist *stetig* bzgl. Π, wenn für alle Prämissen $\Pi' \subset TabForm(\Sigma^*)$ gilt:

$$\mathcal{E}(\Pi \cup \Pi') \subset \mathcal{E}(\Pi) \cup \mathcal{E}(\Pi') \ .$$

□

Beispiel 3.3.16 Die Tableauregel des Kalküls für die Prädikatenlogik erster Stufe, der in Abschnitt 3.6 definiert wird, ist stetig bezüglich aller Prämissen, die keine atomaren Formeln enthalten. Die Regel ist nicht stetig bezüglich Prämissen, die Atome enthalten, weil am Abschluß von Ästen zwei komplementäre Atome beteiligt sind; d. h., die minimale Prämisse der Konklusion $\{\{\bot\}\}$ besteht aus mehr als einer Tableauformel.

Eine Tableauregel, die die „Anwendung“ einer Gleichheit gestattet, also erlaubt, aus der Formel $F(t)$ und der Gleichheit $s \approx t$ die Formel $F(s)$ abzuleiten, ist bezüglich keiner Prämisse stetig. □

Stetigkeit ist eine sehr nützliche Eigenschaft, weil eine Prämisse, bzgl. der eine Tableauregel stetig ist, „gelöscht“ werden kann, sobald alle ihre Konklusionen zu einem Ast B hinzugefügt worden sind, d. h., die Prämisse muß nicht weiter zur Erweiterung von B berücksichtigt werden. Stetigkeit steht in enger Beziehung zur semantischen Eigenschaft der Invertierbarkeit (Def. 3.5.11).

3.4 Semantik von Tableaus

Die Semantik der Tableaus eines Kalküls für eine Logik $\mathbf{L}$ basiert auf der Semantik von $\mathbf{L}$, die durch die Menge $\mathcal{M}(\Sigma)$ von Kripke-Modellen für jede Signatur Σ gegeben ist. Die Markierungen, die Teil der Tableauformeln sind, werden als (mögliche) Welten in Modellen interpretiert, und die Vorzeichen kodieren, daß eine Tableauformel wahr bzw. falsch ist.

Definition 3.4.1 Sei $\mathcal{C}$ ein Tableaukalkül für eine Logik $\mathbf{L}$; und sei $\Sigma \in \mathit{Sig}$ eine Signatur von $\mathbf{L}$. Eine *Tableauinterpretation* für $\mathcal{C}(\Sigma)$ ist ein Paar $\langle \mathbf{m}, I \rangle$, wobei

- $\mathbf{m} \in \mathcal{M}(\Sigma^*)$ ein Modell für die erweiterte Signatur Σ^* ist und
- I eine partielle Funktion ist, die den Markierungen $\sigma \in \mathit{Lab}(\Sigma)$ Welten in $\mathbf{m}$ zuweist, so daß $I(\sigma^0) = w^0$ (d. h., I weist der initialen Markierung σ^0 die initiale Welt w^0 von $\mathbf{m}$ zu).

Eine Tableauformel $\mathsf{S}{:}\sigma{:}F \in \mathit{TabForm}(\Sigma^*)$ wird genau dann von $\langle \mathbf{m}, I \rangle$ *erfüllt*, wenn

1. $I(\sigma)$ definiert und
   (a) $\mathsf{S} = \mathsf{T}$ und F ist wahr in $I(\sigma)$ oder
   (b) $\mathsf{S} = \mathsf{F}$ und F ist falsch in $I(\sigma)$
2. oder $I(\sigma)$ undefiniert ist.

Die Tableauformel $\perp$ wird von *keiner* Tableauinterpretation erfüllt.

Ein Tableauast B wird von $\langle \mathbf{m}, I \rangle$ genau dann *erfüllt*, wenn *alle* Tableauformeln auf B von $\langle \mathbf{m}, I \rangle$ erfüllt werden.

Ein Tableau wird von $\langle \mathbf{m}, I \rangle$ genau dann erfüllt, wenn wenigstens einer seiner Äste von $\langle \mathbf{m}, I \rangle$ erfüllt werden. $\square$

Man beachte, daß eine Tableauformel immer erfüllt ist, wenn die Interpretationsfunktion I für ihre Markierung undefiniert ist.

Es macht in der Regel keinen Sinn, alle möglichen Tableauinterpretationen zu verwenden, um die Semantik von Tableaus zu definieren. Eine angemessene Teilmenge aller Tableauinterpretationen muß gewählt werden; diese Wahl hängt sowohl von der Logik als auch dem jeweiligen Kalkül ab. Einerseits darf die Teilmenge nicht zu groß sein, damit man mit ihr die Korrektheit des Kalküls beweisen kann, indem man nachweist, daß alle Tableauregelanwendungen die Erfüllbarkeit in dieser Teilmenge erhalten. Andererseits muß die Teilmenge groß genug sein, damit man mit ihr die Vollständigkeit des Kalküls beweisen kann, was erfordert, daß jedes überhaupt erfüllbare Tableau von einer Interpretation in der Teilmenge erfüllt wird.

Definition 3.4.2 Sei $\mathcal{C}$ ein Tableaukalkül für eine Logik **L**. Die Semantik von $\mathcal{C}$ ist für alle Signaturen Σ von **L** durch eine Menge $TabInterp(\Sigma^*)$ von Tableauinterpretationen definiert. □

Beispiel 3.4.3 Um die Semantik der Tableaus von Kalkülen für die Prädikatenlogik erster Stufe zu definieren, werden meist nur Tableauinterpretationen verwendet, deren erster Teil ein Herbrand-Modell ist. □

3.5 Semantische Eigenschaften

3.5.1 Der Vorteil semantischer Eigenschaften

Die Berücksichtigung semantischer Eigenschaften ist unverzichtbar, weil selbst starke syntaktische Beschränkungen wie Gutartigkeit immer noch zulassen, daß ein Kalkül sich seltsam oder unerwartet verhält. Formeln könnten zu einem Tableau hinzugefügt werden, die zwar das aus einer Prämisse Π abgeleitete Wissen syntaktisch kodieren, deren Semantik (also Wahrheitswert) aber in keiner Beziehung zu dem der Formeln in Π steht. Ein extremes Beispiel wäre ein Kalkül, der zwei Symbole der Signatur verwendet, um die Formeln in Π in einer Binärdarstellung zu kodieren, und dessen Tableauregel auf dieser Binärdarstellung operiert. Es ist unmöglich, uniforme Methoden auf solche Kalküle anzuwenden – auch wenn sie vollständig und korrekt sein mögen – , weil dafür ein Verständnis der Kodierung notwendig wäre. Um ein konservativeres Verhalten zu erzwingen, könnte man den Kalkülen weitere syntaktische Beschränkungen auferlegen, beispielsweise verlangen, daß sie analytisch seien. Die Eigenschaften von Tableauregeln, die es ermöglichen, Techniken wie das Fasern (Kapitel 6) in uniformer Weise anzuwenden, sind jedoch essentiell von semantischer Natur; die Konklusion einer Regelanwendung muß in gewisser semantischer Beziehung zur Prämisse stehen.

Zwei wichtige semantische Eigenschaften sind bereits Teil der Definition von Tableauinterpretationen (Def. 3.4.1), nämlich, daß die Markierungen in Tableauformeln Welten in Modellen repräsentieren und daß die Vorzeichen für Wahrheitswerte stehen; Markierungen und Vorzeichen enthalten keine andere Information.

Die Definition einer Semantik für Tableaus ist nicht nur wichtig, um Korrektheit und Vollständigkeit eines Kalküls nachweisen zu können; daneben sind die Vorbedingungen vieler Suchraumbeschränkungen und anderer nützlicher Techniken von semantischer Natur. Solche semantischen Vorbedingungen durch uniforme syntaktische Vorbedingungen zu ersetzen ist oft schwierig (oder sogar unmöglich). Nichtsdestotrotz kann ein Tableaukalkül auch dann nützlich und sinnvoll sein, insbesondere korrekt und vollständig, wenn keine Semantik für seine Tableaus zur Verfügung steht.

3.5.2 Korrektheit und Vollständigkeit

Die wichtigsten semantischen Eigenschaften von Tableaukalkülen sind Korrektheit und Vollständigkeit:

Definition 3.5.1 Ein Kalkül $\mathcal{C}$ für eine Logik **L** ist *korrekt*, wenn für alle Signaturen $\Sigma \in Sig$ von **L** und alle endlichen Mengen $\mathfrak{F} \subset Form(\Sigma)$ von Formeln folgendes gilt:

Wenn es einen Tableaubeweis für $\mathfrak{F}$ gibt, dann ist $\mathfrak{F}$ unerfüllbar.

Ein Kalkül $\mathcal{C}$ für eine Logik **L** ist *vollständig*, wenn für alle Signaturen $\Sigma \in Sig$ von **L** und alle endlichen Mengen $\mathfrak{F} \subset Form(\Sigma)$ von Formeln folgendes gilt:

Wenn $\mathfrak{F}$ unerfüllbar ist, dann gibt es einen Tableaubeweis für $\mathfrak{F}$.

□

3.5.3 Korrektheit zusichernde Eigenschaften

Es gibt drei wichtige Korrektheitseigenschaften; zusammen sichern sie die Korrektheit eines Kalküls.

1. Ein initiales Tableau für eine erfüllbare Menge von Formeln ist erfüllbar;
2. Tableauregelanwendungen erhalten die Erfüllbarkeit;
3. ein geschlossenes Tableau ist unerfüllbar.

Zusammen implizieren diese Eigenschaften, daß es keinen Tableaubeweis für eine erfüllbare Formelmenge geben kann. Die dritte der Eigenschaften muß nicht überprüft werden, da sie – gemäß unserer Definition eines geschlossenen Tableaus und der Unerfüllbarkeit der Tableauformel $\perp$ – jeder Kalkül hat.

Lemma 3.5.2 *Ein geschlossenes Tableau ist unerfüllbar.*

Beweis: Wenn ein Tableau T geschlossen ist, dann sind alle seine Äste geschlossen, was bedeutet, daß sie alle die Tableauformel $\perp$ enthalten. Da $\perp$ unerfüllbar ist, ist kein Ast von T von irgendeiner Tableauinterpretation erfüllt. Darum ist auch T unerfüllbar.
□

Definition 3.5.3 Sei $\mathcal{C}$ ein Kalkül für eine Logik **L**. Die folgenden *Korrektheitseigenschaften* von $\mathcal{C}$ seien definiert:

Korrektheitseigenschaft 1 ([schwache] Angemessenheit der Menge der Tableauinterpretationen): Für alle Signaturen Σ von **L** gilt, daß es für jede erfüllbare Formelmenge $\mathfrak{F} \subset Form(\Sigma)$ eine Tableauinterpretation in $TabInterp(\Sigma^*)$, die die initialen Tableaus für $\mathfrak{F}$ erfüllt.

Korrektheitseigenschaft 2 ([schwache] Korrektheit der Erweiterung): Für alle Signaturen Σ von **L** gilt, daß es, wenn es eine Tableauinterpretation in $TabInterp(\Sigma^*)$ gibt, die ein Tableau T erfüllt, und T' ein Nachfolgetableau von T ist, es auch eine Tableauinterpretation in $TabInterp(\Sigma^*)$, die T' erfüllt. □

Ein Tableaukalkül, der die beiden Korrektheitseigenschaften aus Definition 3.5.3 besitzt, kann als korrekt (Def. 3.5.1) nachgewiesen werden (ob er gutartig ist oder nicht).

Satz 3.5.4 *Ein Tableaukalkül, der die Korrektheitseigenschaften aus Definition 3.5.3 (Angemessenheit der Menge der Tableauinterpretationen und Korrektheit der Erweiterung) hat, ist korrekt.*

Beweis: Sei $\mathfrak{F} \subset Form(\Sigma)$ eine endliche erfüllbare Menge von Formeln. Wir beweisen durch Widerspruch, daß es keinen Tableaubeweis für $\mathfrak{F}$ geben kann, was die Korrektheit des Kalküls impliziert.

Angenommen, es gebe einen Tableaubeweis $T_0, \ldots, T_n$ ($n \geq 0$) für $\mathfrak{F}$. Da das Tableau T_n geschlossen ist, ist es unerfüllbar (Lemma 3.5.2).

Da jedoch $\mathfrak{F}$ erfüllbar ist, muß gemäß Eigenschaft 1 (Angemessenheit der Menge der Tableauinterpretationen) eine Tableauinterpretation $\langle \mathbf{m}, I \rangle \in TabInterp(\Sigma^*)$ existieren, die das initiale Tableau T_0 erfüllt. Darum sind – nach Induktion über i und unter Verwendung von Eigenschaft 2 (Korrektheit der Erweiterung) – alle Tableaus T_i ($1 \leq i \leq n$) von einer Tableauinterpretation in $TabForm(\Sigma^*)$ erfüllt, was der Tatsache widerspricht, daß T_n unerfüllbar ist.

Also ist die Annahme falsch, und es gibt keinen Tableaubeweis für $\mathfrak{F}$. □

3.5.4 Vollständigkeit zusichernde Eigenschaften

Bevor Eigenschaften formuliert werden können, die die Vollständigkeit eines Kalküls mit sich bringen, muß der Begriff des *voll expandierten* Astes definiert werden. Die Definition beruht darauf, daß der Kalkül eine Erweiterungsregel hat (Def. 3.3.4) und monoton ist (Def. 3.3.8); ohne diese Eigenschaften ist es schwierig, den Begriff des voll expandierten Astes in uniformer Weise zu definieren. Intuitiv ist ein Ast B voll expandiert, wenn jede mögliche Regelanwendung zumindest einen Nachfolgeast B' erzeugt, der die gleichen Formeln wie B enthält.

Definition 3.5.5 Sei $\mathcal{C}$ ein monotoner Tableaukalkül mit Erweiterungsregel für eine Logik **L**; und sei $\Sigma \in Sig$ eine Signatur.

Ein (ggf. unendlicher) Tableauast B ist *voll expandiert*, wenn $E \subset Form(B)$ für mindestens eine Extensionen E in jeder Konklusion $C \in \mathcal{R}(\Sigma)(B)$ gilt. □

Definition 3.5.6 Sei $\mathcal{C}$ ein Kalkül für eine Logik **L**. Die folgenden *Vollständigkeitseigenschaften* von $\mathcal{C}$ seien definiert:

Vollständigkeitseigenschaft 1 ([schwache] Angemessenheit der Menge der Tableauinterpretationen): Für alle Signaturen Σ von **L** gilt, daß, wenn es eine Tableauinterpretation in $TabInterp(\Sigma^*)$ gibt, die die initialen Tableaus für eine Menge $\mathfrak{F}$ von Formeln erfüllt, es auch ein Modell $\mathcal{M}(\Sigma)$ gibt, das $\mathfrak{F}$ erfüllt.

Vollständigkeitseigenschaft 2 (Erfüllbarkeit voll expandierter Äste): Für alle Signaturen Σ von **L** gilt, daß, wenn ein Ast B voll expandiert ist und nicht geschlossen, es auch eine Tableauinterpretation in $TabInterp(\Sigma^*)$ gibt, die B erfüllt. □

Die Vollständigkeitseigenschaften aus Definition 3.5.6 sind hinreichend, um die Vollständigkeit eines *gutartigen* Kalküls sicherzustellen. Wie aber schon am Ende von Abschnitt 3.3.7 gesagt, wird Gutartigkeit oft fallen gelassen, wenn ein gutartiger Kalkül $\mathcal{C}$ in eine effiziente Beweisprozedur umgesetzt wird. Das ist nicht schädlich, weil die Vollständigkeit dieser nicht-gutartigen Prozeduren aus derjenigen des Kalküls $\mathcal{C}$ folgt, wenn die uniformen Methoden zur Umsetzung eines Kalküls in eine Beweisprozedur angewendet werden, die in Kapitel 5 beschrieben sind.

Satz 3.5.7 *Ein Tableaukalkül, der gutartig ist und die Vollständigkeitseigenschaft aus Definition 3.5.6 (Angemessenheit der Menge der Tableauinterpretationen und Erfüllbarkeit voll expandierter Äste) hat, ist vollständig.*

Beweis: Sei $\mathfrak{F} \subset Form(\Sigma)$ eine endliche unerfüllbare Menge von Formeln; wir beweisen, daß es einen Tableaubeweis für $\mathfrak{F}$ gibt. Sei $(T_n)_{n \geq 0}$ eine (möglicherweise unendliche) Folge von Tableaus für $\mathfrak{F}$, so daß

1. T_0 ein initiales Tableau für $\mathfrak{F}$ ist,
2. T_i ein Nachfolgetableau von T_{i-1} ist ($i \geq 0$),
3. die Folge in fairer Weise konstruiert ist, d. h., jede mögliche Konklusion in $\mathcal{E}(\Pi)$ für alle Prämissen Π, die auf einem Ast in einem der T_i auftreten, ist früher oder später benutzt worden, um jeden nicht geschlossenen Ast zu erweitern, auf dem Π vorkommt (wobei sich entsprechende Äste in T_i und T_{i+j} identifiziert werden).

Da $\mathcal{C}$ monoton und nicht-destruktiv ist, kann eine solche Folge für alle Formelmenge $\mathfrak{F}$ konstruiert werden.

Die Folge $(T_n)_{n\geq 0}$ approximiert einen unendlichen Baum T_∞; gemäß seiner Konstruktion ist jeder nicht geschlossene Ast in T_∞ voll expandiert (Def. 3.5.5). Angenommen, es gibt einen nicht geschlossenen (und also voll expandierten) Ast B in T_∞. Dann wird B gemäß der Erfüllbarkeit voll expandierter Äste (Eigenschaft 2 in Def. 3.5.6) von einer Tableauinterpretation $\langle \mathbf{m}^*, I\rangle \in \mathit{TabInterp}(\Sigma^*)$ erfüllt. Da alle im initialen Tableau T_0 auftretenden Formeln auf B sind, wird auch T_0 von $\langle \mathbf{m}^*, I\rangle$ erfüllt. Darum gibt es gemäß der Angemessenheit der Menge der Tableauinterpretationen (Eigenschaft 1 in Def. 3.5.6) ein Modell $\mathbf{m} \in \mathcal{M}(\Sigma)$, das $\mathfrak{F}$ erfüllt, – im Widerspruch zur Voraussetzung, daß $\mathfrak{F}$ unerfüllbar ist. Also ist die Annahme falsch, daß es einen nicht geschlossenen Ast in T_∞ gibt, und tatsächlich enthalten alle Äste eine Knoten, der mit $\perp$ markiert ist. Nun impliziert aber Königs Lemma, weil Prämissen per Definition endlich sind, daß es ein $n \geq 0$ gibt, so daß schon das endliche Teiltableau T_n von T_∞ geschlossen ist. Damit ist der Beweis des Theorems abgeschlossen, denn T_n ist per Konstruktion ein geschlossenes Tableau für $\mathfrak{F}$. $\square$

3.5.5 Starke Korrektheit und Vollständigkeit zusichernde Eigenschaften

Es ist möglich, stärkere Versionen der Korrektheitseigenschaften aus Definition 3.5.3 zu formulieren. Die schwachen Eigenschaften verlangen Erfüllbarkeit der initialen Tableaus und, daß Tableauregelanwendungen die Erfüllbarkeit erhalten; sie erlauben aber, daß bei jedem Schritt ein anderes Modell bzw. eine andere Tableauinterpretation gewählt wird. Wenn ein Kalkül die weiter unten definierten stärkeren Eigenschaften hat und die Formeln, für die ein Tableaubeweis konstruiert werden soll, ein Modell $\mathbf{m}$ haben, dann wird das initiale Tableau und alle Nachfolgetableaus von ein und derselben Tableauinterpretation erfüllt, die eine Erweiterung von $\mathbf{m}$ ist.

Diese stärkeren Korrektheitseigenschaften sind nicht notwendig, um die Korrektheit eines Kalküls beweisen zu können, aber sie sind wichtig für die Methode des Faserns (Kapitel 6).

Definition 3.5.8 Sei $\mathcal{C}$ ein Kalkül für eine Logik $\mathbf{L}$. Die folgenden *starken Korrektheitseigenschaften* von $\mathcal{C}$ seien definiert:

Starke Korrektheitseigenschaft 1 (Angemessenheit der Menge der Tableauinterpretationen): Für alle Signaturen Σ von $\mathbf{L}$ gilt, daß, wenn eine Menge $\mathfrak{F} \subset \mathit{Form}(\Sigma)$ von einem Modell $\mathbf{m} \in \mathcal{M}(\Sigma)$ erfüllt wird, es auch eine Tableauinterpretation $\langle \mathbf{m}^*, I\rangle \in \mathit{TabInterp}(\Sigma^*)$ gibt, die die initialen Tableaus für $\mathfrak{F}$ erfüllt, wobei $\mathbf{m}^*$ eine Erweiterung von $\mathbf{m}$ ist.

Starke Korrektheitseigenschaft 2 (Korrektheit der Erweiterung): Für alle Signaturen Σ von $\mathbf{L}$ gilt, daß ein Nachfolgetableau T' eines Tableaus T von *denselben* Tableauinterpretationen in $\mathit{TabInterp}(\Sigma^*)$ erfüllt wird wie T. $\square$

Ein Kalkül hat die starke Korrektheitseigenschaft 2 (Korrektheit der Erweiterung), wie sie oben definiert ist (und also die schwache Eigenschaft aus Def. 3.5.3) genau dann, wenn die Konklusionen C in $\mathcal{E}(\Sigma)(B)$ logische Konsequenzen der Formeln auf dem erweiterten Ast B sind. Ein *gutartiger* Kalkül hat diese Eigenschaft genau dann, wenn alle Konklusionen eine logische Konsequenz ihrer minimalen Prämisse sind.

Lemma 3.5.9 *Ein gutartiger Kalkül $\mathcal{C}$ für eine Logik $\mathbf{L}$ hat die starke Korrektheitseigenschaft 2 aus Definition 3.5.8 genau dann, wenn für alle Signaturen $\Sigma \in \mathit{Sig}$ von $\mathbf{L}$, alle Prämissen $\Pi \subset \mathit{TabForm}(\Sigma^*)$ und alle Konklusionen C, so daß Π eine (minimale) Prämisse von C ist, das folgende gilt:*

> *Wenn Π von einer Tableauinterpretation erfüllt wird,*
> *dann wird eine Extension $E \in C$ von dieser Tableauinterpretation erfüllt.*

Beweis: Um den „wenn"-Teil des Lemmas zu beweisen, sei $\langle \mathbf{m}, I \rangle \in \mathit{TabInterp}(\Sigma^*)$ eine Tableauinterpretation, die ein Tableau T erfüllt, und sei T' ein Nachfolgetableau von T; sei B der Ast in T, der erweitert wird, und sei $\Pi \subset \mathit{Form}(B)$ die minimale Prämisse der Konklusion C, die benutzt wird, um B zu erweitern.

Wir haben angenommen, daß T von $\langle \mathbf{m}, I \rangle$ erfüllt wird; darum wird ein Ast B^0 in T von dieser Tableauinterpretation erfüllt. Falls B^0 von B verschieden ist, dann ist B^0 auch ein Ast in T', und wir sind fertig.

Andernfalls, wenn $B^0 = B$ und also die Tableauinterpretation $\langle \mathbf{m}, I \rangle$ die Prämisse $\Pi \subset \mathit{Form}(B)$ erfüllt, dann erfüllt sie auch eine der Extensionen in $E \in C$; darum erfüllt $\langle \mathbf{m}, I \rangle$ denjenigen Ast in T', der durch die Erweiterung von B mit den Formeln in E entstanden ist, und erfüllt also das Tableau T'.

Um den „genau dann"-Teil des Lemmas zu beweisen, betrachten wir ein Tableau T, das aus einem einzigen Ast besteht, der die Formeln in Π enthält. Falls Π erfüllt ist, dann ist auch T erfüllt, was impliziert, daß das Nachfolgetableau T' erfüllt ist, das aus T mit Hilfe der Konklusion C konstruiert wird. Nach Konstruktion kann aber T' nur erfüllt sein, wenn eine der Extensionen in C erfüllt ist. $\square$

Eine stärkere Version der Vollständigkeitseigenschaft 1 (Angemessenheit der Menge der Tableauinterpretationen) kann ebenfalls definiert werden.[1]

Definition 3.5.10 Sei $\mathcal{C}$ ein Kalkül für eine Logik $\mathbf{L}$. Die folgende *starke Vollständigkeitseigenschaft* von $\mathcal{C}$ sei definiert:

[1] Eine stärkere Version der Vollständigkeitseigenschaft 2 kann nicht in gleicher Weise definiert werden, da diese Eigenschaft nicht den Übergang von einem Modell zu einem anderen oder von einer Tableauinterpretation zu einer anderen involviert. In Abschnitt 3.5.7 wird die Eigenschaft „semantisch analytisch" definiert, die die Vollständigkeitseigenschaft 2 in anderer Weise verschärft.

Starke Vollständigkeitseigenschaft 1 (Angemessenheit der Menge der Tableauinterpretationen): Für alle Signaturen Σ von $\mathbf{L}$ gilt, daß, wenn eine Tableauinterpretation $\langle \mathbf{m}^*, I \rangle \in \mathit{TabForm}(\Sigma^*)$ die initialen Tableaus für eine Formelmenge $\mathfrak{F} \subset \mathit{Form}(\Sigma)$ erfüllt, $\mathbf{m}^*$ zu einem Modell $\mathbf{m} \in \mathcal{M}(\Sigma)$ eingeschränkt werden kann, das $\mathfrak{F}$ erfüllt. □

3.5.6 Invertierbare Erweiterungsregeln

Wenn die Umkehrung der Vorbedingung von Lemma 3.5.9 gilt, d. h., wenn jede Konklusion ihre minimale Prämisse logisch impliziert, dann nennen wir eine Tableauregel *invertierbar*.

Definition 3.5.11 Sei $\mathcal{C}$ ein gutartiger Kalkül für eine Logik $\mathbf{L}$; sei $\Sigma \in \mathit{Sig}$ eine Signatur von $\mathbf{L}$; und sei $\Pi \subset \mathit{TabForm}(\Sigma^*)$ eine Prämisse.

Die Tableauregel von $\mathcal{C}$ heiße bezüglich Π *invertierbar*, wenn für alle Konklusionen $C \in \mathcal{E}(\Sigma)(\Pi)$, so daß Π eine minimale Prämisse von C ist, das folgende gilt:

> Wenn eine Tableauinterpretation eine Extension $E \in C$ erfüllt,
> dann erfüllt sie auch Π.

Der Kalkül $\mathcal{C}$ heiße *invertierbar*, wenn seine Tableauregel für alle Signaturen Σ und alle Prämissen $\Pi \subset \mathit{TabForm}(\Sigma^*)$ invertierbar ist. □

Es ist von Vorteil, wenn eine Tableauregel invertierbar ist, weil dann Regelanwendungen nicht nur die Erfüllbarkeit des Tableaus erhalten, sondern auch seine *Unerfüllbarkeit*. Das ist wichtig für die Konstruktion effizienter Beweisprozeduren, weil es erlaubt, die minimale Prämisse einer Konklusion C von einem Ast zu „löschen“, der mit C erweitert worden ist.

Beispiel 3.5.12 Die Tableauregel eines Kalküls für PL1, die erlaubt, die Konklusion $\{\{\mathsf{T}{:}\sigma{:}F\}, \{\mathsf{T}{:}\sigma{:}G\}\}$ aus der Prämisse $\Pi_1 = \{\mathsf{T}{:}\sigma{:}(F \vee G)\}$ abzuleiten, ist invertierbar bzgl. Π_1, weil jede prädikatenlogische Interpretation, die eine der Formeln F und G erfüllt, auch $F \vee G$ erfüllt.

Andererseits ist die Tableauregel, die erlaubt, die Konklusion $\{\{\mathsf{T}{:}\sigma{:}p(t)\}\}$ aus der Prämisse $\Pi_2 = \{\mathsf{T}{:}\sigma{:}(\forall x)(p(x))\}$ abzuleiten, *nicht* invertierbar bzgl. Π_2, weil eine Interpretation, die $p(t)$ erfüllt, nicht notwendig auch $(\forall x)(p(x))$ erfüllt. □

3.5.7 Semantisch Analytische Tableaukalküle

Eine schärfere Version der Vollständigkeitseigenschaft 2 aus Definition 3.5.6 (Erfüllbarkeit voll expandierter Äste) kann formuliert werden, die sicherstellt, daß ein Kalkül „analytisch bis auf die atomare Ebene“ ist. Dies ist *keine* syntaktische Eigenschaft und sie impliziert *nicht*, daß der Kalkül im klassischen Sinne analytische ist (Def. 3.3.7).

Definition 3.5.13 Ein Kalkül $\mathcal{C}$ für eine Logik **L** heiße (schwach) *semantisch analytisch*, wenn für alle Signaturen Σ von **L** das folgende gilt: Wenn ein Tableauast B voll expandiert und nicht geschlossen ist, dann erfüllt jede Tableauinterpretation, die die *Atome* auf B erfüllt, *alle* Formeln auf B – und mindestens eine solche Tableauinterpretation existiert. $\square$

Beispiel 3.5.14 Wenn ein Tableaukalkül für eine Modallogik semantisch analytisch sein soll, dann muß es möglich sein, einen Ast, der die Formel $\mathsf{T}{:}\sigma{:}\Box p$ enthält, mit der Formel $\mathsf{T}{:}\tau{:}p$ für alle Markierungen τ zu erweitern, die Welten repräsentieren, die von der Welt erreichbar sind, die von σ repräsentiert wird. $\square$

Beispiel 3.5.15 Soll ein Tableaukalkül für klassische Aussagenlogik semantisch analytisch sein, dann muß es möglich sein, einen Ast, der die Formel $\mathsf{T}{:}(p \vee q)$ enthält durch Teiläste zu erweitern, die $\mathsf{T}{:}p$ bzw. $\mathsf{T}{:}q$ enthalten. Darum ist der Kalkül nicht semantisch analytisch, wenn die Einschränkung aus Abschnitt 5.5 benutzt wird, die es verbietet, Formeln für die Erweiterung zu verwenden, die keine *Konnektion* zu einer anderen Formel auf dem Ast haben. $\square$

Wenn ein Kalkül semantisch analytisch ist, dann repräsentieren die Atome auf einem voll expandierten Ast B explizit die gesamte Information über B erfüllende Tableauinterpretationen, die aus den (komplexen) Formeln auf B abgeleitet werden kann.

Es gibt jedoch in manchen Logiken versteckte (implizite) Einschränkungen bzgl. der Form von Modellen, die nicht für die Modelle, die einen bestimmten Ast erfüllen, spezifisch sind, sondern alle Tableauinterpretationen betreffen, die verwendet werden, um die Semantik von Tableaus zu definieren. Wenn ein Kalkül beispielsweise zum Fasern (Kapitel 6) benutzt wird, dann ist es wichtig, daß auch solche versteckten Informationen von den Atomen auf einem voll expandierten Ast explizit repräsentiert werden. Diese Eigenschaft wird wie folgt formalisiert:

Definition 3.5.16 Ein Kalkül $\mathcal{C}$ für eine Logik **L** heiße *stark semantisch analytisch*, wenn für alle Signaturen Σ von **L** das folgende gilt:

Wenn

1. B ein voll expandierter, nicht geschlossener Tableauast ist und

2. $\Phi \subset \mathit{TabForm}(\Sigma^*)$ eine Menge *atomarer* Tableauformeln ist, so daß für *kein* ϕ in Φ sowohl ϕ als auch $\overline{\phi}$ Elemente von $\mathit{Form}(B) \cup \Phi$ sind,

dann gibt es eine Tableauinterpretation $\langle \mathbf{m}, I \rangle$ in $\mathit{TabInterp}(\Sigma^*)$, so daß

1. $\langle \mathbf{m}, I \rangle$ die Menge $\mathit{Form}(B) \cup \Phi$ erfüllt und
2. es für alle Welten w in $\mathbf{m}$ eine Markierung σ in $\mathcal{C}$ mit $I(\sigma) = w$ gibt. □

Beispiel 3.5.17 Kripke-Modelle, die benutzt werden, um die Semantik intuitionistischer Logik zu definieren, müssen die (versteckte) Einschränkung erfüllen, daß, wenn eine Formel in einer Welt w wahr ist, sie auch in allen Nachfolgewelten von w wahr ist (wenn eine Formel falsch ist in einer Welt w, dann folgt daraus nichts über ihren Wahrheitswert in den Nachfolgewelten).

Wenn also ein Kalkül für intuitionistische Logik stark semantisch analytisch sein soll, dann muß es möglich sein, $\mathsf{T}{:}\tau{:}G$ aus $\mathsf{T}{:}\sigma{:}G$ für alle Markierungen τ abzuleiten, die eine Nachfolgewelt der von σ repräsentierten Welt repräsentieren. □

Für die meisten Logiken, einschließlich klassischer Logik und Modallogiken, sind die beiden Eigenschaften, stark bzw. schwach semantisch analytisch zu sein, äquivalent.

3.6 Ein gutartiger Tableaukalkül für PL1

3.6.1 Syntax

Um unseren Kalkül $\mathcal{C}_{\mathrm{PL1}}$ für die Prädikatenlogik erster Stufe zu beschreiben, müssen wir für jede Signatur $\Sigma \in \mathit{Sig}_{\mathrm{PL1}}$ die Erweiterung Σ^* definieren, die für die Konstruktion von Tableaus verwendet werden soll, außerdem die Menge der Markierungen, die initiale Markierung und natürlich die Tableauregel.

Erweiterte Signaturen Zum Skolemisieren benutzen wir eine Menge $F^{sko}(\Sigma)$ von Funktionssymbolen, die disjunkt von $F(\Sigma)$ ist und unendlich viele Symbole jeder Stelligkeit $n \geq 0$ enthält. Die Erweiterung einer Signatur $\Sigma = \langle P(\Sigma), F(\Sigma), \alpha(\Sigma) \rangle$ ist also

$$\Sigma^* = \langle P(\Sigma), F(\Sigma) \cup F^{sko}(\Sigma), \alpha(\Sigma) \cup \alpha^{sko}(\Sigma) \rangle \quad .$$

Markierungen Die Modelle der Prädikatenlogik erster Stufe bestehen aus nur einer Welt. Wir verwende die Markierung $*$, um diese einzelne Welt zu repräsentieren. Also ist $\mathit{Lab}(\Sigma) = \{*\}$ für alle Signaturen Σ, und $*$ ist die initiale Markierung. Um die Notation zu vereinfachen, benutzen wir die kürzere Schreibweise $\mathsf{S}{:}G$ für Tableauformeln in PL1-Kalkülen, d. h., die (immer gleiche) Markierung $*$ wird weggelassen.

α	α_1,	α_2
$\mathsf{T}{:}{*}{:}(F \wedge G)$	$\mathsf{T}{:}{*}{:}F$,	$\mathsf{T}{:}{*}{:}G$
$\mathsf{F}{:}{*}{:}(F \vee G)$	$\mathsf{F}{:}{*}{:}F$,	$\mathsf{F}{:}{*}{:}G$
$\mathsf{F}{:}{*}{:}(F \rightarrow G)$	$\mathsf{T}{:}{*}{:}F$,	$\mathsf{F}{:}{*}{:}G$
$\mathsf{T}{:}{*}{:}\neg F$	$\mathsf{F}{:}{*}{:}F$,	$\mathsf{F}{:}{*}{:}F$
$\mathsf{F}{:}{*}{:}\neg F$	$\mathsf{T}{:}{*}{:}F$,	$\mathsf{T}{:}{*}{:}F$

β	β_1,	β_2
$\mathsf{T}{:}{*}{:}(F \vee G)$	$\mathsf{T}{:}{*}{:}F$,	$\mathsf{T}{:}{*}{:}G$
$\mathsf{F}{:}{*}{:}(F \wedge G)$	$\mathsf{F}{:}{*}{:}F$,	$\mathsf{F}{:}{*}{:}G$
$\mathsf{F}{:}{*}{:}(F \rightarrow G)$	$\mathsf{F}{:}{*}{:}F$,	$\mathsf{T}{:}{*}{:}G$

$\gamma(x)$	$\gamma_1(x)$
$\mathsf{T}{:}{*}{:}(\forall x)(F(x))$	$\mathsf{T}{:}{*}{:}F(x)$
$\mathsf{F}{:}{*}{:}(\exists x)(F(x))$	$\mathsf{F}{:}{*}{:}F(x)$

$\delta(x)$	$\delta_1(x)$
$\mathsf{F}{:}{*}{:}(\forall x)(F(x))$	$\mathsf{F}{:}{*}{:}F(x)$
$\mathsf{T}{:}{*}{:}(\exists x)(F(x))$	$\mathsf{T}{:}{*}{:}F(x)$

Tabelle 3.1: Die vier Formeltypen der Prädikatenlogik erster Stufe.

Tableauregel Wir definieren einen gutartigen Kalkül, der also eine Erweiterungsregel $\mathcal{E}$ hat.

Die Menge der nicht atomaren Formeln in $Form(\Sigma^*) = Form^0_{\mathrm{PL1}}(\Sigma^*)$ wird in vier Klassen eingeteilt (Tabelle 3.1): die Klasse α der konjunktiven Formeln, die Klasse β der disjunktiven Formeln, die Klasse γ der universell quantifizierten Formeln und die Klasse δ der existentiell quantifizierten Formeln (*unifying notation*).

Notation 3.6.1 Die Buchstaben α, β, γ und δ werden ausschließlich benutzt, um Tableauformeln des entsprechenden Typs zu bezeichnen. Im Falle der γ- und δ-Formeln wird die von dem (äußersten) Quantoren gebundene Objektvariable x explizit gemacht, indem die Schreibweise $\gamma(x)$ und $\gamma_1(x)$ (bzw. $\delta(x)$ und $\delta_1(x)$) benutzt wird; entsprechend bezeichnet $\gamma_1(t)$ das Ergebnis der Ersetzung aller Vorkommen von x in γ_1 durch t. □

In Tabelle 3.2 ist die Erweiterungsregel des Kalküls $\mathcal{C}_{\mathrm{PL1}}$ für die vier Formeltypen schematisch dargestellt.

Damit der Kalkül monoton ist, benutzen wir ein Schema für δ-Formeln, das zur Konstruktion eines Skolem-Terms kein *neues* Skolem-Funktionssymbol verwendet. Statt dessen wird jeder Äquivalenzklasse von δ-Formeln, die bis auf Umbenennung gebundener Objektvariablen und Ersetzung von Grundtermen gleich sind, das gleiche Funktionssymbol zugewiesen. (Dies ist eine Anpassung der in (Beckert *et al.*, 1993) beschriebenen δ^{++}-Regel für den Fall, daß keine freien Variablen auftreten.)

Definition 3.6.2 Sei eine Signatur $\Sigma \in Sig_{\mathrm{PL1}}$ von PL1 gegeben. Eine *Skolem-Term-Zuweisung* ist eine Funktion sko, die jeder δ-Formel $\phi \in TabForm_{\mathrm{PL1}}(\Sigma^*)$ einen Term

$$sko(\phi) = f(t_1, \ldots, t_k) \in Term^0_{\mathrm{PL1}}(\Sigma^*)$$

zuweist, so daß

1. (a) $f \in F^{sko}(\Sigma)$,
   (b) $k = \alpha^{sko}_{\Sigma}(f)$ und
   (c) $t_1, \dots, t_k$ in ϕ auftretende (Teil-)Terme sind,[2]
2. wenn ein $f' \in F^{sko}(\Sigma)$ in ϕ vorkommt, $f > f'$ gilt, wobei $>$ eine beliebige aber feste Ordnung auf $F^{sko}(\Sigma)$ ist, und
3. für alle δ-Formeln $\psi \in Form^0_{\mathrm{PL1}}(\Sigma^*)$ gilt: wenn $sko(\psi) = f(t'_1, \dots t'_k)$, dann sind ϕ und ψ gleich bis auf die Umbenennung gebundener Objektvariablen und die Ersetzung von Vorkommen von Termen t_i durch t'_i $(1 \leq i \leq k)$. □

Der Zweck von Bedingung 2 in der obigen Definition von sko ist, Zyklen wie den folgenden zu verhindern:

– $sko(\phi) = f(t_1, \dots, t_k)$,
– das Symbol f kommt in ψ vor,
– $sko(\psi) = f'(t'_1, \dots, t'_l)$ und das Symbol f' kommt in ϕ vor.

Gemäß Bedingung 3 ist es erlaubt, dasselbe Skolem-Symbol für eine Äquivalenzklasse von δ-Formeln zu verwenden, die gleich sind bis auf (a) die Umbenennung gebundener Objektvariablen und (b) die Ersetzung variablenfreier Terme; die variablenfreien Terme, die ersetzt werden können, müssen zu Argumenten des Skolem-Terms gemacht werden.

Tatsächlich ist es nicht notwendig, komplexe Terme für die Skolemisierung zu verwenden. Es reicht aus, in eindeutiger Weise jeder δ-Formel (bzw. Klasse von δ-Formeln, die bis auf die Umbenennung gebundener Objektvariablen identisch sind) eine Skolem-*Konstante* zuzuweisen (unter Beachtung von Bedingung 2 in Def. 3.6.2). Die Möglichkeit, Skolem-*Terme* zu verwenden, spielt jedoch ein wichtige Rolle für das *Liften*, d. h., die Konstruktion einer Version des Kalküls $\mathcal{C}_{\mathrm{PL1}}$, die freie Variablen verwendet; dies wird von dem folgenden Beispiel illustriert:

Beispiel 3.6.3 Den δ-Formeln $\delta^t = \mathsf{T}{:}(\exists x)(p(t,x))$ kann das gleiche Skolem-Funktionssymbol f zugewiesen werden, wenn man die Skolem-*Terme* $sko(\delta^t) = f(t)$ (für alle Terme t) benutzt; in diesem Fall kann die Erweiterungsregel für Prämissen, die aus diesen δ-Formeln bestehen, *geliftet* werden.

Wenn statt dessen Skolem-*Konstanten* $sko(\delta^t) = c^t$ benutzt werden, dann kann die Erweiterungsregel nicht geliftet werden, weil $\delta^t[t \mapsto t'] = \delta^{t'}$ aber $\delta^t_1[t \mapsto t'] \neq \delta^{t'}_1$. □

Wir definieren nun formal die Erweiterungsregel $\mathcal{E}_{\mathrm{PL1}}$ des Kalküls $\mathcal{C}_{\mathrm{PL1}}$:

[2] Man beachte, daß die Terme t_i $(1 \leq i \leq k)$ Elemente von $Term^0_{\mathrm{PL1}}(\Sigma)$ sein müssen, also keine Objektvariablen enthalten.

$\dfrac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}}$	$\dfrac{\beta}{\beta_1 \mid \beta_2}$	$\dfrac{\gamma(x)}{\gamma_1(t)}$	$\dfrac{\delta(x)}{\delta_1(t)}$	$\dfrac{\begin{array}{c}\mathsf{T}{:}{*}{:}G\\ \mathsf{F}{:}{*}{:}G\end{array}}{\bot}$
		wobei t ein variablenfreier Term ist	wobei $t = sko(\delta)$	wobei G atomar ist

Tabelle 3.2: Erweiterungsregel-Schemata für Prädikatenlogik erster Stufe.

Definition 3.6.4 Für alle Signaturen Σ und alle Prämissen $\Pi \subset \mathit{TabForm}_{\mathrm{PL1}}(\Sigma^*)$, sei die Menge $\mathcal{E}_{\mathrm{PL1}}(\Sigma)(\Pi)$ möglicher Konklusionen die kleinste Menge, die die folgenden Konklusionen enthält (wobei $\alpha, \beta, \gamma, \delta$ Formeln des entsprechenden Typs bezeichnen):

– $\{\{\alpha_1, \alpha_2\}\}$ für alle $\alpha \in \Pi$,
– $\{\{\beta_1\}, \{\beta_2\}\}$ für alle $\beta \in \Pi$,
– $\{\{\gamma_1(t)\}\}$ für alle $\gamma \in \Pi$ und alle Terme $t \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$,
– $\{\{\delta(t)\}\}$ für alle $\delta \in \Pi$, wobei $t = sko(\delta)$ (Def. 3.6.2),
– $\{\{\bot\}\}$ falls $\mathsf{T}{:}\sigma{:}G, \mathsf{F}{:}\sigma{:}G \in \Pi$ für ein Atom $G \in \mathit{Atom}_{\mathrm{PL1}}(\Sigma^*)$.

□

Man beachte, daß die Formeln in $\mathit{Form}_{\mathrm{PL1}}(\Sigma) = \mathit{Form}^0_{\mathrm{PL1}}(\Sigma)$ prädikatenlogische *Sätze* sind, also keine freien Objektvariablen enthalten und daß die Erweiterungsregel des Kalküls $\mathcal{C}_{\mathrm{PL1}}$ auch keine freien Objektvariablen einführt.

3.6.2 Semantik

Um die Semantik der Tableaus von $\mathcal{C}_{\mathrm{PL1}}$ zu definieren, benutzen wir die aus allen *kanonischen* Tableauinterpretationen bestehende Menge $\mathit{TabInterp}_{\mathrm{PL1}}(\Sigma^*)$. Sie sind wie folgt definiert:

Definition 3.6.5 Eine Tableauinterpretation $\mathbf{m} = \langle\langle D, \mathcal{I}\rangle, I\rangle$ für eine PL1-Signatur Σ ist *kanonisch*, wenn folgendes gilt:

1. $D = \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$.
2. Für alle δ-Formeln $\delta(x) \in \mathit{TabForm}(\Sigma^*)$ gilt:
$$\text{Wenn } val_{\mathcal{I}}(\delta(x)) = true, \text{ dann } val_{\mathcal{I}}(\delta_1(t)) = true,$$
wobei $t = sko(\delta)$.
3. $I(*) = w^0 = \langle D, \mathcal{I}\rangle$. □

Intuitiv weist in einer kanonischen Tableauinterpretation die Interpretationsfunktion $\mathcal{I}$ den Skolem-Termen $t = sko(\delta)$ ein Element des Universums zu, für das die durch δ ausgedrückte Eigenschaft gilt; außerdem wir die Markierung $*$ von I in der richtigen Weise interpretiert.

3.6.3 Korrektheit und Vollständigkeit

Benutzt man die Menge $\mathit{TabInterp}_{\mathrm{PL1}}(\Sigma^*)$ der kanonischen Tableauinterpretationen, wie sie in Def. 3.6.5 definiert sind, dann hat der Kalkül $\mathcal{C}_{\mathrm{PL1}}$ die Korrektheits- und Vollständigkeitseigenschaften aus den Definitionen 3.5.3 und 3.5.6. Insbesondere werden alle Nachfolgetableaus eines Tableaus T von den gleichen kanonischen Tableauinterpretationen wie T erfüllt; und jeder voll expandierte, nicht geschlossene Ast wird von einer kanonischen Tableauinterpretation erfüllt.

Bevor wir unter Verwendung der Kriterien aus den Abschnitten 3.5.3 und 3.5.4 zeigen, daß der Kalkül $\mathcal{C}_{\mathrm{PL1}}$ korrekt und vollständig ist, formulieren und beweisen wir die entsprechende Version von Hintikkas Lemma.

Definition 3.6.6 Eine Menge $\Xi \subset \mathit{TabForm}_{\mathrm{PL1}}(\Sigma^*)$ von Tableauformeln ist eine *Hintikka-Menge*, wenn sie die folgenden Bedingungen erfüllt:

1. Es gibt keine komplementären atomaren Formeln $\mathsf{T}{:}G$, $\mathsf{F}{:}G$ in Ξ.
2. Wenn $\alpha \in \Xi$, dann sind α_1 und α_2 in Ξ.
3. Wenn $\beta \in \Xi$, dann ist β_1 oder β_2 in Ξ.
4. Wenn $\gamma(x) \in \Xi$, dann ist $\gamma_1(t)$ in Ξ für alle Grundterme t in $\mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$.
5. Wenn $\delta(x) \in \Xi$, dann ist $\delta_1(t)$ in Ξ, wobei $t = \mathit{sko}(\delta)$. □

Lemma 3.6.7 *Wenn Ξ eine Hintikka-Menge ist (Def. 3.6.6), dann*

1. *wird Ξ von einer Tableauinterpretation in $\mathit{TabInterp}(\Sigma^*)$ erfüllt;*
2. *wird Ξ von jeder Tableauinterpretation Ξ erfüllt, die die atomaren Formeln in Ξ erfüllt.*

Beweis: Der zweite Teil des Lemmas kann leicht durch Induktion über die Struktur der Tableauformeln in Ξ bewiesen werden.

Eine kanonische Tableauinterpretation $\langle\langle D, \mathcal{I}\rangle, I\rangle$, die die atomaren Formeln in Ξ erfüllt, kann wie folgt definiert werden, was – mit Hilfe des zweiten Teil des Lemmas – den ersten Teil des Lemmas beweist:

- $\langle D, \mathcal{I}\rangle$ ist eine Herbrand-Struktur, d. h., $D = \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$, und $\mathcal{I}(t) = t$ für alle Terme $t \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$.
- Für alle Atome $p(t_1, \ldots, t_{\alpha(p)})$ über Σ^* sei $p^{\mathcal{I}}(t_1, \ldots, t_{\alpha(p)}) = \mathit{true}$ genau dann, wenn $\mathsf{T}{:}p(t_1, \ldots, t_{\alpha(p)}) \in \Xi$, und sonst $p^{\mathcal{I}}(t_1, \ldots, t_{\alpha(p)}) = \mathit{false}$.

- $I(*) = \langle D, \mathcal{I} \rangle$.

Die Tableauinterpretation $\mathcal{I}$ ist wegen Bedingung 1 in der Definition von Hintikka-Mengen (Def. 3.6.6) wohldefiniert. □

Lemma 3.6.8 *Der Tableaukalkül $\mathcal{C}_{\mathrm{PL1}}$ hat die starke Korrektheitseigenschaft 1 aus Definition 3.5.8 (Angemessenheit der Menge der Tableauinterpretationen).*

Beweis: Sei $\langle D, \mathcal{I} \rangle$ eine prädikatenlogische Herbrand-Struktur, die eine Menge $\mathfrak{F}$ von Formeln erfüllt. Da die Skolem-Symbole in $F^{sko}(\Sigma)$ nicht in $\mathfrak{F}$ vorkommen, genügt es, ihre Interpretation so zu wählen, daß die entstehende Struktur $\langle D, \mathcal{I}^* \rangle$ Bedingung 2 in der Definition kanonischer Tableauinterpretationen entspricht, während zugleich die Interpretation der in Σ vorkommenden Symbole unverändert bleibt. Eine kanonische Tableauinterpretation, die die initialen Tableaus für $\mathfrak{F}$ erfüllt, kann dann konstruiert werden, indem man $\langle D, \mathcal{I}^* \rangle$ mit der Interpretation I von Markierungen kombiniert, die durch $I(*) = \langle D, \mathcal{I} \rangle$ definiert ist.

Der Rang $rk(f)$ der Symbole $f \in F^{sko}(\Sigma)$ sei wie folgt definiert:

- $rk(f) = 0$, wenn kein $\delta \in \mathit{TabForm}(\Sigma^*)$ mit $sko(\delta) = f(t_1, \ldots, t_k)$ für irgendwelche $t_1, \ldots, t_k \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$ existiert;

- $rk(f) = 1$, wenn $sko(\delta) = f(t_1, \ldots, t_k)$ für ein $\delta \in \mathit{TabForm}(\Sigma)$ und gewisse Terme $t_1, \ldots, t_k \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma)$;

- $rk(f) = 1 + \max\{rk(f') \mid f' \in F\}$, wobei F die Menge aller $f' \in F^{sko}(\Sigma)$ ist, die in einem $\delta \in \mathit{TabForm}(\Sigma^*)$ vorkommen, so daß $sko(\delta) = f(t_1, \ldots, t_k)$ für irgendwelche Terme $t_1, \ldots, t_k \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$.

Die Funktion rk ist wegen Bedingung 2 in Definition 3.6.2 wohldefiniert.

Wir definieren induktiv eine Folge $(\langle D, \mathcal{I}^n \rangle)_{n \geq 0}$ von prädikatenlogischen Strukturen, die alle das Universum D haben, wobei $\langle D, \mathcal{I}^n \rangle$ eine Struktur über der Signatur Σ^n ist, die die Einschränkung von Σ^* auf Funktionssymbole vom Rang nicht größer als n ist; die Interpretation $\mathcal{I}^{n+1}$ stimmt mit $\mathcal{I}^n$ auf allen Symbolen in $\Sigma^n \cup \Sigma$ überein.

Die initiale Interpretation $\mathcal{I}^0$ sei durch $f^{\mathcal{I}^0} = f^{\mathcal{I}}$ für alle $f \in F(\Sigma)$ definiert, und für alle $f \in F^{sko}(\Sigma)$ vom Rang 0 sei der Wert von $f^{\mathcal{I}^0}$ beliebig gewählt.

Die Symbole $f \in F^{sko}(\Sigma)$ vom Rang $r \leq n$ sind schon durch $\mathcal{I}^n$ interpretiert. Sei $f \in F^{sko}(\Sigma)$ ein Symbol vom Rang $n+1$; wir definieren $f^{\mathcal{I}^{n+1}}(b_1, \ldots, b_k)$ für alle $b_1, \ldots, b_k \in D$ durch: Wenn es Terme $t_1, \ldots, t_k \in \mathit{Term}^0(\Sigma^*)$ gibt, so daß $t_i^{\mathcal{I}^n} = b_i$ $(1 \leq i \leq k)$, und es eine Formel $\delta(x) \in \mathit{TabForm}(\Sigma^*)$ mit $f(t_1, \ldots, t_k) = sko(\delta(x))$ und $val_{\mathcal{I}^n}(\delta) = \mathit{true}$ gibt, dann wähle ein $e \in D$ mit $val_{\mathcal{I}^n, \{x \mapsto e\}}(\delta_1(x)) = \mathit{true}$ und definiere $f^{\mathcal{I}^{n+1}}(b_1, \ldots, b_k) = e$ (da f vom Rang $n+1$ ist, stammen die Symbole in δ alle aus der Signatur Σ^n). Andernfalls, wenn keine solchen Terme $t_1, \ldots, t_k$ und eine

Formel δ existieren, dann lege als Wert von $f^{\mathcal{I}^{n+1}}(b_1, \ldots, b_k)$ ein beliebiges Element von D fest.

Wenn andere Terme $t'_1, \ldots, t'_k$ und eine andere δ-Formel δ' existieren, die die obigen Bedingungen erfüllen, dann gilt

$$val_{\mathcal{I}^n}(\delta(x)) = val_{\mathcal{I}^n}(\delta'(x))$$

und

$$val_{\mathcal{I}^n,\{x \mapsto e\}}(\delta_1(x)) = val_{\mathcal{I}^n,\{x \mapsto e\}}(\delta'_1(x)) \ ,$$

weil $\mathcal{I}^n(t_i) = \mathcal{I}^n(t'_i) = b_i$ ($1 \leq i \leq k$) und die Formeln δ und δ' identisch bis auf die Umbenennung gebundener Objektvariablen und die Ersetzung von Termen t_i durch t'_i sind (Bedingung 3 in Def. 3.6.2).

Man kann sich die Folge $(\langle D, \mathcal{I}^n \rangle)_{n \geq 0}$ als eine Approximation an die prädikatenlogische Interpretation $\langle D, \mathcal{I}^* \rangle$ über Σ^* denken, wobei die Interpretation $\mathcal{I}^*$ auf den Symbolen in Σ^n mit $\mathcal{I}^n, \mathcal{I}^{n+1}, \ldots$ übereinstimmt. Sie erfüllt Bedingung 2 in Definition 3.6.2 per Konstruktion. □

Lemma 3.6.9 *Der Tableaukalkül $\mathcal{C}_{\mathrm{PL1}}$ hat die starke Korrektheitseigenschaft 2 aus Definition 3.5.8 (Korrektheit der Erweiterung).*

Beweis: Da der Kalkül gutartig ist, können wir Lemma 3.5.9 anwenden.

Sei $\Pi \subset \mathit{TabForm}_{\mathrm{PL1}}(\Sigma^*)$ eine minimale Prämisse einer Konklusion C. Und Π werde von einer Tableauinterpretation $\langle \mathbf{m}, I \rangle = \langle \langle D, \mathcal{I} \rangle, I \rangle \in \mathit{TabInterp}_{\mathrm{PL1}}(\Sigma^*)$ erfüllt. Wir zeigen, daß $\langle \mathbf{m}, I \rangle$ eine der Extensionen in C erfüllt, dabei werden die folgenden durch den Typ der Formeln in Π bestimmten Fälle unterschieden:

$\Pi = \{\alpha\}$: In diesem Fall ist $C = \{\{\alpha_1, \alpha_2\}\}$, und da α von $\langle \mathbf{m}, I \rangle$ erfüllt wird, folgt aus der Eigenschaft von α-Formeln (Def. 2.3.2), daß $\langle \mathbf{m}, I \rangle$ sowohl α_1 als auch α_2 erfüllt.

$\Pi = \{\beta\}$: In diesem Fall ist $C = \{\{\beta_1\}, \{\beta_2\}\}$, und da β von $\langle \mathbf{m}, I \rangle$ erfüllt wird, folgt aus der Eigenschaft von β-Formeln (Def. 2.3.2), daß $\langle \mathbf{m}, I \rangle$ mindestens eine der Formeln β_1 und β_2 erfüllt.

$\Pi = \{\gamma\}$: In diesem Fall ist $C = \{\{\gamma_1(t)\}\}$ für ein $t \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$, und da $\gamma(x)$ von $\langle \mathbf{m}, I \rangle$ erfüllt wird, folgt aus der Eigenschaft von γ-Formeln, daß $\langle \mathbf{m}, \mathcal{I} \rangle$ die Formel $\gamma_1(t)$ erfüllt.

$\Pi = \{\delta\}$: In diesem Fall ist $C = \{\{\delta_1(t)\}\}$ mit $t = \mathit{sko}(\delta)$, und da δ von $\langle \mathbf{m}, \mathcal{I} \rangle$ erfüllt wird und $\langle \mathbf{m}, \mathcal{I} \rangle$ kanonisch ist, erfüllt $\langle \mathbf{m}, \mathcal{I} \rangle$ die Formel $\delta_1(t)$.

$\Pi = \{\mathsf{T}{:}\sigma{:}F, \mathsf{F}{:}\sigma{:}F\}$: Eine Prämisse Π von dieser Form wird von keiner Tableauinterpretation erfüllt, was der Annahme widerspricht, daß Π von $\langle \mathbf{m}, \mathcal{I} \rangle$ erfüllt wird; dieser Fall kann also nicht eintreten. □

Lemma 3.6.10 *Der Tableaukalkül $\mathcal{C}_{\mathrm{PL1}}$ hat die starke Vollständigkeitseigenschaft 1 aus Definition 3.5.10 (Angemessenheit der Menge der Tableauinterpretationen).*

Beweis: Wenn $\langle \mathbf{m}^*, \mathcal{I} \rangle \in \mathit{TabForm}(\Sigma^*)$ ein initiales Tableau für $\mathfrak{F}$ erfüllt, dann erfüllt jede Einschränkung von $\mathbf{m}^*$ auf Σ die Formeln in $\mathfrak{F}$, weil die Symbole in $F^{sko}(\Sigma)$, die die einzigen zusätzlichen Symbole in Σ^* sind, nicht in $\mathfrak{F}$ vorkommen. □

Lemma 3.6.11 *Der Tableaukalkül $\mathcal{C}_{\mathrm{PL1}}$ hat Vollständigkeitseigenschaft 2 (Erfüllbarkeit voll expandierter Äste) aus Definition 3.5.6; und er ist semantisch analytisch.*

Beweis: Sei B ein voll expandierter, nicht geschlossener Ast; und sei Φ eine Menge atomarer Formeln aus $\mathit{TabForm}(\Sigma^*)$, so daß für kein ϕ in Φ sowohl ϕ als auch $\overline{\phi}$ ein Element von $\mathit{Form}(B) \cup \Phi$ ist.

Dann ist $\mathit{Form}(B) \cup \Phi$ eine Hintikka-Menge, und Lemma 3.6.7 impliziert, daß es eine Tableauinterpretation $\langle \mathbf{m}, I \rangle$ gibt, die $\mathit{Form}(B) \cup \Phi$ erfüllt. Also hat der Kalkül $\mathcal{C}_{\mathrm{PL1}}$ Vollständigkeitseigenschaft 2 aus Definition 3.5.6.

Weil die initiale Welt w^0 die einzige Welt in $\mathbf{m}$ ist, und per Definition $I(*) = w^0$, ist auch die zweite Bedingung in Definition 3.5.16 erfüllt; und der Kalkül ist tatsächlich stark semantisch analytisch. □

Satz 3.6.12 *Der Tableaukalkül $\mathcal{C}_{\mathrm{PL1}}$ für PL1 ist korrekt und vollständig.*

Beweis: Nach Sätzen 3.5.4 und 3.5.7 ist es ausreichend zu zeigen, daß $\mathcal{C}_{\mathrm{PL1}}$ die beiden Korrektheitseigenschaften aus Definition 3.5.3 und die beiden Vollständigkeitseigenschaften aus Definition 3.5.6 hat. Das folgt jedoch sofort aus Lemmata 3.6.8, 3.6.9, 3.6.10 und 3.6.11. □

3.7 Gutartige Tableaukalküle für Modallogiken

3.7.1 Einführung

Die ersten nicht-strukturellen Kalküle für Modallogiken wurden in (Fitting, 1983) beschrieben; sie verwenden Markierungen, um die Erreichbarkeitsrelation zwischen möglichen Welten zu repräsentieren (anstatt sie implizit in der Struktur der Tableaus zu kodieren). An Fittings Arbeiten anschließend wurden nicht-strukturelle Tableaukalküle für viele Modallogiken definiert, siehe (Goré, 1998) für eine Übersicht. Alle diese Kalküle sind nicht monoton, weil sie Tableauregeln verwenden, die, wenn sie auf eine π-Formel angewendet werden (eine Formeln, die die *Existenz* einer erreichbaren Welt zusichern, in der eine bestimmte Formel wahr ist), zur Skolemisierung der π-Formel eine Markierung einführen, die *neu* bzgl. des Astes oder sogar des Tableaus sein muß. Außerdem sind viele dieser Kalküle nicht gutartig, weil bei einer Erweiterungsregelanwendung *alle* Äste erweitert werden, auf denen die verwendete Prämisse liegt.

In Abschnitt 3.7.2 wird eine Tableaukalkül vorgestellt, der gutartig ist, weil jeder Formel F ihre eigene Markierung (eine Gödelisierung von F) zugewiesen wird. Diese wird zur Skolemisierung der π-Formel σ:$\mathsf{T}\Diamond F$ verwendet. Dieser Kalkül ist eine Grundversion (d. h., eine Version ohne freie Variablen) des Kalküls, der in (Beckert & Goré, 1997) beschrieben ist. Er kann als Beispiel dienen für die in Abschnitt 3.3.7 aufgestellte Behauptung, daß es oft nur kleine Änderungen erfordert, einen „leicht" nicht-gutartigen Kalkül in einen gutartigen Kalkül zu verwandeln.

Da in Kripke-Modellen für Modallogiken Welten nicht notwendig eine Nachfolgewelt haben, darf die Konklusion einer ν-Formel (die die Wahrheit einer Formel in allen erreichbaren Welten zusichert) nur solche Markierungen enthalten, die Welten repräsentieren, deren Existenz bekannt ist. Dieses Wissen wird aus den Markierungen anderer Formeln auf dem Ast abgeleitet, was zu einer Unstetigkeit der Tableauregel bzgl. aller Prämissen führt, die eine ν-Formel enthalten. In Abschnitt 3.7.4 wird eine Variante des Kalküls definiert, dessen Regel stetig ist bzgl. ν-Formeln; dort werden *bedingte* Markierungen verwendet, die nicht die Existenz der Welt, die sie darstellen, implizieren. Ein zusätzlicher Vorteil der Verwendung bedingter Markierungen ist, daß sie es erleichtern, eine Version des Kalküls, die freie Variablen verwendet, zu definieren. Die Überprüfung der Existenz von Welten wird dann Teil des Abschlusses eines Astes.

3.7.2 Markierungen für modallogische Kalküle

In diesem Abschnitt werden Markierungen definiert, die aus natürlichen Zahlen bestehen; sie werden häufig als Markierungen in Tableaukalkülen verwendet. Markierungen, die aus anderen Grundbausteinen aufgebaut sind, können in analoger Weise definiert werden.

Definition 3.7.1 Sei $\mathbb{N}$ die Menge der natürlichen Zahlen. Die Menge $Lab(\mathbb{N})$ der (nicht bedingten) *aus natürlichen Zahlen bestehenden Markierungen* ist wie folgt definiert:

- Das Wort 1 ist ein Element von $Lab(\mathbb{N})$.
- Wenn $\sigma \in Lab(\mathbb{N})$, dann ist das Wort $\sigma.n$ ein Element von $Lab(\mathbb{N})$ für alle $n \in \mathbb{N}$.

Die Menge $CondLab(\mathbb{N})$ der *bedingten aus natürlichen Zahlen bestehenden Markierungen* ist wie folgt definiert:

- Das Wort 1 ist ein Element von $CondLab(\mathbb{N})$;

- Wenn $\sigma \in CondLab(\mathbb{N})$, dann sind die Wörter $\sigma.n$ und $\sigma.(n)$ Elemente von $CondLab(\mathbb{N})$ für alle $n \in \mathbb{N}$.

Die *initiale Markierung* von $Lab(\mathbb{N})$ und $CondLab(\mathbb{N})$ ist 1.

Die *Länge* einer Markierung σ ist die Zahl der Punkte, die sie enthält, plus eins; sie sei mit $|\sigma|$ bezeichnet.

Die Komponenten einer Markierung σ heißen *Positionen* in σ. Eine Position ist *bedingt*, wenn sie von der Form (n) ist, und eine Markierung ist bedingt, wenn sie eine bedingte Position enthält.

Die Äquivalenklasse aller Markierungen in $CondLab(\mathbb{N})$, die mit einer Markierung σ bis auf das Vorhandensein von Klammerungen übereinstimmen, sei mit $[\sigma]$ bezeichnet.

Die Menge aller nicht-leeren *initialen Präfixe* einer Markierung σ, die σ selbst *nicht* einschließt, sei mit $ipr(\sigma)$ bezeichnet. □

Man beachte, daß $Lab(\mathbb{N}) \subset CondLab(\mathbb{N})$ und daß (1) *kein* Element von $CondLab(\mathbb{N})$ ist, weil 1 die initiale Welt in Modellen repräsentiert, die immer existiert.

Im folgenden unterscheiden wir manchmal nicht zwischen den Markierungen $\sigma.n$ and $\sigma.(n)$; in diesen Fällen benuzten wir $\sigma.[n]$, um anzudeuten, daß die Markierung beide Formen haben kann.

Definition 3.7.2 Eine Menge $\Gamma \subset Lab(\mathbb{N})$ von Markierungen ist *stark erzeugt*, wenn folgendes gilt:

1. die initiale Markierung 1 ist ein Element von Γ; und

2. $\sigma \in \Gamma$ impliziert $\tau \in \Gamma$ für alle $\tau \in ipr(\sigma)$. □

Die Markierungen in $Lab(\mathbb{N})$ stellen die Erreichbarkeitsrelation zwischen den Welten dar, für die sie stehen, indem eine Welt, die mit $\sigma.[n]$ bezeichnet ist, von der Welt erreichbar ist, die mit σ bezeichnet ist. Eine Menge stark erzeugter Markierungen kann man als Baum ansehen, dessen Wurzel 1 ist und in dem σ ein unmittelbarer Elternknoten von $\sigma.[n]$ ist.

Definition 3.7.3 Sei eine Modallogik **L** und eine stark erzeugte Menge $\Gamma \subset Lab(\mathbb{N})$ von Markierungen gegeben. Eine Markierung $\tau \in \Gamma$ ist von einer Markierung $\sigma \in \Gamma$ aus **L**-*zugänglich*, in Zeichen $\sigma \rhd \tau$, wenn die in Tabelle 3.3 aufgeführten Bedingungen erfüllt sind.

Eine Markierung $\sigma \in \Gamma$ ist eine **L**-*Sackgasse*, wenn kein $\tau \in \Gamma$ von σ aus **L**-zugänglich ist. □

Logik	$\sigma \rhd \tau$ genau dann, wenn	Logik	$\sigma \rhd \tau$ genau dann, wenn
K	$\tau = \sigma.[n]$	KT	$\tau = \sigma.[n]$ oder $\tau = \sigma$
KB	$\tau = \sigma.[n]$ oder $\sigma = \tau.[m]$	K4	$\tau = \sigma.\theta$
K5	$\tau = \sigma.[n]$, oder $\lvert\sigma\rvert \geq 2$, $\lvert\tau\rvert \geq 2$	K45	$\tau = \sigma.\theta$, oder $\lvert\sigma\rvert \geq 2$, $\lvert\tau\rvert \geq 2$
KD	K-Bedingung, oder σ ist eine K-Sackgasse und $\sigma = \tau$	KDB	KB-Bedingung, oder $\lvert\Gamma\rvert = 1$ und $\sigma = \tau = 1$
KD4	K4-Bedingung, oder σ ist eine K-Sackgasse und $\sigma = \tau$	KD5	K5-Bedingung, oder $\lvert\Gamma\rvert = 1$ und $\sigma = \tau = 1$
KD45	K45-Bedingung, oder $\lvert\Gamma\rvert = 1, \sigma = \tau = 1$	KB4	$\lvert\Gamma\rvert \geq 2$
B	$\tau = \sigma$, oder $\tau = \sigma.[n]$, oder $\sigma = \tau.[m]$	S4	$\tau = \sigma.\theta$ oder $\tau = \sigma$
S5	für alle σ, τ		

Tabelle 3.3: Die Zugänglichkeitsrelation auf Markierungen für die einfachen Modallogiken.

Das folgende Lemma zeigt, daß die **L**-Zugänglichkeitsrelation $\rhd$ auf Markierungen genau der Erreichbarkeitsrelation R auf **L**-Rahmen entspricht (siehe (Goré, 1998) für einen Beweis). Insbesondere hat $\rhd$ die Eigenschaften wie Reflexivität, Transitivität usw., die von den Axiomen der Logik **L** ausgedrückt werden (siehe Tabelle 2.1).

Lemma 3.7.4 *Sei* **L** *eine der einfachen Modallogiken. Wenn* $\Gamma \subset Lab(\mathbb{N})$ *eine stark erzeugte Menge von Markierungen ist, dann ist* $\langle \Gamma, \rhd \rangle$ *ein* **L**-*Rahmen, wobei* $\rhd$ *die* **L**-*Zugänglichkeitsrelation ist.*

3.7.3 Syntax und Semantik der Kalküle für Modallogiken

Die gutartigen Kalküle $\mathcal{C}_{\mathbf{L}}$ für die einfachen Modallogiken **L**, die in diesem Abschnitt vorgestellt werden, gründen auf den Markierungen verwendenden Kalkülen aus (Fitting, 1983). Der wesentliche Unterschied ist, daß wir, um die Kalküle monoton zu machen, ein Erweiterungsregel-Schema für π-Formeln verwenden, das keine neuen Markierungen einführt, sondern – ähnlich dem Schema für δ-Formeln in Abschnitt 3.6 – ein Symbol verwendet, das in eindeutiger Weise der Formel zugeordnet ist, auf die die Regel angewendet wird.

Erweiterte Signaturen Es ist nicht notwendig, die Signaturen zu erweitern; also ist $\Sigma = \Sigma^*$ für alle Modallogiken und alle Signaturen $\Sigma \in Sig_{\mathrm{mod}}$.

α	$\alpha_1,$	α_2
$\mathsf{T}{:}\sigma{:}(F \wedge G)$	$\mathsf{T}{:}\sigma{:}F,$	$\mathsf{T}{:}\sigma{:}G$
$\mathsf{F}{:}\sigma{:}(F \vee G)$	$\mathsf{F}{:}\sigma{:}F,$	$\mathsf{F}{:}\sigma{:}G$
$\mathsf{T}{:}\sigma{:}\neg F$	$\mathsf{F}{:}\sigma{:}F,$	$\mathsf{F}{:}\sigma{:}F$
$\mathsf{F}{:}\sigma{:}\neg F$	$\mathsf{T}{:}\sigma{:}F,$	$\mathsf{T}{:}\sigma{:}F$

β	$\beta_1,$	β_2
$\mathsf{T}{:}\sigma{:}(F \vee G)$	$\mathsf{T}{:}\sigma{:}F,$	$\mathsf{T}{:}\sigma{:}G$
$\mathsf{F}{:}\sigma{:}(F \wedge G)$	$\mathsf{F}{:}\sigma{:}F,$	$\mathsf{F}{:}\sigma{:}G$

$\mathsf{S}{:}\sigma{:}\nu$	$\mathsf{S}{:}\sigma.n{:}\nu_K$	$\mathsf{S}{:}\sigma.n{:}\nu_4$	$\mathsf{S}{:}\sigma{:}\nu_T$
$\mathsf{T}{:}\sigma{:}\Box F$	$\mathsf{T}{:}\sigma.n{:}F$	$\mathsf{T}{:}\sigma.n{:}\Box F$	$\mathsf{T}{:}\sigma{:}F$
$\mathsf{F}{:}\sigma{:}\Diamond F$	$\mathsf{F}{:}\sigma.n{:}F$	$\mathsf{F}{:}\sigma.n{:}\Diamond F$	$\mathsf{F}{:}\sigma{:}F$

$\mathsf{S}{:}\tau.n{:}\nu$	$\mathsf{S}{:}\tau{:}\nu_{4^r}$	$\mathsf{S}{:}\tau{:}\nu_B$	$\mathsf{S}{:}\tau{:}\nu_5$
$\mathsf{T}{:}\tau.n{:}\Box F$	$\mathsf{T}{:}\tau{:}\Box F$	$\mathsf{T}{:}\tau{:}F$	$\mathsf{T}{:}\tau{:}\Box\Box F$
$\mathsf{F}{:}\tau.n{:}\Diamond F$	$\mathsf{F}{:}\tau{:}\Diamond F$	$\mathsf{F}{:}\tau{:}F$	$\mathsf{F}{:}\tau{:}\Diamond\Diamond F$

$\mathsf{S}{:}\sigma{:}\pi$	$\mathsf{S}{:}\sigma.n{:}\pi_1$
$\mathsf{T}{:}\sigma{:}\Diamond F$	$\mathsf{T}{:}\sigma.n{:}F$
$\mathsf{F}{:}\sigma{:}\Box F$	$\mathsf{F}{:}\sigma.n{:}F$

Tabelle 3.4: Die vier Formeltypen der Modallogiken.

Markierungen Die Menge $Lab(\Sigma)$ von Markierungen ist für alle Signaturen Σ die Menge $Lab(\mathbb{N})$ der (unbedingten) aus natürlichen Zahlen bestehenden Markierungen (Def. 3.7.1).

Erweiterungsregel Es gibt vier Typen komplexer (nicht atomarer) Tableauformeln: α-Formeln (konjunktiv) und β-Formeln (disjunktiv) wie in Kalkülen für klassische Logik, ν-Formeln (die die Wahrheit einer Formel in *allen* erreichbaren Welten ausdrücken) und π-Formeln (die die Wahrheit einer Formel in *zumindest einer* erreichbaren Welt ausdrücken); siehe Tabelle 3.4.

Notation 3.7.5 Die Buchstaben ν and π werden ausschließlich verwendet, um Formeln des entsprechenden Typs zu bezeichnen. $\Box$

Die Erweiterungsregeln für die verschiedenen Modallogiken unterscheiden sich im wesentlichen in dem Regelschema für ν-Formeln, d. h., in den Konklusionen von Prämissen, die ν-Formeln enthalten. In Tabelle 3.5 ist die Erweiterungsregel für die verschiedenen Formeltypen schematisch dargestellt. Tabelle 3.6 faßt zusammen, welche Tableauformeln Teil der Konklusion einer Prämisse $\{\{\mathsf{S}{:}\sigma{:}\nu\}\}$ sind, die aus einer ν-Formel besteht; in dieser Tabelle bedeutet 4^d, daß ν_4 enthalten ist, falls $|\sigma| \geq 2$.

Wir geben die Definition der Erweiterungsregel für eine der Modallogiken formal an, nämlich für die Logik K; die formalen Definitionen der Erweiterungsregeln für die an-

$$\frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}} \qquad \frac{\beta}{\beta_1 \mid \beta_2} \qquad \frac{\mathsf{S}{:}\sigma{:}\pi}{\mathsf{S}{:}\sigma.\lceil\pi\rceil{:}\pi_1} \qquad \frac{\begin{array}{c}\mathsf{T}{:}\sigma{:}F\\ \mathsf{F}{:}\sigma{:}F\end{array}}{\perp}$$

wobei $n = \lceil F\rceil$, falls $\pi = \mathsf{T}{:}\sigma{:}\Diamond F$,
und $n = \lceil\neg F\rceil$, falls $\pi = \mathsf{F}{:}\sigma{:}\Box F$

$$\frac{\mathsf{S}{:}\sigma{:}\nu}{\mathsf{S}{:}\sigma.n{:}\nu_K}$$

wobei $\sigma.n$ auf dem Ast vorkommt; für alle Logiken

$$\frac{\mathsf{S}{:}\sigma{:}\nu}{\mathsf{S}{:}\sigma.n{:}\nu_4}$$

wobei $\sigma.n$ auf dem Ast vorkommt; für K4, KD4, S4, S5, und, falls $|\sigma| \geq 2$, für K5, KD5

$$\frac{\mathsf{S}{:}\sigma{:}\nu}{\mathsf{S}{:}\sigma{:}\nu_T}$$

für T, B, S4, S5.

$$\frac{\mathsf{S}{:}\tau.n{:}\nu}{\mathsf{S}{:}\tau{:}\nu_{4^r}}$$

für K5, KD5, K45 KD45, KB4, S5.

$$\frac{\mathsf{S}{:}\tau.n{:}\nu}{\mathsf{S}{:}\tau{:}\nu_B}$$

für KB, KDB, KB4, B.

$$\frac{\mathsf{S}{:}\tau.n{:}\nu}{\mathsf{S}{:}\tau{:}\nu_5}$$

falls $\tau = 1$, für K5, KD5.

Tabelle 3.5: Erweiterungsregel-Schemata für Modallogiken.

Logik	$\nu_{\mathbf{L}}$ für $\mathbf{L} =$	Logik	$\nu_{\mathbf{L}}$ für $\mathbf{L} =$
K, D	K	K45, KD45	K, 4, 4^r
T	K, T	KB4	K, B, 4, 4^r
KB, KDB	K, B	B	K, T, B
K4, KD4	K, 4	S4	K, T, 4
K5, KD5	K, 4^d, 4^r, 5	S5	K, T, 4, 4^r

Tabelle 3.6: Elemente $\nu_{\mathbf{L}}$ der Konklusionen einer ν-Formel.

deren Modallogiken kann leicht in analoger Weise aus ihren schematischen Beschreibungen gewonnen werden.

Definition 3.7.6 Für alle Prämissen $\Pi \subset \mathit{TabForm}_{\mathrm{mod}}(\Sigma)$ enthalte $\mathcal{E}_{\mathrm{K}}(\Sigma)(\Pi)$ die folgenden Konklusionen (wobei $\lceil\cdot\rceil$ eine beliebige aber feste Bijektion der Menge aller Formeln in die Menge der natürlichen Zahlen ist):

– $\{\{\alpha_1, \alpha_2\}\}$ für alle $\alpha \in \Pi$,
– $\{\{\beta_1\}, \{\beta_2\}\}$ für alle $\beta \in \Pi$,
– $\{\{\mathsf{T}{:}\sigma.n{:}F\}\}$ für alle $\mathsf{T}{:}\sigma{:}\Box F \in \Pi$ und alle Markierungen der Form $\sigma.n$, die in Π vorkommen,
– $\{\{\mathsf{F}{:}\sigma.n{:}F\}\}$ für alle $\mathsf{F}{:}\sigma{:}\Diamond F \in \Pi$ und alle Markierungen der Form $\sigma.n$, die in Π vorkommen,
– $\{\{\mathsf{F}{:}\sigma.n{:}F\}\}$ für alle $\mathsf{F}{:}\sigma{:}\Box F \in \Pi$, wobei $n = \lceil\neg F\rceil$,
– $\{\{\mathsf{T}{:}\sigma.n{:}F\}\}$ für alle $\mathsf{T}{:}\sigma{:}\Diamond F \in \Pi$, wobei $n = \lceil F\rceil$,
– $\{\{\bot\}\}$ falls $\mathsf{T}{:}\sigma{:}F, \mathsf{F}{:}\sigma{:}F \in \Pi$ für ein $F \in \mathit{Form}_{\mathrm{mod}}(\Sigma)$.

$\Box$

Semantik Um die Semantik von Tableaus für eine Modallogik **L** zu definieren, benutzen wir die Menge $\mathit{TabInterp}_{\mathbf{L}}(\Sigma^*)$, die aus Tableauinterpretationen besteht, die (a) **L**-Interpretationen und (b) kanonisch sind; Interpretationen sind kanonisch, wenn sie die durch Erweiterungsregelanwendungen auf π-Formeln erzeugte Markierungen in der richtigen Weise interpretieren.

Definition 3.7.7 Sei **L** eine der einfachen Modallogiken; sei $\Sigma \in \mathit{Sig}$ eine Signatur; und sei Lab die Menge $\mathit{CondLab}(\mathbb{N})$ der (bedingten und unbedingten) aus natürlichen Zahlen bestehenden Markierungen (Def. 3.7.1).

Eine Tableauinterpretation $\langle \mathbf{m}, I\rangle$ heiße eine **L**-*Interpretation*, wenn $\mathbf{m} = \langle W, R, V\rangle$ ein **L**-Modell ist und die die Markierungen interpretierende Funktion I die folgenden Eigenschaften hat:

1. $I(1) = w^0$ ist die initiale Welt in $\mathbf{m}$;
2. $I(\sigma.(n)) = I(\sigma.n)$ für alle $\sigma.n$ und $\sigma.(n)$ in Lab;
3. für alle $\sigma \in \mathit{Lab}$ gilt, daß, wenn $I(\tau)$ undefiniert ist für ein $\tau \in \mathit{ipr}(\sigma)$, auch $I(\sigma)$ undefiniert ist;
4. für alle $\sigma, \tau \in \mathit{Lab}$ gilt, daß, wenn (a) $\sigma \rhd \tau$ und (b) $I(\sigma)$ und $I(\tau)$ definiert sind, $I(\sigma)\, R\, I(\tau)$ gilt.

Eine **L**-Interpretation ist *kanonisch*, wenn zusätzlich folgendes gilt:

5. Für alle Markierungen $\sigma = \tau.n \in Lab$:

   Wenn $I(\tau)$ definiert ist und $I(\tau) \models \Diamond F$, dann ist $I(\sigma)$ definiert und $I(\sigma) \models F$,

   wobei F die Formel ist für welche $n = \lceil F \rceil$ ($\lceil \cdot \rceil$ ist die Bijektion der Formeln in die natürlichen Zahlen, die von der Erweiterungsregel verwendet wird). □

Da alle vorkommenden Mengen von Markierungen stark erzeugt sind, stellen die beiden Zusicherungen, daß jede **L**-Interpretation $\langle \mathbf{m}, \mathcal{I} \rangle$ die Markierung 1 definiert, und Bedingung 3 in der obigen Definition sicher, daß die Interpretationsfunktion I so viele Markierungen wie möglich in $ipr(\sigma)$ „definiert". Jedoch ist es absolut zulässig, daß $I(\tau.(n))$ für eine bedingte Markierung $\tau.(n)$ undefiniert ist, und zwar selbst dann, wenn $I(\tau)$ definiert ist.

Beispiel 3.7.8 Die Interpretationsfunktion I muß für die Markierungen 1 und 1.1 definiert sein. Für $1.(1)$ braucht sie nicht definiert zu sein; aber wenn sie es ist, dann muß auch $I(1.(1).1)$ definiert sein. □

Benutzt man die Menge $TabInterp_{\mathbf{L}}(\Sigma^*)$ kanonischer **L**-Interpretationen um die Semantik von Tableaus zu definieren, dann hat der Kalkül $\mathcal{C}_{\mathbf{L}}$ die Korrektheits- und Vollständigkeitseigenschaften aus Definitionen 3.5.3 und 3.5.6 (wobei **L** eine der einfachen Modallogiken ist). Wenn ein Tableau von einer Tableauinterpretation erfüllt wird, dann werden auch alle seine Nachfolgetableaus von derselben Tableauinterpretation erfüllt; und jeder voll expandierte, nicht geschlossene Ast wird von einer kanonischen **L**-Interpretation erfüllt.

Wiederum basiert der Beweis, daß voll expandierte Äste erfüllbar sind, auf einer geeigneten Version von Hintikkas Lemma:

Definition 3.7.9 Sei **L** eine der einfachen Modallogiken; und sei $\Sigma \in Sig_{\mathrm{mod}}$ eine Signatur. Eine Menge $\Xi \subset TabForm_{\mathrm{mod}}(\Sigma^*)$ von Tableauformeln, die keine bedingten Markierungen enthält, ist eine **L**-*Hintikka-Menge*, wenn sie die folgenden Bedingungen erfüllt:

1. Es gibt keine komplementären atomaren Formeln $\mathsf{T}{:}\sigma{:}p$ und $\mathsf{F}{:}\sigma{:}p$ in Ξ.
2. Wenn $\alpha \in \Xi$, dann sind α_1 und α_2 in Ξ.
3. Wenn $\beta \in \Xi$, dann ist zumindest eine der Formeln β_1 und β_2 in Ξ.
4. Wenn $\mathsf{S}{:}\sigma{:}\nu \in \Xi$, dann ist $\mathsf{S}{:}\tau{:}\nu_{\mathrm{K}}$ in Ξ für *alle* $\tau \in Lab$, für die $\sigma \rhd \tau$ gilt.
5. Wenn $\mathsf{S}{:}\sigma{:}\pi \in \Xi$, dann ist $\mathsf{S}{:}\tau{:}\pi_1$ in Ξ für *wenigstens ein* $\tau \in Lab$, für das $\sigma \rhd \tau$ gilt. □

Lemma 3.7.10 *Sei* $\mathbf{L}$ *eine der einfachen Modallogiken; und sei* $\Sigma \in Sig_{\mathrm{mod}}$ *eine Signatur. Wenn* Ξ *eine* $\mathbf{L}$*-Hintikka-Menge ist (Def. 3.7.9), dann*

1. *wird* Ξ *von einer Tableauinterpretation in* $TabInterp_{\mathbf{L}}(\Sigma^*)$ *(d. h., einer kanonischen* $\mathbf{L}$*-Interpretation) erfüllt und*

2. *wird* Ξ *von jeder Tableauinterpretation erfüllt, die die atomaren Formeln in* Ξ *erfüllt.*

Beweis: Der zweite Teil des Lemmas kann leicht durch Induktion über die Struktur der Tableauformeln in Ξ bewiesen werden.

Wegen Bedingung 1 in der Definition modaler Hintikka-Mengen (Def. 3.7.9) kann eine kanonische Tableauinterpretation $\langle\langle W, R, V\rangle, I\rangle$, die die atomaren Formeln in Ξ erfüllt, wie folgt definiert werden, was – mit Hilfe des zweiten Teil des Lemmas – den ersten Teil des Lemmas beweist:

- Sei W die Menge Lab aller Markierungen, die in Ξ vorkommen;
- sei $I(\sigma) = \sigma$, falls $\sigma \in W$; und sei $I(\sigma)$ andernfalls undefiniert;
- für alle $\sigma, \tau \in W$ gelte $\sigma \, R \, \tau$ genau dann, wenn $\sigma \rhd \tau$;
- sei $V(p) = \{\sigma \mid \mathsf{T}{:}\sigma{:}p \in \Xi\}$. □

Lemma 3.7.11 *Für alle einfachen Modallogiken* $\mathbf{L}$ *hat der Tableaukalkül* $\mathcal{C}_{\mathbf{L}}$ *die starke Korrektheitseigenschaft 1 aus Definition 3.5.8 (Angemessenheit der Menge der Tableauinterpretationen).*

Beweis: Sei $\mathbf{m} = \langle W, R, V\rangle$ ein $\mathbf{L}$-Modell, das die Formelmenge $\mathfrak{F}$ erfüllt. Wir wissen, daß $w^0 \models F$ für alle $F \in \mathfrak{F}$, wobei w^0 die initiale Welt in W ist.

Nun sei F_n für alle $n \in \mathbb{N}$ die Formel für die $n = \lceil F \rceil$ (wobei $\lceil \cdot \rceil$ die Bijektion der Menge der Formeln in die Menge der natürlichen Zahlen ist, die von der Erweiterungsregel verwendet wird), und I sei wie folgt definiert: Es gelte $I(1) = w^0$, und für jede Markierung der Form $\tau.n$ gelte:

- Wenn es eine Welt $w \in W$ gibt, die von $I(\tau)$ erreichbar ist, mit $w \models F_n$, dann setze $I(\tau.n) = I(\tau.(n)) = w$;
- sonst, wenn es keine solche Welt gibt, dennoch aber eine Welt w' existiert, die von $I(\tau)$ erreichbar ist, dann setze $I(\tau.n) = I(\tau.(n)) = w'$;
- andernfalls, wenn es keine Welt gibt, die von $\mathcal{I}(\tau)$ erreichbar ist, dann seien $I(\tau.n)$ und $I(\tau.(n)$ undefiniert.

Die $\mathbf{L}$-Interpretation $\langle \mathbf{m}, I\rangle$ ist per Konstruktion kanonisch, und außerdem erfüllt sie die Tableauformeln auf initialen Tableaus für $\mathfrak{F}$, weil $I(1) = w^0 \models F$ für alle $F \in \mathfrak{F}$. □

Lemma 3.7.12 *Für alle einfachen Modallogiken* $\mathbf{L}$ *hat der Tableaukalkül* $\mathcal{C}_{\mathbf{L}}$ *die starke Korrektheitseigenschaft 2 aus Definition 3.5.8 (Korrektheit der Erweiterung).*

Beweis: Da der Kalkül gutartig ist, können wir Lemma 3.5.9 anwenden.

Sei $\Pi \subset \mathit{TabForm}_{\mathrm{mod}}(\Sigma^*)$ eine minimale Prämisse einer Konklusion C; und eine kanonische $\mathbf{L}$-Interpretation $\langle \mathbf{m}, I\rangle \in \mathit{TabInterp}_{\mathbf{L}}(\Sigma^*)$ erfülle Π. Es ist leicht zu überprüfen (getrennt nach Fällen entsprechend der Form von Π), daß $\langle \mathbf{m}, I\rangle$ eine der Extensionen in C erfüllt. □

Lemma 3.7.13 *Für alle einfachen Modallogiken* $\mathbf{L}$ *hat der Tableaukalkül* $\mathcal{C}_{\mathbf{L}}$ *die starke Vollständigkeitseigenschaft 1 aus Definition 3.5.10 (Angemessenheit der Menge der Tableauinterpretationen).*

Beweis: Der Kalkül hat diese Eigenschaft trivialerweise, weil die Signaturen nicht erweitert worden sind (also $\Sigma^* = \Sigma$), und darum jedes $\mathbf{L}$-Modell in $\mathcal{M}(\Sigma^*)$ eine Einschränkung seiner selbst auf Σ ist. □

Lemma 3.7.14 *Für alle einfachen Modallogiken* $\mathbf{L}$ *hat der Kalkül* $\mathcal{C}_{\mathbf{L}}$ *die Vollständigkeitseigenschaft 2 (Erfüllbarkeit voll expandierter Äste) aus Definition 3.5.6; und er ist stark semantisch analytisch.*

Beweis: Sei B ein voll expandierter, nicht geschlossener Ast; und sei Φ eine Menge atomarer Formeln aus $\mathit{TabForm}_{\mathrm{mod}}(\Sigma^*)$, so daß für kein ϕ in Φ sowohl ϕ als auch $\overline{\phi}$ ein Element von $\mathit{Form}(B) \cup \Phi$ ist.

Dann ist $\mathit{Form}(B) \cup \Phi$ eine $\mathbf{L}$-Hintikka-Menge, und Lemma 3.7.10 impliziert, daß es eine Tableauinterpretation $\langle \mathbf{m}, I\rangle$ gibt, die $\mathit{Form}(B) \cup \Phi$ erfüllt. Also hat $\mathcal{C}_{\mathrm{PL1}}$ Vollständigkeitseigenschaft 2 aus Definition 3.5.6.

Gemäß der Konstruktion dieser Tableauinterpretation (siehe den Beweis des Lemmas 3.7.10) gilt $I(\sigma) = \sigma$ für alle $\sigma \in W$, und die zweite Bedingung in Definition 3.5.16 ist also erfüllt; und der Kalkül ist tatsächlich stark semantisch analytisch. □

Satz 3.7.15 *Für alle einfachen Modallogiken* $\mathbf{L}$ *ist der Tableaukalkül* $\mathcal{C}_{\mathbf{L}}$ *für* $\mathbf{L}$ *korrekt und vollständig.*

Beweis: Gemäß Sätzen 3.5.4 und 3.5.7 reicht es aus, zu zeigen, daß $\mathcal{C}_{\mathrm{L}}$ die beiden Korrektheitseigenschaften aus Definition 3.5.3 und die beiden Vollständigkeitseigenschaften aus Definition 3.5.6 hat. Das folgt jedoch sofort aus Lemmata 3.7.11, 3.7.12, 3.7.13 und 3.7.14. $\square$

Beispiel 3.7.16 Wir beweisen, daß

$$G = \Box(\neg p \vee q) \wedge \Box p \wedge (\Diamond\neg q \vee \Diamond\neg p)$$

in der Modallogik K unerfüllbar ist (und also ihre Negation $\neg G$ eine K-Tautologie ist). Abbildung 3.1 zeigt ein (voll expandiertes) geschlossenes Tableau, das Teil eines Tableaubeweises für (die K-Unerfüllbarkeit von) G ist. Die Knoten des Tableaus sind numeriert; ein Paar $[i; j]$ ist an den i-ten Knoten angefügt, die Nummer j deutet an, daß der Knoten i durch eine Anwendung der Erweiterungsregel auf eine Prämisse erzeugt wurde, die die Formel des Knotens j enthält.

Man beachte, daß, nachdem Formel 5 zum Tableau hinzugefügt ist, die einzige mögliche Konklusion (die nicht schon auf dem Tableau ist) diejenige ist, die aus Formel 5 ableitbar ist und aus den Formeln 6 und 7 besteht; die beiden ν-Formeln 3 und 4 können zu diesem Zeitpunkt nicht benutzt werden, weil das Tableau keine Markierungen der Form $1.n$ enthält.

Die Markierungen, die durch die Anwendung der Tableauregel auf die aus den π-Formeln 5 bzw. 6 bestehenden Prämissen eingeführt werden, sind $1.1 = 1.\lceil\neg q\rceil$ und $1.2 = 1.\lceil\neg p\rceil$. $\square$

3.7.4 Ein Kalkül für die Modallogik K mit einer bezüglich ν-Formeln stetigen Erweiterungsregel

Die im vorangegangenen Abschnitt vorgestellte Erweiterungsregel für Modallogiken ist nicht stetig bezüglich Prämissen, die ν-Formeln enthalten, weil die Bedingung beachtet werden muß, daß Markierungen, die durch Regelanwendungen auf eine ν-Formeln eingeführt werden, in der Prämisse schon vorkommen müssen. Beispielsweise kann aus der Prämisse $\Pi = \{\mathsf{T}{:}1{:}\Box p\}$ keine Konklusion abgeleitet werden, und ebenso kann nichts aus $\Pi' = \{\mathsf{T}{:}1.1{:}q\}$ abgeleitet werden; aus der Vereinigung von Π und Π' kann jedoch $\mathsf{T}{:}1.1{:}p$ abgeleitet werden.

Der wesentliche Nachteil einer bezüglich ν-Formeln nicht stetigen Erweiterungsregel ist, daß es unmöglich ist, eine Version des Kalküls zu definieren, die freie Variablen verwendet. Die ν-Formeln erlauben die Ableitung vieler ähnlicher Konklusionen, wie beispielsweise der Tableauformel $\mathsf{T}{:}1.n{:}p$ für alle Markierungen der Form $1.n$, die auf dem Ast vorkommen; alle diese Formeln durch eine einzige Formel $\mathsf{T}{:}1.X{:}p$ darzustellen, die eine freie Variable X enthält, ist nur möglich, wenn alle Instanzen von

- [**1**;–] $\mathsf{T}{:}1{:}\Box(\neg p \vee q) \wedge \Box p \wedge (\Diamond\neg q \vee \Diamond\neg p)$
- [**2**;1] $\mathsf{T}{:}1{:}\Box(\neg p \vee q)$
- [**3**;1] $\mathsf{T}{:}1{:}\Box p \wedge (\Diamond\neg q \vee \Diamond\neg p)$
- [**4**;3] $\mathsf{T}{:}1{:}\Box p$
- [**5**;3] $\mathsf{T}{:}1{:}\Diamond\neg q \vee \Diamond\neg p$
  - [**6**;5] $\mathsf{T}{:}1{:}\Diamond\neg q$
  - [**8**;6] $\mathsf{T}{:}1.1{:}\neg q$
  - [**9**;8] $\mathsf{F}{:}1.1{:}q$
  - [**10**;2] $\mathsf{T}{:}1.1{:}\neg p \vee q$
    - [**11**;10] $\mathsf{T}{:}1.1{:}\neg p$
    - [**13**;11] $\mathsf{F}{:}1.1{:}p$
    - [**14**;4] $\mathsf{T}{:}1.1{:}p$
    - [**15**;13,14] $\bot$
    - [**12**;10] $\mathsf{T}{:}1.1{:}q$
    - [**16**;9,12] $\bot$
  - [**7**;5] $\mathsf{T}{:}1{:}\Diamond\neg p$
  - [**17**;7] $\mathsf{T}{:}1.2{:}\neg p$
  - [**18**;17] $\mathsf{F}{:}1.2{:}p$
  - [**19**;4] $\mathsf{T}{:}1.2{:}p$
  - [**20**;18,19] $\bot$

Abbildung 3.1: Das Tableau aus Beispiel 3.7.16.

$\mathsf{T}{:}1.X{:}p$ zulässige Konklusionen sind (in unserem Beispiel ist das nicht der Fall für Instanzen $\mathsf{T}{:}1.n{:}p$, so daß $1.n$ *nicht* auf dem Ast vorkommt).

Dieses Problem kann vermieden werden, wenn man die Vorbedingung fallen läßt, daß alle Markierungen, die durch Regelanwendungen auf ν-Formeln eingeführt werden, schon in der Prämisse vorkommen müssen. Dann kann im obigen Beispiel, die Tableauformel $\mathsf{T}{:}1.(1){:}p$ aus Π abgeleitet werden – ob die Markierung 1.1 auf dem Ast vorkommt oder nicht.

Dann wird es aber notwendig, um die Korrektheit des Kalküls zu bewahren, die Existenz von Welten, die bedingten Markierungen entsprechen, später zu überprüfen, wenn Äste abgeschlossen werden. Zum Beispiel ist das (scheinbar) widersprüchliche Paar $\mathsf{T}{:}1.(1){:}p$ und $\mathsf{F}{:}1.(1){:}p$ nicht notwendig inkonsistent, weil die Welt $I(1.(1))$ in einer Tableauinterpretation möglicherweise nicht existiert. Bevor wir dieses Paar als inkonsistent bezeichnen können, müssen wir also sicherstellen, daß $I(1.(1))$ in allen $\mathbf{L}$-Interpretationen definiert ist, die den Tableauast B erfüllen, der geschlossen werden soll. Glücklicherweise kann dieses Wissen aus anderen Tableauformeln auf B abgeleitet werden. In unserem Beispiel würde eine Formel auf B wie beispielsweise $G = \mathsf{T}{:}1.1{:}q$ es „rechtfertigen", das Paar $\mathsf{T}{:}1.(1){:}p$ und $\mathsf{F}{:}1.(1){:}p$ zum Abschluß von B zu verwenden; denn jede $\mathbf{L}$-Interpretation $\langle \mathbf{m}, I\rangle$, die B erfüllt, muß auch G erfüllen, und also muß $I(1.(1)) = I(1.1)$ eine Welt in dem Modell $\mathbf{m}$ sein. Der wesentliche Punkt ist hier, daß die Markierung 1.1 von G gerade in der Position *unbedingt* ist,

$$\frac{\sigma{:}\nu}{\nu_K(\sigma.(n))}$$
für alle $n \in \mathbb{N}$

$$\frac{\begin{array}{c}\mathsf{T}{:}\sigma{:}F\\ \mathsf{F}{:}\sigma'{:}F\end{array}}{\bot}$$
wobei $[\sigma] = [\sigma']$, und
σ, σ' sind durch Formeln
auf dem Ast gerechtfertigt

Tabelle 3.7: Die neuen Regelschemata des Kalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ für die Logik K.

in der $1.(1)$ bedingt ist. Diese Überlegungen werden nun auf den allgemeinen Fall beliebiger Markierungen aus $CondLab(\mathbb{N})$ verallgemeinert.

Definition 3.7.17 Eine Markierung $\sigma \in CondLab(\mathbb{N})$ mit j-ter Position $[n_j]$ (wobei $1 \leq j \leq |\sigma|$) ist durch eine Menge $\Pi \in TabForm_{\mathrm{mod}}$ modaler Tableauformeln *gerechtfertigt*, wenn es eine Teilmenge Ψ von Π gibt, so daß für jedes j:

1. eine in Ψ vorkommende Markierung hat eine unbedingte aber sonst gleiche j-te Position n_j; und

2. für alle Markierungen τ, die in Ψ vorkommen, gilt: wenn $|\tau| \geq j$, dann ist die j-te Position in τ entweder n_j oder (n_j). $\square$

Die Bedingung, daß die Markierungen komplementärer atomarer Formeln gerechtfertigt sein müssen, um mit ihnen einen Ast abschließen zu können, macht die Erweiterungsregel unstetiger für Prämissen, die Paare solcher komplementären atomaren Formeln enthalten. Das ist aber nicht sonderlich problematisch, weil die Regel für solche Prämissen ohnehin unstetig ist.

Abgesehen von der abweichenden Erweiterungsregel und davon, daß $CondLab(\mathbb{N})$ statt $Lab(\mathbb{N})$ als Menge der Markierungen verwendet wird, stimmen Syntax und Semantik des neuen Kalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ für die Modallogik K mit Syntax und Semantik des Kalküls $\mathcal{C}_{\mathrm{K}}$ überein, der im vorangegangenen Abschnitt definiert worden ist. Die Signaturen werden nicht erweitert; und die Menge $TabInterp$ der Tableauinterpretationen, die die Semantik von Tableaus beschreiben, besteht aus den kanonischen **L**-Interpretationen (Def. 3.7.7).

Tabelle 3.7 zeigt die *neuen* Regelschemata für ν-Formeln und zum Astabschluß; *alle* Schemata der Erweiterungsregel des Kalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ sind in Tabelle 3.8 zusammengefaßt. Die Erweiterungsregel $\mathcal{E}_{\mathrm{K}}^{\mathrm{con}}$ von $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ ist wie folgt formal definiert.

Definition 3.7.18 Für alle Prämissen $\Pi \subset TabForm_{\mathrm{mod}}$ sei $\mathcal{E}_{\mathrm{K}}^{\mathrm{con}}(\Pi)$ die kleinste Menge, die die folgenden möglichen Konklusionen enthält (wobei $\lceil\cdot\rceil$ eine Bijektion von $Form_{\mathrm{mod}}(\Sigma)$ in die Menge der natürlichen Zahlen ist):

$$\frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}} \qquad \frac{\beta}{\beta_1 \mid \beta_2} \qquad \frac{\mathsf{T}{:}\sigma{:}\Box F}{\mathsf{T}{:}\sigma.(n){:}F} \qquad \frac{\mathsf{F}{:}\sigma{:}\Diamond F}{\mathsf{F}{:}\sigma.(n){:}F} \qquad \frac{\mathsf{T}{:}\sigma{:}\Diamond F}{\mathsf{T}{:}\sigma.n{:}F}$$

für alle $n \in \mathbb{N}$ — wobei $n = \lceil F \rceil$

$$\frac{\mathsf{F}{:}\sigma{:}\Box F}{\mathsf{F}{:}\sigma.n{:}F} \qquad \frac{\mathsf{T}{:}\sigma{:}\neg F}{\mathsf{F}{:}\sigma{:}F} \qquad \frac{\mathsf{F}{:}\sigma{:}\neg F}{\mathsf{T}{:}\sigma{:}F} \qquad \frac{\begin{array}{c}\mathsf{T}{:}\sigma{:}F\\ \mathsf{F}{:}\sigma'{:}F\end{array}}{\bot}$$

wobei $n = \lceil \neg F \rceil$

wobei $[\sigma] = [\sigma']$, und σ, σ' sind durch Formeln auf dem Ast gerechtfertigt

Tabelle 3.8: Regelschemata des Kalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$.

- $\{\{\alpha_1, \alpha_2\}\}$ für alle $\alpha \in \Pi$,
- $\{\{\beta_1\}, \{\beta_2\}\}$ für alle $\beta \in \Pi$,
- $\{\{\mathsf{T}{:}\sigma.(n){:}F\}\}$ für alle $\mathsf{T}{:}\sigma{:}\Box F \in \Pi$ und alle $n \in \mathbb{N}$,
- $\{\{\mathsf{F}{:}\sigma.(n){:}F\}\}$ für alle $\mathsf{F}{:}\sigma{:}\Diamond F \in \Pi$ und alle $n \in \mathbb{N}$,
- $\{\{\mathsf{F}{:}\sigma.n{:}F\}\}$ für alle $\mathsf{F}{:}\sigma{:}\Box F \in \Pi$, wobei $n = \lceil \neg F \rceil$,
- $\{\{\mathsf{T}{:}\sigma.n{:}F\}\}$ für alle $\mathsf{T}{:}\sigma{:}\Diamond F \in \Pi$, wobei $n = \lceil F \rceil$,
- $\{\{\bot\}\}$ falls es $\mathsf{T}{:}\sigma{:}F, \mathsf{F}{:}\sigma{:}F \in \Pi$ gibt, so daß $[\sigma] = [\sigma']$ und σ, σ' durch Π gerechtfertigt sind.

$\Box$

Die neuen Regelschemata für ν-Formeln und zum Astabschluß können auch für andere der einfachen Modallogiken verwendet werden, die (a) seriell sind (in diesem Fall ist der Rechtfertigungstest nicht notwendig, weil die Interpretation aller Markierungen definiert ist) oder die (b) weder symmetrisch noch euklidisch sind. Kalküle für symmetrische und euklidische Logiken sind problematisch, weil ihre Erweiterungsregeln Markierungen verkürzen können. Beispielsweise kann die Tableauformel $\mathsf{T}{:}1{:}p$ aus $\mathsf{T}{:}1.(1){:}\Box p$ abgeleitet werden, wenn die Logik symmetrisch ist. Die Semantik serieller Logiken garantiert, daß alle Markierungen tatsächlich Welten darstellen, aber in nicht-seriellen Logiken ist es möglich, daß die Markierung 1 definiert ist, die Markierung $1.(1)$ aber nicht. Darum wird ein zusätzlicher Mechanismus benötigt, um sicherzustellen, daß die Tableauformel $\mathsf{T}{:}1{:}p$ nur dann am Abschluß eines Astes beteiligt ist, wenn die Markierung $1.(1)$ tatsächlich definiert ist. Dies kann man erreichen, indem man eine zusätzliche Menge von Markierungen an jede Tableauformel anheftet, die alle gerechtfertigt sein müssen, wenn die Formel benutzt wird, um einen Ast abzuschließen (siehe (Beckert & Goré, 1997)).

Um die Vollständigkeit von $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ zu beweisen, muß die Definition von Hintikka-Mengen so verändert werden, daß sie die Möglichkeit berücksichtigt, daß Markierungen *bedingt* sind (nur Bedingung 1 weicht von der Definition modaler Hintikka-Mengen ohne bedingte Markierungen ab, siehe Def. 3.7.9).

Definition 3.7.19 Eine Menge $\Xi \subset \mathit{TabForm}_{\mathrm{mod}}(\Sigma^*)$ von Tableauformeln ist eine K-*Hintikka-Menge mit bedingten Markierungen*, wenn sie die folgenden Bedingungen erfüllt:

1. Es gibt keine komplementären atomaren Formeln $\mathsf{T}{:}\sigma{:}p$ und $\mathsf{F}{:}\sigma'{:}p$ in Ξ, so daß $[\sigma] = [\sigma']$ und σ, σ' sind durch Ξ gerechtfertigt.
2. Wenn $\alpha \in \Xi$, dann sind α_1 und α_2 in Ξ.
3. Wenn $\beta \in \Xi$, dann ist mindestens eine der Formeln β_1 und β_2 in Ξ.
4. Wenn $\mathsf{S}{:}\sigma{:}\nu \in \Xi$, dann ist $\mathsf{S}{:}\sigma.(n){:}\nu_{\mathrm{K}}$ in Ξ für *alle* $n \in \mathbb{N}$.
5. Wenn $\mathsf{S}{:}\sigma{:}\pi \in \Xi$, dann ist $\mathsf{S}{:}\sigma.n{:}\pi_1 \in \Xi$ für *wenigstens ein* $n \in \mathbb{N}$. $\square$

Lemma 3.7.20 *Wenn Ξ eine K-Hintikka-Menge mit bedingten Markierungen ist, dann*

1. *wird Ξ von einer Tableauinterpretation in $\mathit{TabInterp}_{\mathrm{K}}(\Sigma^*)$ (d. h., einer kanonischen K-Interpretation) erfüllt und*
2. *wird Ξ von jeder Tableauinterpretation erfüllt, die die atomaren Formeln in Ξ erfüllt.*

Beweis: Wiederum kann der zweite Teil des Lemmas leicht durch Induktion über die Struktur der Tableauformeln in Ξ bewiesen werden.

Wegen Bedingung 1 in der Definition modaler Hintikka-Mengen mit bedingten Markierungen (Def. 3.7.19) kann eine kanonische Tableauinterpretation $\langle\langle W, R, V\rangle, I\rangle$, die die atomaren Formeln in Ξ erfüllt, wie folgt definiert werden, was – mit Hilfe des zweiten Teil des Lemmas – den ersten Teil des Lemmas beweist:

- Sei $W = \{[\sigma] \mid \sigma \in \mathit{Lab}$ ist durch Ξ gerechtfertig$\}$;
- sei $I(\sigma) = [\sigma]$, falls $[\sigma] \in W$, und sei $I(\sigma)$ andernfalls undefiniert;
- für alle $\sigma, \tau \in W$ sei $[\tau]$ genau dann von $[\sigma]$ erreichbar, wenn $\tau = \sigma.n$ oder $\tau = \sigma.(n)$ für ein $n \in \mathbb{N}$;
- sei $V(p) = \{[\sigma] \mid \mathsf{T}{:}\sigma{:}p \in \Xi\}$. $\square$

Satz 3.7.21 *Der Tableaukalkül $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ für die Modallogik K ist korrekt und vollständig.*

Beweis: Dieser Satz kann in der gleichen Weise wie Korrektheit und Vollständigkeit des Kalküls $\mathcal{C}_{\mathrm{K}}$ (Theorem 3.7.15) bewiesen werden.

Der Beweis der Korrektheitseigenschaft 1 (Angemessenheit der Menge der Tableauinterpretationen) bleibt unverändert.

Zu zeigen, daß der Kalkül die starke Korrektheitseigenschaft 2 (Korrektheit der Erweiterung) hat, ist etwas schwieriger, weil nun bedingte Markierungen in einem Tableau auftreten können. Aber nur in dem Fall, daß eine Prämisse erlaubt, einen Ast abzuschließen, kann das zu Komplikationen führen. Dann muß das folgende Lemma angewendet werden (das unmittelbar aus den Definitionen folgt): Sei $\langle \mathbf{m}, I \rangle$ eine kanonische $\mathbf{L}$-Interpretation, und sei σ eine Markierung, die durch eine Menge Π von Tableauformeln gerechtfertig ist. Wenn $\langle \mathbf{m}, I \rangle$ die Formeln in Π erfüllt, dann ist $I(\sigma)$ definiert.

Die Vollständigkeitseigenschaften können in gleicher Weise bewiesen werden wie für den Kalkül $\mathcal{C}_{\mathrm{K}}$ – mit dem einzigen Unterschied, daß der Begriff der Hintikka-Menge aus Definition 3.7.19 und das entsprechende Lemma 3.7.20 verwendet werden, die die bedingte Markierungen berücksichtigen. □

Beispiel 3.7.22 Wir setzen Beispiel 3.7.22 fort und zeigen die Unerfüllbarkeit derselben Formel G wie dort, nun allerdings mit dem neuen Kalkül $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$. Abbildung 3.2 zeigt ein geschlossenes Tableau T^{con} für G, das mit Hilfe der Erweiterungsregel von $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ konstruiert ist.

Da die Bedingung fallen gelassen wurde, daß die Markierungen, die durch Anwendungen der Erweiterungsregel auf ν-Formeln eingeführt werden, schon auf dem Ast vorkommen, ist es nun möglich, die Formeln 6 und 7 zum Tableau hinzuzufügen, obwohl ihre Markierung $1.(1)$ noch nicht auf dem Ast vorkommt (sie ist jedoch in ihrer zweiten Position bedingt).

Der linke Ast B_1 von T^{con} wird geschlossen, indem die Erweiterungsregel auf die Prämisse angewendet wird, die aus dem komplementären Paar $\mathsf{T}{:}1.(1){:}p$ und $\mathsf{F}{:}1.(1){:}p$ in den Knoten 7 bzw. 14 besteht. Die Markierung $1.(1)$ dieser Atome ist auf B_1 durch die Formeln 10 und 11 gerechtfertigt, deren Markierung 1.1 ist. In diesem Fall enthalten die komplementären Formeln bedingte Markierungen, die nur durch eine dritte Formel auf dem Ast gerechtfertigt sind, so daß der Rechtfertigungstest tatsächlich unverzichtbar ist. Der mittlere Ast B_2 enthält die komplementären Formeln $\mathsf{F}{:}1.1{:}q$ und $\mathsf{T}{:}1.(1){:}q$ in den Knoten 11 bzw. 13. Deren Markierung ist wiederum durch die Formeln 10 und 11 gerechtfertigt; letztere ist in diesem Fall Teil des komplementären Paares. Der rechte Ast B_3 enthält das Paar $\mathsf{F}{:}1.2{:}p$ und $\mathsf{T}{:}1.(2){:}p$ komplementärer Atome in den Knoten 18 bzw. 19. Die Markierung $1.(2)$ der Formel in Knoten 19 ist durch Formel 18 gerechtfertigt. □

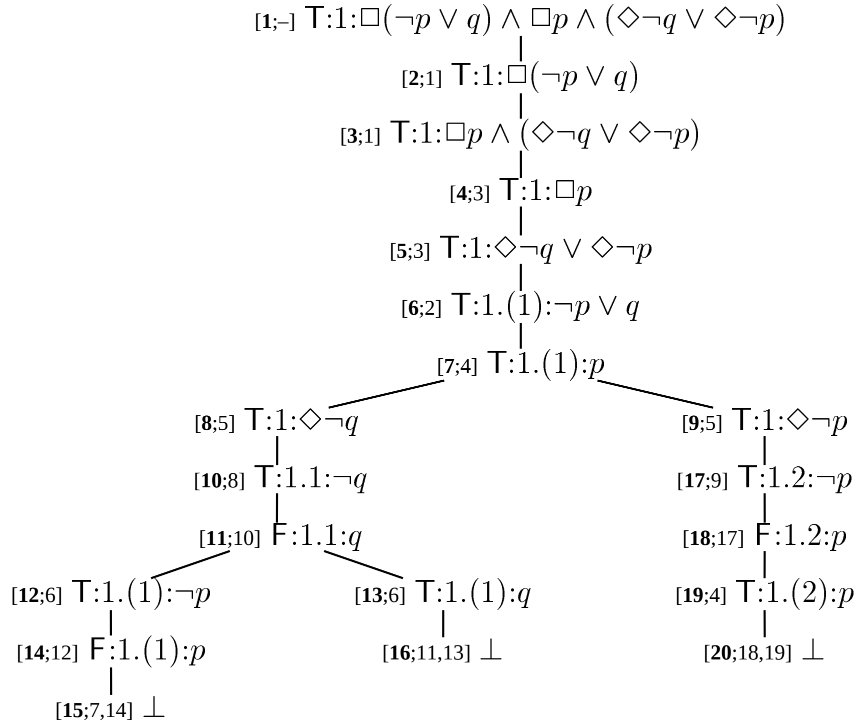


Abbildung 3.2: Das Tableau T^{con} aus Beispiel 3.7.22.

3.8 Gutartige Kalküle für die Fragmente MLSS und MLSSF der Mengentheorie

3.8.1 Einführung

In diesem Abschnitt wird ein gutartiger Tableaukalkül $\mathcal{C}_{\mathrm{MLSS}}$ für die Logik MLSS vorgestellt, die ein entscheidbares Fragment der Mengentheorie ist. Außerdem wird eine Erweiterung $\mathcal{C}_{\mathrm{MLSSF}}$ des Kalküls für das größere Fragment MLSSF beschrieben, das durch Anreicherung von MLSS mit freien (uninterpretierten) Funktionssymbolen entsteht (Abschnitt 2.6). Die gutartigen Kalküle $\mathcal{C}_{\mathrm{MLSS}}$ und $\mathcal{C}_{\mathrm{MLSSF}}$ sind leichte Variationen der in (Beckert & Hartmer, 1998; Hartmer, 1997) beschriebenen Kalküle, die nicht gutartig sind. Sie bauen auf den Tableaukalkülen für MLSS aus (Cantone, 1997) auf.

Der Kalkül $\mathcal{C}_{\mathrm{MLSS}}$ kann benutzt werden, um eine korrekte und vollständige Entscheidungsprozedur für MLSS zu konstruieren. Formeln müssen nicht in Normalform sein, während die Erweiterungsregel des von Cantone definierten Kalküls nur normalisierte MLSS-Atome handhaben kann (die keine komplexen Terme enthalten) und eine Vorverarbeitung erfordert, die die Eingabeformeln in normalisierte Formeln transformiert. Der erweiterte Kalkül $\mathcal{C}_{\mathrm{MLSSF}}$ für MLSSF verwendet E-Unifikationstechniken, um Termpaare zu finden, die erlauben einen Ast abzuschließen, wenn es gelingt zu zeigen, daß sie die gleiche Menge repräsentieren; auf diese Weise kann der Suchraum deutlich reduziert werden.

Mehrere andere Verfahren Behandlung der Mengentheorie mit Tableaukalkülen oder Sequenzenkalkülen sind vorgeschlagen worden (ohne Beschränkung auf ein bestimmtes Fragment): Brown (1978) stellte einen prädikatenlogischen Sequenzenkalkül vor, der Spezialregeln für viele mengentheoretische Symbole enthält. De Nivelle (1997) und Pastre (1978) präsentierten Sequenzenkalküle für Mengentheorie; Shults (1997) beschreibt einen Tableaukalkül mit speziellen mengentheoretischen Regeln. Alle diese Kalkül sind jedoch unvollständig (d. h., keine Semi-Entscheidungsverfahren).

3.8.2 Die Menge der Markierungen und die Erweiterung der Signaturen

Markierungen Die Modelle von MLSS und MLSSF bestehen aus nur einer Welt. Wir benutzen die Markierung $*$, um diese einzelne Welt zu repräsentieren. Also ist $Lab = \{*\}$, und $*$ ist die initiale Markierung. Wie in Kalkülen für PL1 verwenden wir die Kurzschreibweise $\mathsf{S}{:}G$ für Tableauformeln, d. h., die Markierung $*$ wird weggelassen.

Erweiterung des Signaturen Eine Ungleichung $\mathsf{F}{:}(s \approx t)$ impliziert die Existenz eines Elementes, das in einer der beiden Mengen s und t nicht aber in der anderen vorkommt. Die Erweiterungsregel unseres Kalküls nutzt diese Tatsache für die Einführung einer Skolem-Konstanten, die das existierende Element darstellt, wenn sie auf eine Ungleichung angewendet wird. Für die Skolemisierung verwenden wir eine unendlichen Menge $F^{sko}(\Sigma)$ von Konstanten, die disjunkt ist von $F(\Sigma)$.

Die Erweiterung einer MLSS- oder MLSSF-Signatur $\Sigma = \langle P(\Sigma), F(\Sigma), \alpha(\Sigma)\rangle$, die zur Konstruktion von Tableauformeln verwendet wird, ist also

$$\Sigma^* = \langle P(\Sigma), F(\Sigma) \cup F^{sko}(\Sigma), \alpha(\Sigma) \cup \alpha^{sko}(\Sigma)\rangle \quad ,$$

wobei $\alpha^{sko}(\Sigma)(c) = 0$ für alle $c \in F^{sko}(\Sigma)$.

3.8.3 Die Erweiterungsregel für MLSS

Schemata für nicht-atomare Formeln Die nicht-atomaren MLSS- und MLSSF-Formeln werden wie gewöhnlich in α- und β-Formeln eingeteilt. Die Erweiterungsregel $\mathcal{E}_{\mathrm{MLSS}}$ von $\mathcal{C}_{\mathrm{MLSS}}$ ist für aus α- und β-Formeln bestehende Prämissen durch die Schemata aus Tabelle 3.9 definiert (wir benutzen eine Verallgemeinerung, bei der die aus α- und β-Formeln abgeleiteten Konklusionen aus einer beliebigen Zahl von Tableauformeln bzw. Extensionen bestehen können).

$$\frac{\alpha}{\begin{array}{c}\alpha_1\\ \vdots\\ \alpha_n\end{array}} \qquad\qquad \frac{\beta}{\beta_1 \mid \cdots \mid \beta_n}$$

Tabelle 3.9: Verallgemeinerte Schemata für α- und β-Formeln.

Name	α	$\alpha_1, \ldots, \alpha_n$
(R1)	$\mathsf{T}{:}(s \sqsubseteq t)$	$\mathsf{T}{:}(s \approx s \sqcap t)$
(R2)	$\mathsf{F}{:}(s \sqsubseteq t)$	$\mathsf{F}{:}(s \approx s \sqcap t)$
(R3)	$\mathsf{T}{:}(s ⋿ t_1 \sqcap t_2)$	$\mathsf{T}{:}(s ⋿ t_1),\ \mathsf{T}{:}(s ⋿ t_2)$
(R4)	$\mathsf{T}{:}(s ⋿ t_1 \setminus t_2)$	$\mathsf{T}{:}(s ⋿ t_1),\ \mathsf{F}{:}(s ⋿ t_2)$
(R5)	$\mathsf{F}{:}(s ⋿ t_1 \sqcup t_2)$	$\mathsf{F}{:}(s ⋿ t_1),\ \mathsf{F}{:}(s ⋿ t_2)$
(R6)	$\mathsf{F}{:}(s ⋿ \{t_1, \ldots, t_n\}_n)$	$\mathsf{F}{:}(s \approx t_1), \ldots, \mathsf{F}{:}(s \approx t_n)$

Name	β	$\beta_1, \ldots, \beta_n$
(R7)	$\mathsf{T}{:}(s ⋿ t_1 \sqcup t_2)$	$\mathsf{T}{:}(s ⋿ t_1),\ \mathsf{T}{:}(s ⋿ t_2)$
(R8)	$\mathsf{F}{:}(s ⋿ t_1 \sqcap t_2)$	$\mathsf{F}{:}(s ⋿ t_1),\ \mathsf{F}{:}(s ⋿ t_2)$
(R9)	$\mathsf{F}{:}(s ⋿ t_1 \setminus t_2)$	$\mathsf{F}{:}(s ⋿ t_1),\ \mathsf{T}{:}(s ⋿ t_2)$
(R10)	$\mathsf{T}{:}(s ⋿ \{t_1, \ldots, t_n\}_n)$	$\mathsf{T}{:}(s \approx t_1), \ldots, \mathsf{T}{:}(s \approx t_n)$

Tabelle 3.10: Regelschemata für das Aufspalten komplexer Mengenterme.

Schemata für das Aufspalten komplexer Mengenterme Es gibt zehn verschiedene Erweiterungsregel-Schemata für das Aufspalten komplexer Mengenterme; sie wenden einfache mengentheoretische Lemmata an, wie z. B. „wenn $s \in t_1 \cup t_2$, dann $s \in t_1$ oder $s \in t_2$”, um (a) Atome, die das Ist-Teilmenge-von-Prädikat $\sqsubseteq$ enthalten, zu eliminieren und durch (Un-)Gleichungen zu ersetzen und (b) komplexe Terme auf der rechten Seite des Ist-Element-von-Prädikates $⋿$ in ihre Bestandteile aufzuspalten. Diese Schemata können als Instanzen der verallgemeinerten Schemata für α- und β-Formeln aus Tabelle 3.10 beschrieben werden; sie sind in Tabelle 3.10 aufgelistet.

Schemata für Gleichungen und Ungleichungen Es gibt drei Typen spezieller Erweiterungsregel-Schemata für Handhabung der Gleichheit und Ungleichheit von Mengen.

Erstens gibt es zwei Schemata ((EQ1) und (EQ2) in Tabelle 3.11), die die „Anwendung“ einer Gleichung $\mathsf{T}{:}(t_1 \approx t_2)$ auf andere Atome erlauben – allerdings nur in sehr eingeschränkter Weise: eine Gleichung kann nur auf die oberste Ebene (also nicht auf Teilterme) der rechten Seite eines positiven Atoms mit dem Prädikatensymbol $⋿$ angewendet werden. Das heißt, eine Gleichung kann nur angewendet werden, um eines der Atome $\mathsf{T}{:}(s ⋿ t_1)$ und $\mathsf{T}{:}(s ⋿ t_2)$ aus dem jeweils anderen abzuleiten. Diese Ein-

$$\frac{\mathsf{T}{:}(t_1 \approx t_2) \quad \mathsf{T}{:}(s \sqsubseteq t_1)}{\mathsf{T}{:}(s \sqsubseteq t_2)} \qquad \frac{\mathsf{T}{:}(t_1 \approx t_2) \quad \mathsf{T}{:}(s \sqsubseteq t_2)}{\mathsf{T}{:}(s \sqsubseteq t_1)} \qquad \frac{\mathsf{T}{:}(s_1 \sqsubseteq t) \quad \mathsf{F}{:}(s_2 \sqsubseteq t)}{\mathsf{F}{:}(s_1 \approx s_2)}$$

(EQ1) (EQ2) (R11)

$$\frac{\mathsf{F}{:}(t_1 \approx t_2)}{\begin{array}{c|c} \mathsf{T}{:}(c \sqsubseteq t_1) & \mathsf{F}{:}(c \sqsubseteq t_1) \\ \mathsf{F}{:}(c \sqsubseteq t_2) & \mathsf{T}{:}(c \sqsubseteq t_2) \end{array}}$$

wobei $c = sko_{\mathrm{MLSSF}}(\mathsf{F}{:}(t_1 \approx t_2))$
(R12)

$$\frac{}{\mathsf{T}{:}(s \sqsubseteq t) \mid \mathsf{F}{:}(s \sqsubseteq t)}$$

wobei s bzw. $\{\ldots, s, \ldots\}$
und t bzw. $\{\ldots, t, \ldots\}$
Top-level-Terme auf dem Ast sind
(Cut)

Tabelle 3.11: Regelschemata für Gleichungen und Ungleichungen und das Schnittregel-Schema.

schränkung ist wichtig, da die Möglichkeit, Gleichungen wahllos auf andere Atome anzuwenden, zu einem sehr viel größeren Suchraum führen würde.

Zweitens ist es möglich, die Ungleichung $\mathsf{F}{:}(s_1 \approx s_2)$ aus $\mathsf{T}{:}(s_1 \sqsubseteq t)$ und $\mathsf{F}{:}(s_2 \sqsubseteq t)$ abzuleiten ((R11) in Tabelle 3.11). Dieses Erweiterungsregel-Schema beruht auf der Tatsache, daß zwei Objekte verschieden sein müssen, wenn eines von ihnen in einer Menge vorkommt das andere aber nicht.

Drittens gilt auch das umgekehrte: Wenn zwei Mengen t_1 und t_2 verschieden sind, dann enthält eine von ihnen ein Element e, das nicht auch Element der anderen Menge ist. Unglücklicherweise führt dies zu einem verzweigenden Regelschema ((R12) in Tabelle 3.11), weil e eine Element von t_1 (und nicht von t_2) oder ein Element von t_2 (und nicht von t_1) sein kann. Statt ein *neues* Symbol einzuführen, das das unbekannte Element e repräsentiert, verwende wir eine *Skolem-Konstanten-Zuweisung*, die jeder Ungleichung auf eindeutige Weise eine Konstante zuordnet (ähnlich wie die Skolem-Term-Zuweisung aus Definition 3.6.2, die δ-Formeln einen Skolem-Term zuweist); auf diese Weise wird die Monotonie und also die Gutartigkeit der Erweiterungsregel bewahrt.

Definition 3.8.1 Sei eine MLSS- oder MLSSF-Signatur Σ gegeben. Eine *Skolem-Konstanten-Zuweisung* (für MLSS bzw. MLSSF) ist eine Funktion sko_{MLSSF}, die jeder Ungleichung $\phi = \mathsf{F}{:}(t_1 \approx t_2)$ in $TabForm(\Sigma^*)$ eine Konstante

$$sko_{\mathrm{MLSSF}}(\phi) = c \in F^{sko}(\Sigma)$$

zuweist, so daß für alle $c' \in F^{sko}(\Sigma)$ folgendes gilt: wenn c' in ϕ vorkommt, dann $c > c'$, wobei $>$ eine beliebige aber feste Ordnung auf $F^{sko}(\Sigma)$ ist. □

Das Schnittregel-Schema Das Schnittregel-Schema ((Cut) in Tabelle 3.11) kann angewendet werden, um einen Ast B zu erweitern, wobei nur solche Atome der Form $s \sqsubseteq t$ als Schnittformel verwendet werden können, deren Terme s und t

- Top-level-Terme (nicht nur Teilterme) eines Atoms auf B sind, oder
- direkte Teilterme von Termen der Form $\{\ldots, s, \ldots\}$ bzw. $\{\ldots, t, \ldots\}$ sind, die wiederum Top-level-Term eines Atoms auf B sind.

Das Schnittregel-Schema wird nur selten benötigt, um einen Beweis zu konstruieren; benötigt wird es beispielsweise, um implizite Enthaltenseins-Zyklen auf einem Ast aufzuspüren, siehe Abschnitt 3.8.3.

Beispiel 3.8.2 Wenn $\mathsf{T}{:}(t_1 \sqsubseteq \{t_2,\, t_3 \sqcap t_4\})$ und $\mathsf{T}{:}(t_5 \sqcap t_6 \approx t_7)$ Atome auf einem Ast sind, dann dürfen $t_1, t_2, (t_3 \sqcap t_4), (t_5 \sqcap t_6), t_7$ als Terme in Schnittformeln verwendet werden, und t_3, t_4, t_5, t_6 dürfen nicht verwendet werden (es sei denn, das wäre durch andere Atome auf dem Ast gerechtfertigt) □

Schemata für den Astabschluß Eine Anwendung der Erweiterungsregel des Kalküls $\mathcal{C}_{\mathrm{MLSS}}$ fügt Formeln zu einem Tableauast hinzu, die in allen Mengenstrukturen wahr sind, die Modelle des Astes sind; der Zweck der Regelschemata für den Astabschluß ist es, Inkonsistenzen aufzudecken, also Formeln oder Formelmengen, die in allen Mengenstrukturen falsch sind. Es gibt vier Arten solcher Inkonsistenzen, die berücksichtigt werden müssen:

1. In keiner Mengenstruktur können eine Formel ϕ und ihr Komplement $\overline{\phi}$ wahr sein; darum erlaubt eine Prämisse, die ein Paar $\phi, \overline{\phi}$ enthält, – wie in allen Kalkülen – die Ableitung der Konklusion $\{\{\perp\}\}$ (es ist für die Vollständigkeit des Kalküls ausreichend, nur komplementäre *Atome* zu berücksichtigen).
2. Kein Objekt ist ein Element der leeren Menge; darum ist jedes Atom der Form $\mathsf{T}{:}(t \sqsubseteq \emptyset)$ unerfüllbar.
3. Da kein Objekt von sich selbst verschieden ist, sind Atome der Form $\mathsf{F}{:}(t \approx t)$ unerfüllbar.
4. Die Existenz eines Enthaltenseins-Zyklus, d. h., von Menge $u_1, \ldots, u_k$, so daß $u_i \in u_{i+1}$ $(1 \leq i < k)$ und $u_k \in u_1$, würde dem Fundierungs-Axiom der Mengentheorie widersprechen. Tatsächlich gibt es per Konstruktion in der von-Neumann-Hierarchie keine Mengen, die einen Enthaltenseins-Zyklus bilden. Also erlauben Atome, die einen Enthaltenseins-Zyklus definieren, einen Ast abzuschließen; insbesondere ist $\mathsf{T}{:}(t \sqsubseteq t)$ unerfüllbar.

Im folgenden ist die Erweiterungsregel des Kalküls $\mathcal{C}_{\mathrm{MLSS}}$ formal definiert:

Definition 3.8.3 Sei Σ eine MLSS-Signatur.

Für alle Prämissen $\Pi \subset \mathit{TabForm}_{\mathrm{MLSS}}(\Sigma^*)$ sei $\mathcal{E}_{\mathrm{MLSS}}(\Sigma^*)(\Pi)$ die kleinste Menge, die die folgenden Konklusionen enthält:

– $\{\{\alpha_1, \ldots, \alpha_n\}\}$
  für alle $\alpha \in \Pi$ (Tabellen 3.1 und 3.10);

– $\{\{\beta_1\}, \ldots, \{\beta_n\}\}$
  für alle $\beta \in \Pi$ (Tabellen 3.1 und 3.10);

– $\{\{\mathsf{T}{:}(s \sqsubseteq t_2)\}\}$
  für alle (a) $\mathsf{T}{:}(s \sqsubseteq t_1)$ und (b) $\mathsf{T}{:}(t_1 \approx t_2)$ oder $\mathsf{T}{:}(t_2 \approx t_1)$ in Π;

– $\{\{\mathsf{F}{:}(s_1 \approx s_2)\}\}$
  für alle $\mathsf{T}{:}(s_1 \sqsubseteq t)$ und $\mathsf{F}{:}(s_2 \sqsubseteq t)$ in Π;

– $\{\{\mathsf{T}{:}(s \sqsubseteq t)\}, \{\mathsf{F}{:}(s \sqsubseteq t)\}\}$
  für alle Mengenterme s und t, so daß

  1. s oder $\{\ldots, s, \ldots\}_n$ und
  2. t oder $\{\ldots, t, \ldots\}_n$

  als Top-level-Terme in Atomen in Π auftreten;

– $\{\{\mathsf{T}{:}(c \sqsubseteq t_1)\,, \mathsf{F}{:}(c \sqsubseteq t_2)\}, \{\mathsf{F}{:}(c \sqsubseteq t_1)\,, \mathsf{T}{:}(c \sqsubseteq t_2)\}\}$
  für alle $\mathsf{F}{:}(t_1 \approx t_2)$ in Π, wobei $c = \mathit{sko}_{\mathrm{MLSSF}}(\mathsf{F}{:}(t_1 \approx t_2))$;

– $\{\{\bot\}\}$,
  falls

  1. $\mathsf{T}{:}\sigma{:}G, \mathsf{F}{:}\sigma{:}G \in \Pi$ für ein Atom G,
  2. $\mathsf{T}{:}(t \sqsubseteq \emptyset) \in \Pi$,
  3. $\mathsf{F}{:}(t \approx t) \in \Pi$,
  4. es für ein $k \in \mathbb{N}$ Atome $\mathsf{T}{:}(t_i \sqsubseteq t_{i+1})$ ($1 \leq i < k$) und $\mathsf{T}{:}(t_k \sqsubseteq t_1)$ in Π gibt. $\square$

3.8.4 Korrektheit, Vollständigkeit, Terminierung

Der im vorangegangenen Abschnitt definierte Kalkül $\mathcal{C}^{\mathrm{MLSS}}$ ist korrekt und vollständig (ein Beweis findet sich in (Hartmer, 1997)). Er hat die Korrektheits- und Vollständigkeitseigenschaften aus Definitionen 3.5.6 und 3.5.8.

Satz 3.8.4 *Der Kalkül $\mathcal{C}^{\mathrm{MLSS}}$ für* MLSS *ist korrekt und vollständig.*

Ohne weitere Einschränkungen ist der Kalkül $\mathcal{C}^{\mathrm{MLSS}}$ jedoch keine Entscheidungsprozedur. Das Regelschema für Ungleichungen ((R12) in Tabelle 3.11) führt zusätzliche Konstanten ein und das Schnittregel-Schema kann – in Verbindung mit Schema (R11) – benutzt werden, um neue Ungleichungen aus diesen zusätzlichen Konstanten zu konstruieren; die Wechselwirkung dieser Erweiterungsregel-Schemata kann zu unendlichen Ästen führen.

Glücklicherweise kann der Kalkül leicht in eine Entscheidungsprozedur verwandelt werden, indem man die die Vollständigkeit bewahrende Einschränkung hinzufügt, daß solche Ketten $c_1, c_2, \ldots$ niemals unendlich sein dürfen, bei denen die Konstante c_i zum Ast hinzugefügt wird, indem das Regelschema (R12) auf eine Ungleichung angewendet wird, die die Konstante c_{i-1} enthält; die maximale Länge solcher Ketten ist die Zahl der (Teil-)Terme in der Formelmenge deren Erfüllbarkeit überprüft werden soll.

Definition 3.8.5 Sei $T_1, \ldots, T_k$ eine Folge von Tableaus für eine Menge $\mathfrak{F}$ von MLSS-Formeln, die mit Hilfe der Erweiterungsregel $\mathcal{E}_{\mathrm{MLSS}}$ (Def. 3.8.3) konstruiert worden ist.

Der *Rang* $rk(s)$ eines Mengenterms s in der Folge von Tableaus ist wie folgt definiert: Wenn s in $\mathfrak{F}$ vorkommt oder durch eine Anwendung eines der Regelschemata (R1) und (R2) eingeführt wurde, dann sei $rk(s) = 0$; andernfalls, wenn also s eine Konstante ist, die dadurch eingeführt worden ist, daß das Regelschema (R12) auf eine Ungleichung $\mathsf{F}{:}(t_1 \approx t_2)$ angewendet worden ist, dann ist sein Rang

$$rk(s) = 1 + \max\{rk(t_1), rk(t_2)\} \ .$$

$\square$

Definition 3.8.6 Ein Tableau T für eine Menge $\mathfrak{F}$ von MLSS-Formeln ist *erschöpft*, wenn keine Erweiterungsregelanwendung auf T möglich ist, ohne daß

- eine Konstante eingeführt wird, deren Rang größer als die Zahl der (Teil-)Formeln in $\mathfrak{F}$ ist oder
- ausschließlich Formeln zu einem Ast B von T hinzugefügt werden, die schon auf B vorkommen.

Man beachte, das ein Tableau, das erschöpft ist (Def. 3.8.6), nicht immer auch voll expandiert ist (Def. 3.5.5).

Satz 3.8.7 *Es gibt genau dann ein erschöpftes, nicht geschlossenes $\mathcal{C}_{\mathrm{MLSS}}$-Tableau für eine Menge $\mathfrak{F}$ von MLSS-Formeln, wenn $\mathfrak{F}$ erfüllbar ist.*

Wenn also eine Folge von Tableaus für eine Menge $\mathfrak{F}$ von MLSS-Formeln in *fairer* Weise konstruiert wird (d. h., alle möglichen Regelanwendungen werde früher oder später ausgeführt), dann wird die Konstruktion nach endlich vielen Schritten mit einem Tableau terminieren, das (a) geschlossen ist – dann ist $\mathfrak{F}$ unerfüllbar – oder das (b) erschöpft ist – dann ist $\mathfrak{F}$ erfüllbar.

3.8.5 Einschränkungen des Suchraums

Auch wenn der Suchraum des Kalküls $\mathcal{C}_{\mathrm{MLSS}}$ mit der im vorangegangenen Abschnitt beschriebenen Einschränkung endlich ist, ist er doch sehr groß – wegen des Indeterminismus des Schnittregel-Schemas und weil die Zahl der Konstanten, die eingeführt werden können, exponentiell ist in der Größe der Formelmenge deren Erfüllbarkeit überprüft werden soll.

Glücklicherweise ist es möglich, die Anwendung des Schnittregel-Schemas stark einzuschränken, was gleichzeitig auch die Zahl der neu eingeführten Konstanten reduziert, weil eine Konstante c_k vom Rang k nur aus einer Ungleichung erzeugt werden kann, die eine Konstante c_{k-1} vom Rang $k-1$ enthält. nachdem das Schnittregel-Schema mit einem Atom als Schnittformel angewendet worden ist, das c_{k-1} enthält. Die Idee ist nun, zuerst alle Regelschemata außer dem Schnittregel-Schema so lange anzuwenden, bist keine weiteren Regelanwendungen mehr möglich sind, die neue Formeln zu Ästen hinzufügen, und dann eine *Realisation* eines offenen Astes zu konstruieren. Die Realisation eines Astes B approximiert eine Mengenstruktur, die die Formeln auf B erfüllt (falls B erfüllbar ist); sie erfüllt zumindest alle Atome der Form $\mathsf{T}{:}(t_1 \sqsubseteq t_2)$ auf B. Wenn die Realisation nicht auch alle anderen Atome auf B erfüllt, dann kann sie verwendet werden, um Schnittregelanwendungen zu finden, die (zumindest teilweise) nützlich und sinnvoll sind.

Das Umschalten zwischen der Erweiterung von Tableauästen und der Konstruktion möglicher Modelle und die Art und Weise, wie diese Modelle konstruiert werden, ähnelt der Methode, die in (Cantone, 1997) beschrieben ist.

Definition 3.8.8 Sei Σ eine MLSS-Signatur; und sei Π eine Menge von MLSS-Tableauformeln über Σ^*. Wir definieren folgende Bezeichnungen:

- $\mathcal{G}$ bezeichne die Menge aller Mengenterme über Σ, die in Π als (Teil-)Terme vorkommen;
- $\mathcal{V}$ bezeichne die Menge (a) aller Terme $t \in \mathcal{G}$, so daß $\mathsf{T}{:}(t \sqsubseteq s)$ ein Atom in Π ist, und (b) aller Konstanten aus Σ, die in Π vorkommen;
- $\mathcal{T}$ bezeichne die Menge aller Konstanten, die in Π vorkommen, aber nicht Element von $\mathcal{V}$ sind;
- $\sim$ bezeichne die Äquivalenzrelation auf $\mathcal{G} \cup \mathcal{T}$, die durch die Gleichungen in Π induziert wird;
- $\mathcal{T}'$ bezeichne die Menge aller $c \in \mathcal{T}$, so daß $c \not\sim s$ für alle $s \in \mathcal{G}$;
- $\mathcal{V}'$ bezeichne die Menge $(\mathcal{V} \cup \mathcal{T}) \setminus \mathcal{T}'$;
- u_c sei für jedes $c \in \mathcal{T}'$ ein Element der von-Neumann-Hierarchie $\mathfrak{V}$, das von allen $u_{c'}$ mit $c \neq c'$ verschieden ist. □

Man beachte, daß $\mathcal{T}'$ die Konstanten enthält, die durch Anwendungen des Regelschemas (R12) für Ungleichungen eingeführt worden sind und die *nicht* gleich anderen Termen sind (bzgl. der Gleichungen auf dem Ast). Die Interpretation dieser Konstanten muß von der Interpretation aller anderer Terme verschieden sein; während Terme in $\mathcal{V}'$ die gleiche Interpretation haben können.

Definition 3.8.9 Sei Σ eine MLSS-Signatur; sei Π eine Menge von Tableauformeln über Σ^*; und sei t ein Mengenterm Π. Dann ist die Menge $\mathcal{P}_\Pi(t)$ der *impliziten Vorgänger* von t wie folgt definiert:

1. $\mathcal{P}_\Pi(\emptyset) = \emptyset$;
2. $\mathcal{P}_\Pi(c) = \{s \in \mathcal{V} \cup \mathcal{T} \mid \mathsf{T}{:}(s \sqsubseteq c) \in \Pi\}$ für Konstanten c;
3. $\mathcal{P}_\Pi(t_1 \sqcup t_2) = \mathcal{P}_\Pi(t_1) \cup \mathcal{P}(t_2)$;
4. $\mathcal{P}_\Pi(t_1 \sqcap t_2) = \mathcal{P}_\Pi(t_1) \cap \mathcal{P}_\Pi(t_2)$;
5. $\mathcal{P}_\Pi(t_1 \setminus t_2) = \mathcal{P}_\Pi(t_1) \setminus \mathcal{P}_\Pi(t_2)$; und
6. $\mathcal{P}_\Pi(\{t_1, \ldots, t_n\}_n) = \{s \in \mathcal{V} \cup \mathcal{T} \mid \mathsf{T}{:}(s \sqsubseteq \{t_1, \ldots, t_n\}_n) \in \Pi\} \cup \{t_1, \ldots, t_n\}$.

$\square$

Die Mengen impliziter Vorgänger können benutzt werden, um implizite Enthaltenseins-Zyklen aufzudecken. Wenn beispielsweise $s \in \mathcal{P}_\Pi(t), t \in \mathcal{P}_\Pi(s)$ für Terme s, t gilt, dann kann der Ast geschlossen werden, und es ist nicht notwendig, die Erweiterungsregel (und besonders das Schnittregel-Schema) anzuwenden, um den Zyklus explizit zu machen. Also kann der Kalkül mit Hilfe der Vorgänger-Mengen stärker gemacht werden, indem ein weiteres Regelschema zum Abschluß von Ästen hinzugefügt wird:

Definition 3.8.10 Der Kalkül $\mathcal{C}'_{\mathrm{MLSS}}$ stimme mit dem Kalkül $\mathcal{C}_{\mathrm{MLSS}}$ bis auf seine Erweiterungsregel $\mathcal{E}'_{\mathrm{MLSS}}$ überein, die wie folgt definiert ist:

Für alle MLSS-Signaturen Σ und alle Prämissen $\Pi \subset \mathit{TabForm}_{\mathrm{MLSS}}(\Sigma^*)$ bestehe die Menge $\mathcal{E}'_{\mathrm{MLSS}}(\Sigma)(\Pi)$ aus

– den Konklusionen in $\mathcal{E}_{\mathrm{MLSS}}(\Sigma)(\Pi)$ (Def. 3.8.3) und

– der Konklusion $\{\{\bot\}\}$, falls die Mengen der impliziten Vorgänger der Terme in Π einen Zyklus enthalten, d. h., falls es Mengenterme $t_1, \ldots, t_n$ gibt, die als (Teil-) Terme in Π auftreten, so daß $t_1 \in \mathcal{P}_\Pi(t_2)$, …, $t_{n-1} \in \mathcal{P}(t_n)$, $t_n \in \mathcal{P}_\Pi(t_1)$. $\square$

Satz 3.8.11 *Der Kalkül $\mathcal{C}'_{\mathrm{MLSS}}$ für* MLSS *ist korrekt und vollständig.*

Die Menge $\mathcal{P}_\Pi(t)$ impliziter Vorgänger enthält diejenigen Terme, die Elemente der Menge repräsentieren, die von t repräsentiert wird, und deren Enthaltensein aus den Atomen der Form $\mathsf{T}{:}(s \sqsubseteq a)$ in Π (wobei a eine Konstante ist) und durch Anwendung der Definitionen der Mengenoperationen abgeleitet werden kann, Die *Realisation* einer Menge von Tableauformeln geht darüber noch hinaus: sie ist eine partielle Definition einer Mengenstruktur, in der verschiedene Terme durch die gleiche Menge interpretiert werden können.

Definition 3.8.12 Sei Σ eine MLSS-Signatur; und sei Π eine Menge von Tableauformeln über Σ^* mit $\bot \notin \Pi$. Die *Realisation* $\mathcal{R}_\Pi(t)$ eines Terms t, der in Π vorkommt, ist wie folgt definiert:

1. $\mathcal{R}_\Pi(t) = \emptyset$, falls $t = \emptyset$,
2. $\mathcal{R}_\Pi(t) = \{\mathcal{R}_\Pi(s) \mid s \in \mathcal{P}_\Pi(t)\} \cup \{u_t\}$, falls $t \in T'$,[3] und
3. $\mathcal{R}_\Pi(t) = \{\mathcal{R}_\Pi(s) \mid s \in \mathcal{P}_\Pi(t)\}$ sonst. □

Die Realisation ist effektiv berechenbar und kann benutzt werden, um die Anwendungen des Schnittregel-Schemas einzuschränken: vorausgesetzt, ein Ast B ist erschöpft bezüglich aller anderer Erweiterungsregel-Schemata, muß das Schnittregel-Schema nur auf solche Terme angewendet werden, die in Atomen vorkommen, die *nicht* von der Realisation $\mathcal{R}_\Pi$ erfüllt werden, wobei $\Pi = Form(B)$ (wenn es kein solches Atom gibt, dann ist der Ast erfüllbar, und die Suche nach einem Tableaubeweis terminiert).

Wenn beispielsweise $\mathsf{F}{:}(t_1 \sqsubseteq t_2)$ ein Atom in Π ist, aber $\mathcal{R}_\Pi(t_1) \in \mathcal{R}_\Pi(t_2)$, dann muß es einen Term s geben, so daß (a) $\mathcal{R}_\Pi(s) = \mathcal{R}_\Pi(t_1)$ (d. h., die Realisation von s ist dieselbe wie die von t_1) und (b) s ist ein implizites Element von t_2 (d. h., $s \in \mathcal{P}_\Pi(t_2)$) – aber dieses implizite Enthaltensein ist (noch) nicht explizit dargestellt (es gibt kein Atom $\mathsf{T}{:}(s \sqsubseteq t_2)$ in Π). In diesem Fall, wird das Schnittregel-Schema mit dem Atom $\mathsf{T}{:}(s \sqsubseteq t_2)$ als Schnittformel angewendet.

Der folgende Satz sagt aus, daß die Vollständigkeit des Kalküls $\mathcal{C}'_{\mathrm{MLSS}}$ erhalten bleibt, wenn Realisationen verwendet werden, um die Anwendung des Schnittregel-Schemas einzuschränken (ein Beweis findet sich in (Hartmer, 1997)). Man beachte, daß der Kalkül nicht abgeschwächt wird, sondern daß die Einschränkung eine Technik ist, um Beweisprozeduren, die auf $\mathcal{C}'_{\mathrm{MLSS}}$ basieren, effizienter zu machen.

Definition 3.8.13 Wenn $\mathfrak{F}$ eine unerfüllbar Menge von MLSS-Formeln ist, dann gibt es einen Tableaubeweis $T_1, \ldots, T_n$ für $\mathfrak{F}$, der unter Beachtung folgender Einschränkungen konstruiert ist:

Das Schnittregel-Schema wird nur dann angewendet, um ein Tableau T_{i+1} aus T_i abzuleiten, wenn der Ast B von T_i, der erweitert wird,

[3] Man muß sicherstellen, daß die u_c für alle Terme t von $\mathcal{R}_\Pi(t)$ verschieden sind; es ist immer möglich, solche u_c zu wählen.

1. nicht geschlossen ist und
2. mit keinem anderen Erweiterungsregelschema eine Formeln zu B hinzuzufügt werden kann, die nicht schon auf B ist;

und die Schnittformel $\mathsf{T}{:}(s \approx t)$, die für die Erweiterung verwendet wird, eine der folgenden Bedingungen erfüllt, wobei $\Pi = Form(B)$:

1. (a) $\mathsf{T}{:}(t \approx t') \in \Pi$,
   (b) $\mathcal{R}_\Pi(t) \neq \mathcal{R}_\Pi(t')$, und
   (c) i. $s \in \mathcal{P}_\Pi(t)$, $s \notin \mathcal{P}_\Pi(t')$, und $\mathsf{F}{:}(s \sqsubseteq t) \notin \Pi$, oder
   ii. $s \in \mathcal{P}_\Pi(t')$, $s \notin \mathcal{P}_\Pi(t)$, und $\mathsf{T}{:}(s \sqsubseteq t') \notin \Pi$,
2. (a) $\mathsf{F}{:}(t \approx t')$, $\mathsf{F}{:}(c \sqsubseteq t)$, und $\mathsf{T}{:}(c \sqsubseteq t') \in \Pi$ (für eine Konstante c),
   (b) $\mathcal{R}_\Pi(t) = \mathcal{R}_\Pi(t')$, $\mathcal{R}_\Pi(s) = \mathcal{R}_\Pi(c)$, und
   (c) $s \in \mathcal{P}_\Pi(t)$, $s \notin \mathcal{P}_\Pi(t')$, und $\mathsf{T}{:}(s \sqsubseteq t) \notin \Pi$,
3. (a) $\mathsf{F}{:}(t' \sqsubseteq t) \in \Pi$,
   (b) $\mathcal{R}_\Pi(t') \in \mathcal{R}_\Pi(t)$, $\mathcal{R}_\Pi(s) = \mathcal{R}_\Pi(t')$, und
   (c) $s \in \mathcal{P}_\Pi(t)$, und $\mathsf{T}{:}(s \sqsubseteq t) \notin \Pi$. □

3.8.6 Ein Vergleich mit Cantones Kalkül

Der im vorangegangenen Abschnitt beschriebene Kalkül $\mathcal{C}_{\mathrm{MLSS}}$ für MLSS hat gewisse Gemeinsamkeiten mit dem von Cantone (1997) beschriebenen. Ein wesentlicher Unterschied ist, daß Cantones Kalkül auf *normalisierte* Atome beschränkt ist, d. h., auf Atome, die keine komplexen Mengenterme enthalten.

Definition 3.8.14 Eine atomare MLLS-Tableauformel ϕ ist genau dann *normalisiert*, wenn sie von der Form

– $\mathsf{S}{:}(a \sqsubseteq b)$,
– $\mathsf{S}{:}(a \approx b)$,
– $\mathsf{T}{:}(a \approx b \sqcup c)$,
– $\mathsf{T}{:}(a \approx b \sqcap c)$,
– $\mathsf{T}{:}(a \approx b \setminus c)$ oder
– $\mathsf{T}{:}(a \approx \{b_1, \ldots, b_n\}_n)$ $(n \geq 1)$

ist, wobei a, b, c und $b_1, \ldots, b_n$ Konstanten sind. □

Es gibt eine erfüllbarkeitserhaltende Transformation beliebiger endlicher Mengen Γ von MLSS-Tableauformeln in Mengen normalisierter Atome, die darauf beruht, neue Konstanten als „Abkürzungen" für komplexe Terme einzuführen. Beispielsweise wird

$$\mathsf{T}{:}(a \sqsubseteq (b \sqcap b'))$$

durch

$$\mathsf{T}{:}(c \approx (b \sqcap b')) \text{ und } \mathsf{T}{:}(a \sqsubseteq c)$$

ersetzt, wobei c eine neue Konstante ist. Der Overhead für die Ausführung dieser Transformation ist vernachlässigbar, weil ihre Berechnungskomplexität polynomiell in der Größe der zu transformierenden Menge ist. Allerdings führt die Einführung neuer Konstanten zu einem sehr viel größeren Suchraum – um so mehr, als diese neuen Konstanten in Gleichungen auftreten.

Die Regelschemata (R7), (R3), (R4) und (R10) unseres Kalküls sind – in Verbindung mit den Regelschemata (EQ1) und (EQ2) – Erweiterungen für *komplexe* Mengenterme der entsprechenden Schemata in Cantones Kalkül. Beispielsweise entspricht unser Regelschema (R3), das erlaubt,

$$\mathsf{T}{:}(a \sqsubseteq b) \text{ und } \mathsf{T}{:}(a \sqsubseteq b') \quad \text{aus} \quad \mathsf{T}{:}(a \sqsubseteq (b \sqcap b'))$$

abzuleiten, Cantones Regelschema, das erlaubt,

$$\mathsf{T}{:}(a \sqsubseteq b) \text{ und } \mathsf{T}{:}(a \sqsubseteq b') \quad \text{aus} \quad \mathsf{T}{:}(c \approx (b \sqcap b')) \text{ und } \mathsf{T}{:}(a \sqsubseteq c)$$

abzuleiten (für alle a, b, c).

Es gibt in Cantones Kalkül kein Regelschema, das unseren Schemata für Atome, die Nichtenthaltensein ausdrücken ((R5), (R8) und (R9)), entspricht. Betrachten wir die drei Atome

$$\phi = \mathsf{F}{:}(c \sqsubseteq (b_1 \sqcup b_2) \setminus b_3), \quad \psi_1 = \mathsf{T}{:}(c \sqsubseteq b_1), \quad \psi_2 = \mathsf{F}{:}(c \sqsubseteq b_3) \ ,$$

deren Konjunktion unerfüllbar ist. Um einen Ast, der diese Atome enthält, zu schließen, werden unsere Regelschemata (R9) und (R5) angewendet, um das Atom ϕ aufzuspalten und abzuleiten, daß eines der Komplemente $\overline{\psi_1}$ und $\overline{\psi_2}$ wahr sein muß, was erlaubt, die zwei entstehenden Teiläste zu schließen (siehe das Tableau[4] in Abbildung 3.3 (a)). Da in Cantones Kalkül kein Regelschemata zum Aufspalten von ϕ existiert, müssen statt dessen Schemata für Atome, die Enthaltensein ausdrücken, verwendet werden, um $\overline{\phi}$ aus ψ_1 und ψ_2 abzuleiten. Zuerst muß ϕ normalisiert werden; das Ergebnis sind die Atome

$$\mathsf{F}{:}(c \sqsubseteq d_1), \quad \mathsf{T}{:}(d_1 \approx d_2 \setminus b_3), \quad \mathsf{T}{:}(d_2 \approx b_1 \sqcup b_2) \ ,$$

[4] An den i-te Knoten in diesem Tableau ist die Bezeichnung $[i; j; (\mathrm{R})]$ angeheftet, was andeutet, daß seine Formel durch Anwendung des Regelschemas (R) von $\mathcal{C}_{\mathrm{MLSSF}}$ auf eine Prämisse erzeugt wurde, die aus der Formel des j-ten Knotens besteht.

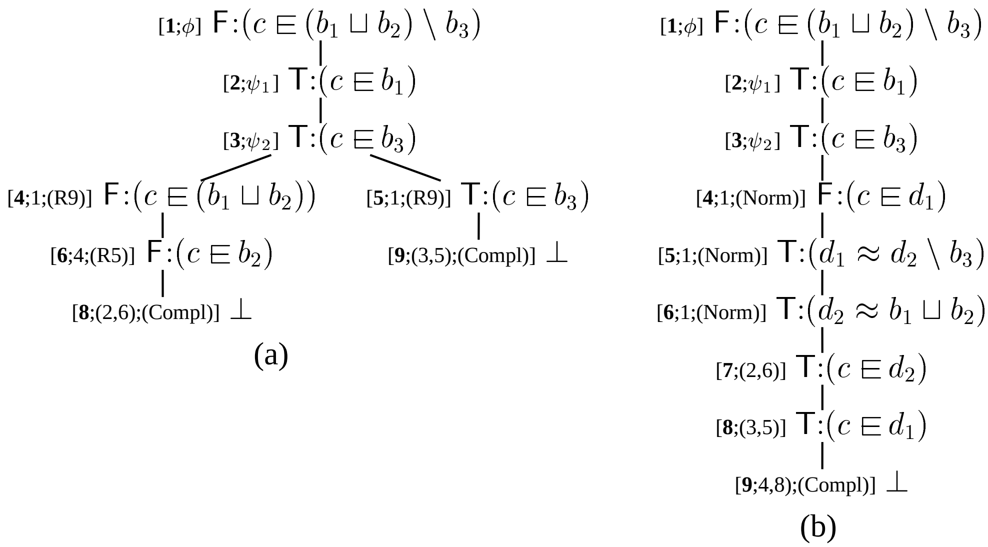


Abbildung 3.3: Konstruktion eines geschlossenen Teiltableaus mit (a) dem Kalkül $\mathcal{C}_{\mathrm{MLSS}}$ und (b) Cantones Kalkül.

wobei d_1 den Term $(b_1 \sqcup b_2) \setminus b_3$ abkürzt und d_2 den Term $b_1 \sqcup b_2$. Dann können mit zwei Regelanwendungen T:$(c \sqsubseteq d_2)$ und T:$(c \sqsubseteq d_1)$ abgeleitet werden. Das letztere dieser beiden Atome kann verwendet werden, um den Ast abzuschließen; es entspricht der Normalisierung des Atoms $\overline{\phi}$ (siehe das Tableau in Abbildung 3.3 (b)).

Die Notwendigkeit (und Möglichkeit), komplexere Terme aus einfacheren zu konstruieren, führt zu einem größeren Suchraum. Unser Kalkül, der komplexe Terme in einfacherer aufspaltet, ist zielgerichteter.

3.8.7 Ein gutartiger Kalkül für MLSSF

Um den im vorangegangenen Abschnitt beschriebenen Kalkül $\mathcal{C}_{\mathrm{MLSS}}$ für MLSS zu einem Kalkül $\mathcal{C}_{\mathrm{MLSSF}}$ für das größere Fragment MLSSF zu erweitern, genügt es,

- die Vorbedingung für die Anwendbarkeit der Gleichheitsregel-Schemata abzuschwächen (die neuen Schemata (EQ1') und (EQ2') sind in Tabelle 3.12 enthalten) und

- ein Schnittregel-Schema hinzuzufügen, daß Gleichungen als Schnittformeln verwendet (das Schema (Cut') in Tabelle 3.12).

Die neuen Regelschemata müssen nur auf funktionale Mengenterme angewendet werden. Nicht-funktionale Terme (auch wenn sie nicht reine Mengenterme sind), können die Erweiterungsregel-Schemata des Kalküls $\mathcal{C}_{\mathrm{MLSS}}$ handhaben. Die Erweiterungsregel $\mathcal{E}_{\mathrm{MLSSF}}$ von $\mathcal{C}_{\mathrm{MLSSF}}$ ist unten formal definiert.

$$\frac{\begin{matrix}\mathsf{T}{:}(s \approx t) \\ \phi[s]\end{matrix}}{\phi[t]} \qquad \frac{\begin{matrix}\mathsf{T}{:}(t \approx s) \\ \phi[s]\end{matrix}}{\phi[t]}$$

wobei das Vorkommen von s in ϕ innerhalb eines funktionalen Terms ist

(EQ1') (EQ2')

$$\frac{}{\mathsf{T}{:}(t_1 \approx t_2) \mid \mathsf{F}{:}(t_1 \approx t_2)}$$

wobei t_1, t_2 Terme auf dem Ast sind, von denen wenigstens einer funktional ist

(Cut')

Tabelle 3.12: Zusätzliche Erweiterungsregel-Schemata für MLSSF.

Definition 3.8.15 Sei Σ eine MLSSF-Signatur.

Für alle Prämissen $\Pi \subset \mathit{TabForm}_{\mathrm{MLSSF}}(\Sigma^*)$ sei $\mathcal{E}_{\mathrm{MLSSF}}(\Sigma^*)(\Pi)$ die kleinste Menge, die die folgenden Konklusionen enthält:

- $\{\{\alpha_1, \ldots, \alpha_n\}\}$
  für alle $\alpha \in \Pi$ (Tabellen 3.1 und 3.10);
- $\{\{\beta_1\}, \ldots, \{\beta_n\}\}$
  für alle $\beta \in \Pi$ (Tabellen 3.1 und 3.10);
- $\{\{\phi[t]\}\}$
  für alle (a) $\phi[s]$ und (b) $\mathsf{T}{:}(s \approx t)$ oder $\mathsf{T}{:}(t \approx s)$ in Π, so daß s ein echter Teilterm eines funktionalen Terms in $\phi[s]$ ist, wobei $\phi[t]$ aus $\phi[s]$ entsteht, indem ein Vorkommen von s durch t ersetzt wird;
- $\{\{\mathsf{F}{:}(s_1 \approx s_2)\}\}$
  für alle $\mathsf{T}{:}(s_1 \sqsubseteq t)$ und $\mathsf{F}{:}(s_2 \sqsubseteq t)$ in Π;
- $\{\{\mathsf{T}{:}(s \sqsubseteq t)\}, \{\mathsf{F}{:}(s \sqsubseteq t)\}\}$
  für alle Mengenterme s und t, so daß

  1. s oder $\{\ldots, s, \ldots\}_n$ und
  2. t oder $\{\ldots, t, \ldots\}_n$

  als Top-level-Terme in Atomen in Π auftreten;
- $\{\{\mathsf{T}{:}(t_1 \approx t_2)\}, \{\mathsf{F}{:}(t_1 \approx t_2)\}\}$
  für alle Mengenterme t_1 und t_2, die in Π vorkommen und von denen wenigstens einer ein funktionaler Term ist;
- $\{\{\mathsf{T}{:}(c \sqsubseteq t_1), \mathsf{F}{:}(c \sqsubseteq t_2)\}, \{\mathsf{F}{:}(c \sqsubseteq t_1), \mathsf{T}{:}(c \sqsubseteq t_2)\}\}$
  für alle $\mathsf{F}{:}(t_1 \approx t_2)$ in Π, wobei $c = \mathit{sko}_{\mathrm{MLSSF}}(\mathsf{F}{:}(t_1 \approx t_2))$
- $\{\{\bot\}\}$,
  falls

  1. $\mathsf{T}{:}\sigma{:}G, \mathsf{F}{:}\sigma{:}G \in \Pi$ für ein Atom G,
  2. $\mathsf{T}{:}(t \sqsubseteq \emptyset) \in \Pi$,

3. $\mathsf{F}{:}(t \approx t) \in \Pi$,
4. es für ein $k \in \mathbb{N}$ Atome $\mathsf{T}{:}(t_i \sqsubseteq t_{i+1})$ $(1 \leq i < k)$ und $\mathsf{T}{:}(t_k \sqsubseteq t_1)$ in Π gibt. □

Der Kalkül $\mathcal{C}_{\mathrm{MLSSF}}$ für MLSSF ist korrekt und vollständig (ein Beweis findet sich in (Hartmer, 1997)); er ist jedoch keine Entscheidungsprozedur.

Satz 3.8.16 *Der Kalkül $\mathcal{C}_{\mathrm{MLSSF}}$ für MLSSF ist korrekt und vollständig.*

3.8.8 Einschränkung des Schnittregel-Schemas mit Hilfe von Starrer *E*-Unifikation

Die im vorangegangenen Abschnitt eingeführten zusätzlichen Regelschemata des Kalküls $\mathcal{C}_{\mathrm{MLSSF}}$ sind hochgradig nicht-deterministisch. In diesem Abschnitt beschreiben wir ein Erweiterungsregel-Schema für MLSSF, das sehr viel zielgerichteter ist, und zu einem kleineren Suchraum führt. Es basiert auf dem Konzept der *starren E-Unifikation*.

Definition 3.8.17 Sei Σ eine MLSSF-Signatur.

Eine *Gleichung mit starren Variablen* ist eine Formel mit starren Variablen über Σ^*_{fv} von der Form $t \approx t'$.

Ein *starres E-Unifikationsproblem* $\langle E, s, t\rangle$ besteht aus einer endlichen Menge E starrer Gleichungen und aus Termen s und t über Σ^*_{fv}.

Eine Substitution $\sigma \in \mathit{Subst}(\Sigma^*_{\mathrm{fv}})$ ist genau dann eine *Lösung* des Problems $\langle E, s, t\rangle$, wenn $s\sigma$ und $t\sigma$ gleich sind in der freien Algebra, die durch $E\sigma$ definiert wird, wobei die freien Variablen in $E\sigma$ als Konstanten behandelt werden. □

Das Problem zu entscheiden, ob ein gegebenes starres E-Unifikationsproblem eine Lösung besitzt, ist entscheidbar (es ist NP-vollständig). Im allgemeinen ist die Zahl der Lösungen unendlich. Ein Überblick über Methoden zur Lösung starrer E-Unifikationsprobleme findet sich in (Beckert, 1998b).

Die Idee ist nun, starre E-Unifikation für die Behandlung des funktionalen Anteils von MLSSF zu verwenden, und die MLSS-Erweiterungsregel für die Behandlung des nicht-funktionalen (also mengentheoretischen) Anteils. Das zusätzliche Erweiterungsregel-Schema, das wir im folgenden beschreiben, ist das Bindeglied zwischen diesen beiden Teilen.

Betrachten wir z. B. einen Ast B, der die beiden Atome

$$\mathsf{T}{:}(f(a) \approx b) \ \text{ und } \ \mathsf{T}{:}(g(f(a \sqcap (b \sqcup a))) \sqsubseteq g(b))$$

enthält. Der Ast B ist unerfüllbar, weil $a \cap (b \cup a) = a$ und also

$$g(f(a \cap (b \cup a))) = g(f(a)) = g(b) \;\; ;$$

das impliziert $g(b) \in g(b)$, was ein Enthaltenseins-Zyklus ist. Um den Ast B mit Hilfe der Erweiterungsregel von $\mathcal{C}_{\text{MLSSF}}$ um ein geschlossenes Teiltableau zu erweitern, muß man zunächst die mengentheoretischen Gleichungen finden, die bewiesen werden müssen (eine solche Gleichung wird bewiesen, indem man sie als Schnittformel benutzt; nachdem der Ast, der das Komplement der Gleichung enthält, abgeschlossen ist, steht sie auf dem verbliebenen offenen Ast zur Verfügung). Im obigen Beispiel ist $a \cap (b \cup a) = a$ die Gleichung, die man beweisen muß.

Die Frage, welche mengentheoretischen Gleichungen bewiesen werden müssen, damit ein Ast B geschlossen werden kann, wird wie folgt in starre E-Unifikationsprobleme übersetzt: für jedes Paar s, t von Termen, die, wären sie gleich, den Abschluß des Astes gestatten würden (z. B. die Terme s, t, wenn $\mathsf{F}{:}(s \approx t)$ auf B ist), wird ein starres E-Unifikationsproblem erzeugt. In s und t werden alle (maximalen) nicht-funktionalen Teilterme durch starre Variablen ersetzt; die daraus resultierenden Terme s^{rv} und t^{rv} und die Gleichungen des Astes bilden ein starres E-Unifikationsproblem. Jede Lösung des Problem entspricht einer Gleichung zwischen nicht-funktionalen Teiltermen, die, wenn man sie beweisen kann, es erlaubt, den Ast abzuschließen. Die den Lösungen entsprechenden Ungleichungen werden als Konklusion verwendet, d. h., sie können (disjunktiv verknüpft) zum Ast hinzugefügt werden.

Definition 3.8.18 Sei Λ eine Menge atomare MLSSF-Tableauformel. Die Menge Λ^{rv} werde aus Λ konstruiert, indem alle nicht-funktionalen (Teil-)Terme t in Λ durch eine starre Variable X_t ersetzt werden.

Sei die Substitution τ_Λ wie folgt definiert: $\tau(X_t) = t$ für alle Terme t, die in Λ vorkommen und ersetzt worden sind (d. h., τ_Λ ist die inverse Operation der Transformation, die Λ^{rv} aus Λ konstruiert: $\Lambda^{\text{rv}}\tau = \Lambda$). □

Beispiel 3.8.19 Sei

$$\Lambda = \{\mathsf{T}{:}((a \sqcap c) \sqcup b \approx c),\ \mathsf{T}{:}[f(c) \sqsubseteq g(a \sqcap c, f(d \setminus e))]\} \;\; ,$$

dann ist

$$\Lambda^{\text{rv}} = \{\mathsf{T}{:}(X_1 \approx X_2),\ \mathsf{T}{:}[f(X_2) \sqsubseteq g(X_3, f(X_4))]\}$$

das Ergebnis der Transformation. □

Definition 3.8.20 Sei Σ eine MLSSF-Signatur; sei $\Pi \subset \mathit{TabForm}_{\text{MLSSF}}(\Sigma^*)$ eine Prämisse; sei Λ_Π die Menge aller Atome in Π, die von der Form $\mathsf{T}{:}(t_1 \approx t_2)$, $\mathsf{T}{:}(t_1 \sqsubseteq t_2)$ oder $\mathsf{F}{:}(t_1 \sqsubseteq t_2)$ sind; und sei E_Π^{rv} die Menge aller Gleichungen mit starren Variablen in Λ_Π^{rv}.

Weiterhin sei $\mu = \{x_1 \leftarrow r_1, \dots, x_n \leftarrow r_n\}$ $(n \geq 1)$ eine simultane Lösung[5]

1. starrer E-Unifikationsprobleme $\langle E_\Pi^{\mathrm{rv}}, s_1, t_1 \rangle$ und $\langle E_\Pi^{\mathrm{rv}}, s_2, t_2 \rangle$, so daß $\mathsf{T}{:}(s_1 \sqsubseteq t_1)$ und $\mathsf{F}{:}(s_2 \sqsubseteq t_2)$ in $\Lambda_\Pi^{\mathrm{rv}}$ vorkommen, oder
2. starrer E-Unifikationsprobleme $\langle E_\Pi^{\mathrm{rv}}, t_1, t'_1 \rangle$, …, $\langle E_\Pi^{\mathrm{rv}}, t_n, t'_n \rangle$ $(n \geq 1)$, so daß Atome $\mathsf{T}{:}(t_1 \sqsubseteq t'_2, \dots, t_{n-1} \sqsubseteq t'_n)$ und $\mathsf{T}{:}(t_n \sqsubseteq t'_1)$ in $\Lambda_\Pi^{\mathrm{rv}}$ einen potentiellen Enthaltenseins-Zyklus bilden.

Dann ist die Konklusion

$$\{\{\mathsf{F}{:}(\tau_{\Lambda_\Pi}(X_1) \approx r_1\tau_{\Lambda_B})\},\ \dots, \{\mathsf{F}{:}(\tau_{\Lambda_B}(X_n) \approx r_n\tau_{\Lambda_B})\}$$

eine *EU-Konklusion* von Π. □

Mit Hilfe des Begriffs der EU-Konklusion kann die verbesserte Erweiterungsregel $\mathcal{E}_{\mathrm{MLSSF}}^{\mathrm{EU}}$ des Kalküls $\mathcal{C}_{\mathrm{MLSSF}}^{\mathrm{EU}}$ wie folgt definiert werden:

Definition 3.8.21 Sei Σ eine MLSSF-Signatur.

Für alle Prämissen $\Pi \subset \mathit{TabForm}_{\mathrm{MLSSF}}(\Sigma^*)$ bestehe die Menge $\mathcal{E}_{\mathrm{MLSSF}}^{\mathrm{EU}}(\Sigma^*)(\Pi)$ aus den Konklusionen in $\mathcal{E}_{\mathrm{MLSS}}(\Sigma^*)(\Pi)$ (Def. 3.8.3) und (zusätzlich) allen EU-Konklusionen von Π. □

Beispiel 3.8.22 Wir setzen das Beispiel vom Anfang dieses Abschnittes fort und wenden die Erweiterungsregel des neuen Kalküls $\mathcal{C}_{\mathrm{MLSSF}}^{\mathrm{EU}}$ an, um ein geschlossenes Teiltableau unter einem Ast zu konstruieren, der die Atome

$$\mathsf{T}{:}(f(a) \approx b) \ \text{ und } \ \mathsf{T}{:}(g(f(a \sqcap (b \sqcup a))) \sqsubseteq g(b))$$

enthält. Das einzige E-Unifikationsproblem, das aus diesen Atomen extrahiert werden kann, ist

$$\langle \{f(X_a) \approx X_b\}, g(f(X_{a\sqcap(b\sqcup a)})), g(X_b) \rangle \ .$$

Eine allgemeinste Lösung dieses Problems ist die Substitution $\{X_a \mapsto X_{a\sqcap(b\sqcup a)}\}$. Daher erlaubt das neue Erweiterungsregel-Schema von $\mathcal{C}_{\mathrm{MLSSF}}^{\mathrm{EU}}$, den Ast mit dem Atom $\mathsf{F}{:}(a \approx a \sqcap (b \sqcup a))$ zu erweitern; er kann dann zu einem geschlossenen Teiltableau erweitert werden (siehe Abbildung 3.4). □

[5] Die Substitution μ muß eine simultane Lösung mehrerer E-Unifikationsprobleme sein, d. h., ein *simultanes* starres E-Unifikationsproblem muß gelöst werden. Im allgemeinen ist die simultane starre E-Unifikation unentscheidbar. Aber die simultanen Probleme, die hier gelöst werden müssen, gehören zu einer entscheidbaren Teilklasse, nämlich der Klasse simultaner starrer E-Unifikationsprobleme von der Form $\{\langle E, s_1, t_1 \rangle, \dots, \langle E, s_n, t_n \rangle\}$, bei denen die Gleichungsmengen identisch sind. In diesem Fall ist jede Substitution, die (a) eine Lösung des nicht-simultanen Problems $\langle E, f(s_1, \dots, s_n), f(t_1, \dots, t_n) \rangle$ ist und (b) keine Variable mit einem Term instantiiert, der f enthält, eine Lösung des ursprünglichen Problems (das Funktionssymbol f darf nicht im ursprünglichen Problem vorkommen).

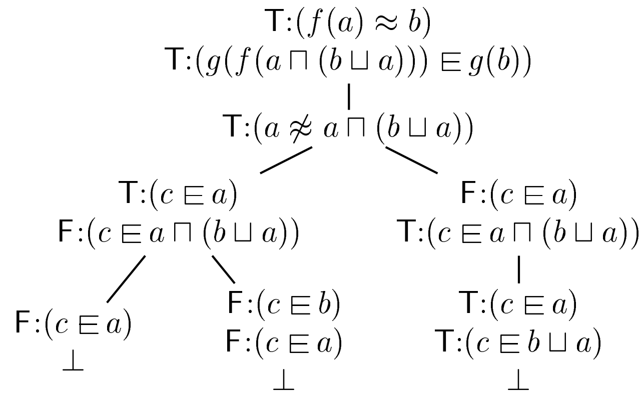


Abbildung 3.4: Ein geschlossenes Teiltableau, das mit der Erweiterungsregel des Kalküls $\mathcal{C}^{\mathrm{EU}}_{\mathrm{MLSSF}}$ konstruiert ist (3.8.22).

Das neue Erweiterungsregel-Schema überschneidet sich teilweise mit anderen Regelschemata. Es erlaubt beispielsweise, $\mathsf{F}{:}(s_1 \approx s_2)$ aus $\mathsf{T}{:}(s_1 \sqsubseteq t)$ und $\mathsf{F}{:}(s_2 \sqsubseteq t)$ abzuleiten, falls s_1 und s_2 nicht-funktionale Terme sind. Das gleiche ist auch durch Anwendung des Regelschemas (R11) möglich.

Es ist nicht notwendig, starre E-Unifikationsprobleme zu berücksichtigen, die aus Ungleichungen $\mathsf{F}{:}(s \approx t)$ in L^{rv}_{Π} konstruiert werden können. Weil nämlich, nachdem das Regelschema (R11) angewendet worden ist, der Ast entweder die Atome $\mathsf{T}{:}(c \sqsubseteq s)$, $\mathsf{F}{:}(c \sqsubseteq t)$ oder die Atome $\mathsf{F}{:}(c \sqsubseteq s)$, $\mathsf{T}{:}(c \sqsubseteq t)$ enthält.

Die Erweiterungsregel von $\mathcal{C}^{\mathrm{EU}}_{\mathrm{MLSSF}}$ ist korrekt, und das neue Schema hilft, den Suchraum zu verkleinern. Es besteht die Vermutung, daß $\mathcal{C}^{\mathrm{EU}}_{\mathrm{MLSSF}}$ auch vollständig ist, daß also die auf starrer E-Unifikation basierende Regelschema die Schemata (EQ1'), (EQ2'), und (Cut') es Kalküls $\mathcal{C}_{\mathrm{MLSSF}}$ tatsächlich ersetzen kann, dies ist aber bis jetzt unbewiesen.

Satz 3.8.23 *Der Kalkül $\mathcal{C}^{\mathrm{EU}}_{\mathrm{MLSSF}}$ für MLSSF ist korrekt.*

Vermutung 3.8.24 *Der Kalkül $\mathcal{C}^{\mathrm{EU}}_{\mathrm{MLSSF}}$ für MLSSF ist vollständig.*

Beispiel 3.8.25 Wir beweisen, daß die Formel

$$G \;=\; [a \sqsubseteq [(f(a) \setminus f(a \sqcup (b \sqcap a))) \sqcup d \sqcup e] \;\wedge\; e \sqcup b \sqsubseteq a] \;\rightarrow\; a \sqsubseteq d$$

eine MLSSF-Tautologie ist; sie enthält das freie Funktionssymbol f. Intuitiv ist wie folgt einzusehen, daß die Formel G eine Tautologie ist: Wir machen die Annahme, daß a ein Element von wenigstens einer der drei Mengen $u = f(a) \setminus f(a \cup (b \cap a))$, d und e ist und daß $e \cup b$ ein Element von a ist. Nun kann a nicht in der Menge u enthalten sein, weil $a = (a \cup (b \cap a))$ und darum u die leere Menge ist (für alle Interpretationen von f); die Menge e kann a nicht enthalten, denn andernfalls gäbe es einen Enthaltenseins-Zyklus $a \in (e \cup b) \in a$. Also muß a in der Menge d enthalten sein.

Abbildung 3.5 zeigt ein geschlossenes Tableau für $\neg G$.[6] Die Unerfüllbarkeit von $\neg G$ impliziert die Allgemeingültigkeit von G.

Formel (11) ist aus den Formeln (9) und (10) durch Anwendung des EU-Regelschemas abgeleitet. Die Substitution $\{x_a \leftarrow x_{a \sqcup (b \sqcap a)}\}$ ist eine Lösung der starren E-Unifikationsprobleme $\langle \emptyset, X_a, X_a \rangle$ und $\langle \emptyset, \langle f(X_a), f(X_{a \sqcup (b \sqcap a)}) \rangle \rangle$, die aus 9 und 10 entstehen. Dementsprechend ist die Ungleichung $\mathsf{F}{:}(a \approx a \sqcup (b \sqcap a))$ zum Ast hinzugefügt.

Bei der Anwendung des Regelschemas (R12) auf die Formel (11), durch die die Formeln (12)–(15) zum Tableau hinzugefügt werden, wird die Skolem-Konstante

$$c_1 = sko_{\mathrm{MLSSF}}(\mathsf{F}{:}(a \approx a \sqcup (b \sqcap a)))$$

eingeführt.

Der Ast, der in dem Knoten (30) endet, ist durch den Enthaltenseins-Zyklus $e \sqcup b \sqsubseteq a$ und $a \sqsubseteq e \sqcup b$ (Formeln (6) und (28)) geschlossen. Alle anderen Äste sind durch Paare komplementärer Atome geschlossen.

Wenn das Regelschema zum Abschluß von Ästen verwendet wird, das auf der Berechnung von Mengen impliziter Vorgänger zum Auffinden impliziter Enthaltenseins-Zyklen beruht (Def. 3.8.10), dann wird die Schnittregelanwendung, die die Formeln (28) und (29) erzeugt, nicht benötigt, sondern der Teilast, dessen Blatt der Knoten (26) ist, kann sofort geschlossen werden; er enthält einen impliziten Zyklus, weil aus $a \in e$ folgt, daß $a \in e \cup b$ (dieser Zyklus wird durch die Schnittregelanwendung explizit gemacht). Die Menge aller möglichen impliziten Vorgänger von Termen auf dem Teilast, der in Knoten (26) endet, ist $\{a, b, d, e, f(a), f(a \sqcup (b \sqcap a)), (e \sqcup b)\}$. Die Vorgänger-Mengen der Konstanten sind:

$$\begin{array}{rcl}
\mathcal{P}_\Pi(a) & = & \{e \sqcup b\} \\
\mathcal{P}_\Pi(b) & = & \emptyset \\
\mathcal{P}_\Pi(d) & = & \emptyset \\
\mathcal{P}_\Pi(e) & = & \{a\}
\end{array}$$

Die Vorgänger-Menge von $e \sqcup b$ ist

$$\mathcal{P}_\Pi(e \sqcup b) = \mathcal{P}_\Pi(e) \cup \mathcal{P}_\Pi(b) = \{a\} \ .$$

Damit haben wir $a \in \mathcal{P}_\Pi(e \sqcup b)$ und $e \sqcup b \in \mathcal{P}_\Pi(a)$, was das Vorhandensein eines impliziten Enthaltenseins-Zyklus anzeigt. $\square$

[6] An den i-ten Knoten in diesem Tableau ist die Bezeichnung $[i; j; (\mathrm{R})]$ angeheftet, was andeutet, daß seine Formel durch Anwendung des Regelschemas (R) von $\mathcal{C}^{\mathrm{EU}}_{\mathrm{MLSSF}}$ auf eine Prämisse erzeugt wurde, die aus der Formel des j-ten Knotens besteht.

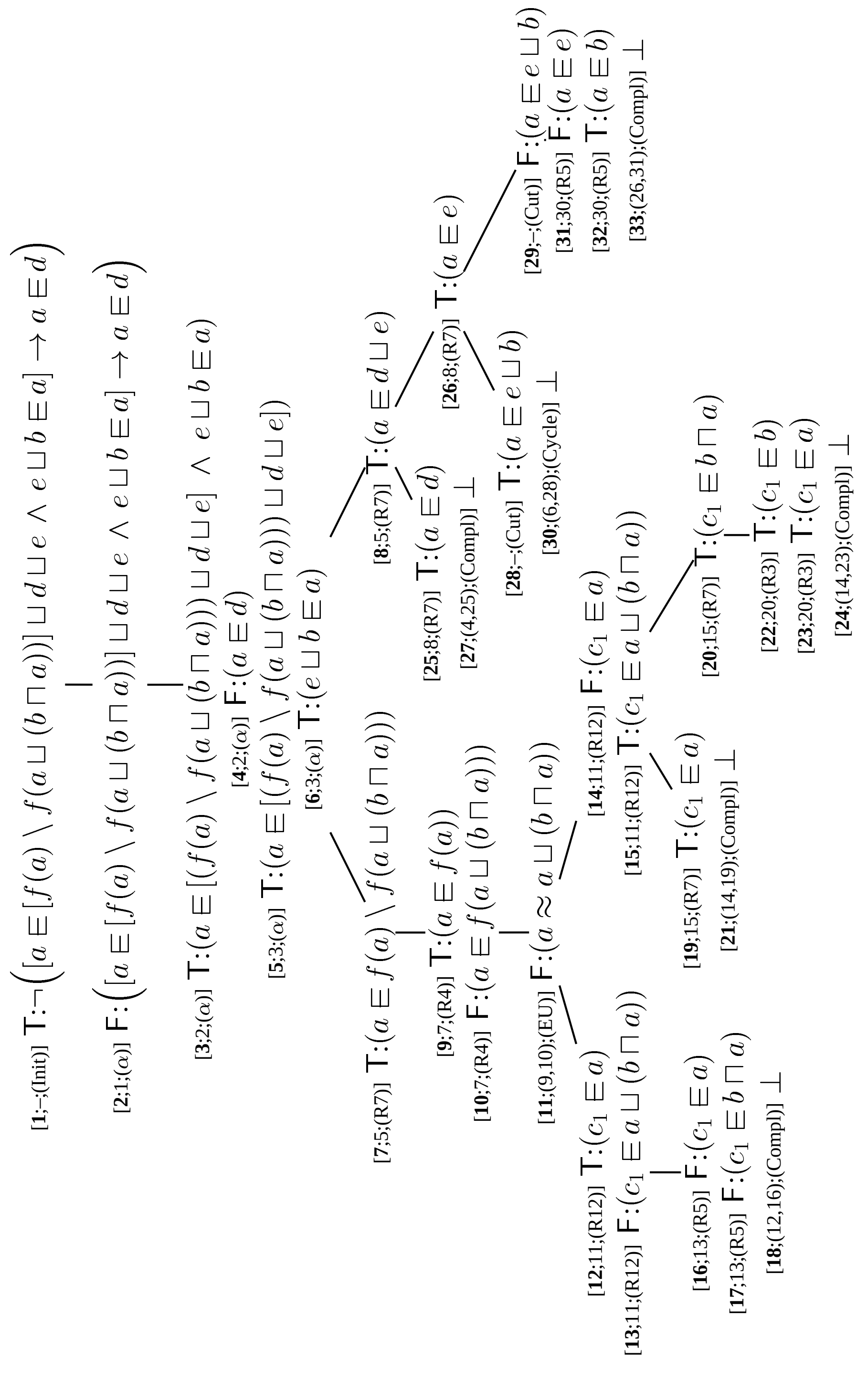


Abbildung 3.5: Tableaubeweis für eine MLSSF-Formel (Beispiel 3.8.25).

4 Erweiterungen

4.1 Einführung

In diesem Kapitel werden Techniken zur Verbesserung von Tableaukalkülen vorgestellt, die die auf den Kalkülen basierenden Beweisprozeduren effizienter machen. Nur solche Techniken werden in diesem Kapitel behandelt, die es erfordern, den Kalkül zu verändern. Heuristiken und Methoden zur Organisation der Beweissuche, d. h., Techniken zur Konstruktion einer effizienten Beweisprozedur basierend auf einem gegebenen Kalkül, werden im nächsten Kapitel behandelt.

Verstärkung des Kalküls Ein Kalkül wird verstärkt, wenn er in solcher Weise verändert wird, daß (zumindest für manche Formeln) kürzere Beweise existieren; das kann beispielsweise dadurch geschehen, daß man die Tableauregel so erweitert, daß zusätzliche Konklusionen aus bestimmten Prämissen abgeleitet werden können. In den meisten Fällen wird ein Kalkül indeterministischer, wenn er verstärkt wird; das heißt, es gibt bei jedem Erweiterungsschritt mehr verschiedene Möglichkeiten fortzufahren als mit dem ursprünglichen Kalkül. Man muß daher abwägen zwischen dem Vorteil, daß es kürzere Beweise gibt, und dem Nachteil, daß diese kürzeren Beweise schwieriger zu finden sein können, weil der Suchraum mehr Verzweigungspunkte hat.

Leider gibt es kein allgemeines Kriterium, um zu entscheiden, ob eine bestimmte Kalkül-Verstärkung nützlich für das Automatische Beweisen ist. Einige automatische Deduktionssysteme benutzen sogar abgeschwächte Kalküle, die weniger Wahlmöglichkeiten bieten, – auf Kosten einer Vergrößerung der jeweils kleinsten existierenden Beweise; dazu gehören beispielsweise solche Beweisprozeduren für prädikatenlogische Klausellogik, die eine starke Konnektionsbedingung verwenden (siehe Abschnitt 5.5).

Die nicht-analytische Schnittregel, die erlaubt, für alle Markierungen σ und Formeln ϕ die Konklusion $\{\{\sigma{:}\mathsf{T}{:}\phi\}, \{\sigma{:}\mathsf{F}{:}\phi\}\}$ aus der leeren Prämisse abzuleiten, ist ein typisches Beispiel für eine Kalkül-Erweiterung, bei der der Nachteil des zusätzlichen Indeterminismus den Vorteil, daß kürzere Beweise existieren, bei weitem überwiegt. In automatischen Deduktionssystemen wird die Schnittregel *niemals* ohne Einschränkung verwendet. Es gibt jedoch eingeschränkte Versionen der Schnittregel, wie beispielsweise die Erzeugung lokaler Lemmata (Abschnitt 4.5), mit der kürzere Beweise konstruiert werden können, ohne daß zu viele zusätzliche Wahlmöglichkeiten bei der Regelanwendung eingeführt werden.

Eine Methode zur Verstärkung von Kalkülen, die immer von Vorteil ist, ist die in Abschnitt 4.6 beschriebene *Pruning*-Technik, denn sie erlaubt, kleinere Tableaubeweise zu konstruieren, und führt keinerlei neue Wahlmöglichkeiten ein.

Eine anderer nützlicher Weg, Tableaukalküle zu verstärken, ist die Einführung *universeller Variablen* (Abschnitt 4.3); diese Technik kann zu exponentiell kürzeren Beweisen führen und führt nur wenige zusätzliche Wahlmöglichkeiten ein.

Aufschieben von Entscheidungen Oft ist es möglich, mehrere verschiedene Tableaus – und damit verschiedene Teile des Suchraums – durch ein einzelnes Tableau zu repräsentieren, wenn man zusätzliche syntaktische Hilfsmittel einführt. Ein typisches Beispiel ist die Technik der *starren Variablen* (Abschnitt 4.2), wobei Tableaus, die gleich sind bis auf die Ersetzung bestimmter Terme durch andere Terme, alle durch ein einziges Tableau repräsentiert werden, in dem eine freie (starre) Variable als Platzhalter für die verschiedenen Terme verwendet wird.

In gewissem Sinne führt das Aufschieben von Entscheidungen zu einer Breitensuche, bei der verschiedene Teile des Suchraums solange gleichzeitig durchsucht werden, bis genug Information angesammelt wurde, so daß eine informierte Entscheidung getroffen werden kann, auf welchen Teil des Suchraums sich die Suche im weiteren konzentrieren sollte. Wenn eine Tableauregelanwendung es beispielsweise erfordert, daß eine starre Variable X mit einem bestimmten Term t instantiiert wird, dann ist die Entscheidung, X mit t zu instantiieren in dem Sinne eine informierte Entscheidung, als man weiß, daß die Instantiierung sinnvoll ist, weil eine Regelanwendung existiert, die ohne die Instantiierung nicht möglich ist.

Techniken zum Aufschieben von Entscheidungen sind immer vorteilhaft, wenn man als Gütemaß die Zahl der Regelanwendungen betrachtet, die ausgeführt werden, bis ein Tableaubeweis gefunden ist. Solche Techniken können allerdings schwierig zu implementieren sein. Wenn beispielsweise eine starre Variable mit einem Term instantiiert wird, der sich später als die falsche Wahl herausstellt, dann kann es sehr viel schwerer werden, einen Tableaubeweis zu finden (nicht aber unmöglich, vorausgesetzt der Kalkül ist beweiskonfluent). Es ist also vorteilhaft, starre Variablen so lange nicht zu instantiieren, als es nicht absolut sicher ist, daß eine bestimmte Instantiierung nützlich ist. Eine Instantiierung, die möglicherweise nützlich ist, möglicherweise aber auch nicht, sollte statt dessen als Heuristik verwendet werden, d. h. Erweiterungsregelanwendungen, die mit solch einer Instantiierung kompatibel sind, werden bevorzugt, sind aber nicht notwendigerweise die einzigen, die betrachtet werden. Weil diese Methode aber schwierig zu implementieren ist, wenden praktisch alle tableaubasierten Deduktionssysteme für die Prädikatenlogik erster Stufe eine Substitution starrer Variablen durch Terme auf das ganze Tableau an, sobald festgestellt wurde, daß sie den Abschluß eines einzelnen Astes ermöglicht, – anstatt zuerst für jeden Ast des Tableaus nach einer Substitution zu suchen, die erlaubt ihn abzuschließen, und dann zu versuchen, alle diese Substitutionen zu kombinieren, um so eine einzelne Substitution zu konstruieren, mit der alle Äste zugleich abgeschlossen werden können.

Ein anderer Nachteil von Techniken zum Aufschieben von Entscheidungen ist, daß sie die Interaktion zwischen dem Beweissystem und dem Benutzer erschweren (der Grund für eine solche Interaktion kann sein, daß das System nicht voll automatisiert ist oder auch daß ein gefundener Beweis dem Benutzer mitgeteilt werden soll). Menschen bevorzugen Kalküle mit einer einfachen Erweiterungsregel. Jede ihrer Anwendungen sollte syntaktisch (nicht unbedingt semantisch) einfach sein, d. h., sie sollten einfach nachzuvollziehen und ihre Vorbedingungen einfach zu überprüfen sein, und sie sollte keine Seiteneffekte haben. Kalkül-Erweiterungen zum Aufschieben von Entscheidungen resultieren jedoch typischerweise in einem syntaktisch komplizierteren Kalkül; Regelanwendungen können nicht-lokale Seiteneffekte haben, wie beispielsweise die Instantiierung freier Variablen (das Problem der Benutzerinteraktion wird für den Fall der Prädikatenlogik erster Stufe in (Stenz *et al.*, 1999) diskutiert).

4.2 Tableaukalküle mit starren Variablen

4.2.1 Die Idee starrer Variablen

Das Konzept der starren Variablen[1], das aus dem Automatischen Beweisen in Prädikatenlogik erster Stufe bekannt ist, basiert auf der Beobachtung, daß die Erweiterungsregel vieler Kalküle für Logiken mit Termen für bestimmte Prämissen stabil ist bezüglich der Ersetzung von Termen in Π durch andere Terme; insbesondere gibt es Prämissen, aus der eine Konklusion $C(t)$ für alle Terme t abgeleitet werden kann.

Beispiel 4.2.1 Ein typisches Beispiel ist die Erweiterungsregel des Kalküls $\mathcal{C}_{\mathrm{PL1}}$ aus Abschnitt 3.6. Sie ist stabil für Prämissen, die aus α-, β-, γ- und δ-Formeln bestehen. Zum Beispiel kann die Konklusion $\{\{\alpha_1[t \mapsto t'], \alpha_2[t \mapsto t']\}\}$ für alle Terme t' aus einer Prämisse abgeleitet werden, die die α-Formel $\alpha[t \mapsto t']$ enthält. Prämissen, die eine γ-Formel γ enthalten, erlauben die Konklusion $C(t) = \{\{\gamma_1(t)\}\}$ für alle Terme t abzuleiten.

Die Erweiterungsregel von $\mathcal{C}_{\mathrm{PL1}}$ ist nur für Prämissen instabil, aus denen $\perp$ abgeleitet werden kann, die also erlauben einen Ast abzuschließen; beispielsweise kann aus $\{\mathsf{T}{:}\phi(s),\ \mathsf{F}{:}\phi(t)\}[t \mapsto t']$ nur dann $\perp = \perp[t \mapsto t']$ abgeleitet werden, wenn $t' = s$. $\square$

Wenn ein Kalkül eine Erweiterungsregel hat, dann kann man, anstatt Terme bei ihrer Einführung zu „raten“, diese durch eine starre Variable repräsentieren, die später

[1] In der Literatur – insbesondere der über Kalküle für die Prädikatenlogik erster Stufe – werden verschiedene andere Namen für freie und besonders für starre Variablen verwendet, so beispielsweise *Parameter*, *Dummy-Variable* und *Meta-Variable*.

„bei Bedarf“ instantiiert werden, wenn die Erweiterungsregel auf eine Prämisse angewendet wird, für die sie nicht stabil ist, wenn also die Regelanwendung ohne die Instantiierung der starren Variablen nicht möglich ist. Gewöhnlich können Unifikationsalgorithmen verwendet werden, um *allgemeinste* Substitutionen zu finden, die eine bestimmte Regelanwendung erlauben.

Intuitiv steht eine Formel, die eine starre Variable X enthält, für eine einzelnes (unbekanntes) Element der Menge aller Formeln, die man erhält, wenn man X durch einen Term ersetzt (vgl. Abschnitt 4.3, wo freie Variablen eingeführt werden, die *alle* Terme repräsentieren und *universelle* Variablen genannt werden). Alle Vorkommen einer starren Variablen in einem Tableau müssen immer mit dem selben Term instantiiert werden (was der Grund dafür ist, daß diese Variablen „starr“ genannt werden). Darum kann die Instantiierung einer starren Variablen um eine Erweiterungsregelanwendung möglich zu machen, eine andere Regelanwendung verhindern, was besonders problematisch ist, wenn der Kalkül nicht beweiskonfluent ist.

Kalküle mit starren Variablen werden gewöhnlich durch „Heben“ (*liften*) eines Grundkalküls konstruiert. Bisher ist das jedoch zumeist ein Ad-hoc-Verfahren. Ausgangspunkt ist die Beobachtung, das die Erweiterungsregel eines Kalküls für einige oder die meisten Prämissen stabil ist unter der Ersetzung von Termen durch andere Terme. Dann wird die Prämisse in einem ersten Schritt für diejenigen Prämissen gehoben, die die Ableitung einer Konklusion $C(t)$ für alle Terme t erlauben; in der Version mit starren Variablen kann aus einer solchen Prämisse die Konklusion $C(X)$ abgeleitet werden. In einem zweiten Schritt wird die Erweiterungsregel der Version mit starren Variablen für andere Prämissen entworfen (oft in Ad-hoc-Manier), so daß der Kalkül korrekt und vollständig ist. Das kann jedoch schwierig sein, und es kann versteckte Fallen geben – besonders, wenn der Grundkalkül nicht gutartig ist. Man muß genau überprüfen, für welche Prämissen die ursprüngliche Erweiterungsregel (ohne starre Variablen) stabil ist und für welche nicht.

Beispiel 4.2.2 Die Tatsache, daß das Heben eines Grundkalküls schwierig sein kann, wird durch die Geschichte der Kalküle mit starren Variablen für Prädikatenlogik erster Stufe veranschaulicht: Als die ersten Versionen definiert wurden, übersahen einige Autoren, daß die Grund-Erweiterungsregel, die sie benutzten, für δ-Formeln nicht stabil war, da sie verlangte, daß die durch eine Regelanwendung auf eine aus einer δ-Formel bestehende Prämisse eingeführte Konstante *neu* sei. Andere Autoren lösten (bzw. umgingen) dieses Problem, indem sie ein korrektes Erweiterungsregel-Schema für δ-Formeln entwarfen, das jedoch nicht durch Heben des entsprechenden Schemas des Grundkalküls entstand (siehe Abschnitt 4.4 für eine Diskussion, wie das Erweiterungsregel-Schema für δ-Formeln verbessert werden kann). $\square$

Solche versteckten Fallen können vermieden werden, indem man zunächst den Grundkalkül verändert, so daß er für möglichst viele Prämissen stabil wird, bevor man ihn hebt. Das Entwurfsproblem ist dann, die Regel des Grundkalküls stabil zu machen;

ein Beispiel ist die Erweiterungsregel aus Abschnitt 3.6, die etwas unintuitiv sein mag, die aber auch für δ-Formeln stabil ist.

Wenn ein Kalkül $\mathcal{C}^{\mathrm{fv}}$ mit starren Variablen durch Heben aus einem stabilen Grundkalkül $\mathcal{C}^{\mathrm{gd}}$ konstruiert worden ist, dann folgen Korrektheit und Vollständigkeit von $\mathcal{C}^{\mathrm{fv}}$ aus Korrektheit und Vollständigkeit von $\mathcal{C}^{\mathrm{gd}}$ und müssen nicht extra bewiesen werden. Insbesondere muß man also, da die Beziehung zwischen der Grundversion und der Version mit starren Variablen eines Kalküls rein syntaktisch ist, nicht extra eine geeignete Semantik für Tableaus mit starren Variablen entwerfen. Die Semantik eines Tableaus mit starren Variablen kann in uniformer Weise, basierend auf der Semantik seiner Grundinstanzen, definiert werden (Abschnitt 4.2.9).

4.2.2 Syntax von Tableaukalkülen mit freien Variablen

Die Ersetzung von Termen durch freie Variablen ist nicht auf den Formelteil G einer Tableauformel $\mathsf{S}{:}\sigma{:}G$ beschränkt, sondern diese können auch in der Markierung σ und dem Vorzeichen S verwendet werden. Das heißt, wir machen die Annahme, daß die Menge der Tableauformeln eine Sprache mit Termen ist. Markierungen und Formeln können gegebenenfalls Variablen derselben Sorte enthalten (und also auch Variablen gemeinsam haben). Freie Variablen, die in Vorzeichen verwendet werden, müssen von einer besonderen Sorte s sein, so daß T und F die einzigen Terme von der Sorte s sind.

Definition 4.2.3 Ein Kalkül $\mathcal{C}^{\mathrm{fv}}$ für eine Logik $\mathbf{L}$ heiße ein *Kalkül mit freien Variablen* (bzgl. der Menge Var freier Variablen), wenn für jede Signatur $\Sigma \in Sig$ von $\mathbf{L}$ diejenige Erweiterung Σ^*_{fv} von Σ, die verwendet wird, um Tableauformeln aufzubauen, auch Erweiterung einer Signatur $\Sigma^*_{\mathrm{gd}} \in Sig$ ist (die ebenfalls eine Erweiterung von Σ ist), so daß folgende gilt:

1. die Menge $TabForm(\Sigma^*_{\mathrm{gd}})$ von Tableauformeln über Σ_{gd} ist eine Sprache mit Termen (Def. 2.2.1);

2. die Menge $TabForm(\Sigma^*_{\mathrm{fv}})$ von Tableauformeln über Σ_{fv} ist die (gemäß Definition 2.2.2) aus $TabForm(\Sigma^*_{\mathrm{gd}})$ und der Menge Var freier Variablen konstruiert Sprache mit freien Variablen, d. h.,

$$TabForm(\Sigma^*_{\mathrm{fv}}) = (\,TabForm(\Sigma^*_{\mathrm{gd}}))^{\mathrm{fv}} \quad .$$

Die Termmengen von $TabForm(\Sigma^*_{\mathrm{fv}})$ und $TabForm(\Sigma^*_{\mathrm{gd}})$ seien mit $TabTerm(\Sigma^*_{\mathrm{fv}})$ bzw. $TabTerm(\Sigma^*_{\mathrm{gd}})$ bezeichnet.

Ein Kalkül $\mathcal{C}^{\mathrm{gd}}$ ist ein *Grundkalkül* (bzgl. der Menge Var freier Variablen), wenn seine Tableauformeln keinerlei freie Variablen aus Var enthalten. □

Beispiel 4.2.4 Eines der seltenen Beispiele für die Nützlichkeit freier Variablen in Vorzeichen ist ein Erweiterungsregel-Schema, das erlaubt, die nicht verzweigende Konklusion $\{\{X{:}F,\ X{:}G\}\}$ aus einer Prämisse, die die Äquivalenz $\{\mathsf{T}{:}(F \leftrightarrow G)\}$ enthält, abzuleiten. □

Die Klasse der Tableaukalküle mit freien Variablen ist nur dadurch charakterisiert, daß ihre Tableauformeln freie Variablen enthalten. Alle Definitionen aus Kapitel 3 von Begriffen wie Ast, Tableau, Tableaubeweis, Beweiskonfluenz, Monotonie, usw. bleiben unverändert. Die Begriffe eines Kalkül mit Erweiterungsregel und der Gutartigkeit sind jedoch in dem Fall, daß ein Kalkül starre Variablen verwendet, nicht angemessen und müssen entsprechend angepaßt werden (siehe Abschnitte 4.2.3 und 4.2.5).

4.2.3 Starre Variablen verwendende Tableaukalküle mit Erweiterungsregel

Der Begriff eines Kalküls mit Erweiterungsregel (Abschnitt 3.3.3) – und also der eines gutartigen Kalküls (Abschnitt 3.3.7) – sind für Kalküle, die starre Variablen verwenden nicht angemessen, weil bei einem Kalkül mit Erweiterungsregel alle Tableauformeln auf einem Tableau T unverändert bleiben müssen, wenn es erweitert wird; es ist nicht zulässig, in T auftretende starre Variablen zu instantiieren. Darum führen wir den Begriff einer *Erweiterungsregel mit starren Variablen ein*, wobei eine Konklusion (neben einer endlichen Menge von Extensionen) eine Substitution enthält, die auf das ganze Tableau angewendet wird, wenn C für die Erweiterung des Tableaus verwendet wird. Von der Anwendung dieser Substitution abgesehen, dürfen Tableauregelanwendungen auch weiter nur lokale Auswirkungen haben. Das heißt, eine Regelanwendung erweitert *einen* Ast eines Tableaus und, was die Möglichkeiten für die Erweiterung eines Astes B sind, hängt nur von B selbst ab; keine zusätzlichen Vorbedingungen sind erlaubt, wie Beispielsweise das Vorhandensein bestimmter Formeln auf anderen Ästen.

Definition 4.2.5 Sei $\mathcal{C}$ ein Tableaukalkül mit freien Variablen für eine Logik **L**; und sei $\Sigma \in \mathit{Sig}$ eine Signatur von **L**.

Eine *Konklusion mit starren Variablen* ist ein Paar $\langle C, \sigma \rangle$, das aus einer endlichen Menge C von Extensionen (Def. 3.3.4) und einer Substitution $\sigma \in \mathit{Subst}(\Sigma^*_{\mathrm{fv}})$ besteht, so daß $C = C\sigma$.

Eine *Erweiterungsregel* $\mathcal{E}(\Sigma)$ *mit starren Variablen* ist eine Funktion, die jedem (endlichen) Tableauast, dessen Knoten mit Formeln aus $\mathit{TabForm}(\Sigma^*_{\mathrm{fv}})$ markiert sind, eine Menge $\mathcal{E}(\Sigma)(B)$ (möglicher) Konklusionen mit starren Variablen zuweist, die unendlich sein darf, aber aufzählbar sein muß. □

Definition 4.2.6 Sei $\mathcal{C}$ eine Tableaukalkül mit starren Variablen für eine Logik **L**; und sei $\Sigma \in \mathit{Sig}$ eine Signatur von **L**.

Eine Erweiterungsregel $\mathcal{E}(\Sigma)$ mit starren Variablen *charakterisiert* die Tableauregel $\mathcal{R}(\Sigma)$ von $\mathcal{C}$, wenn für alle Tableaus T über Σ^*_{fv} das folgende gilt: Ein Tableau T' ist genau dann ein Nachfolgetableau von T (d. h., $T' \in \mathcal{R}(\Sigma)(T)$), wenn es einen Ast B in T und eine Konklusion $\langle C, \sigma \rangle$ mit starren Variablen in $\mathcal{E}(\Sigma)(B)$ gibt, so daß das Tableau T' konstruiert werden kann, indem

1. der Ast B für jede Extension E in C mit einem neuen Teilast erweitert wird, wobei die Knoten in diesem Teilast mit den Elementen von E markiert sind, und

2. die Substitution σ auf das Tableau angewendet wird.

Wenn die Erweiterungsregel $\mathcal{E}(\Sigma)$ mit starren Variablen die Tableauregel $\mathcal{R}(\Sigma)$ von $\mathcal{C}$ für alle Signaturen Σ charakterisiert, dann heiße $\mathcal{E}$ *die* Erweiterungsregel mit starren Variablen von $\mathcal{C}$; und $\mathcal{C}$ heiße ein *Kalkül mit starren Variablen und mit Erweiterungsregel* $\mathcal{E}$. $\square$

4.2.4 Erweiterungsregeln mit starren Variablen, die bezüglich Substitution monoton sind

Intuitiv ist die Anwendung der Substitution σ, die Teil einer Konklusion $\langle C, \sigma \rangle$ mit starren Variablen ist, eine Vorbedingung für die Ableitung von C aus den Formeln auf dem Ast. Wenn ein (Grund-)Kalkül monoton ist (wie in Abschnitt 3.3.5 definiert), dann ist die Vorbedingung für die Ableitung einer bestimmten Konklusion, daß bestimmte Formeln auf dem erweiterten Ast vorhanden sind – nicht aber die Abwesenheit von Formeln. Dementsprechend kann man den Begriff der Monotonie bzgl. der Anwendung von Substitutionen definieren: Nur, daß eine bestimmte Variable in bestimmter Weise instantiiert *ist* oder *werden kann*, ist als Vorbedingung zulässig – nicht aber, daß die Variable *nicht* in dieser Weise instantiiert ist. Jedoch kann natürlich eine andere, inkompatible Instantiierung die Regelanwendung sehr wohl verhindern.

Definition 4.2.7 Sei $\mathcal{C}$ ein Kalkül mit starren Variablen und mit Erweiterungsregel $\mathcal{E}$ für eine Logik **L**.

Die Erweiterungsregel $\mathcal{E}$ und der Kalkül $\mathcal{C}$ heißen *monoton bzgl. Substitution*, wenn für alle Signaturen $\Sigma \in \mathit{Sig}$ von **L** und alle Äste über Σ^* das folgende gilt:

> Wenn $\langle C, \sigma \rangle \in \mathcal{E}(B)$ und $\rho, \tau \in \mathit{Subst}(\Sigma^*)$ mit $\sigma = \rho \circ \tau$,
> dann $\langle C, \rho \rangle \in \mathcal{E}(B\tau)$.

$\square$

Beispiel 4.2.8 Angenommen, daß – in einem Kalkül für PL1, der monoton ist bzgl. Substitution, – $\langle \{\{\mathsf{T}{:}p(a,b)\}\}, \{X \mapsto a, Y \mapsto b\} \rangle$ eine mögliche Konklusion für eine Ast $B(X,Y)$ ist, dann ist $\langle \{\{\mathsf{T}{:}p(a,b)\}\}, \{Y \mapsto b\} \rangle$ eine mögliche Konklusion für $B(a,Y)$, und $\langle \{\{\mathsf{T}{:}p(a,b)\}\}, \mathit{id} \rangle$ ist eine mögliche Konklusion für $B(a,b)$. $\square$

4.2.5 Gutartige Tableaukalküle mit starren Variablen

Gutartigkeit, die Eigenschaft, die das „Wohlverhalten“ von Tableaukalkülen erzwingt, wird für Kalküle mit starren Variablen in ähnlicher Weise definiert wie für Grundkalküle. Ein Grundkalkül ist gutartig, wenn er nicht-strukturell und monoton ist und ein Kalkül mit Erweiterungsregel ist. Ein Kalkül mit starren Variablen ist gutartig, wenn er nicht-strukturell ist (wie im Grundfall), er monoton ist (wie im Grundfall) und darüberhinaus auch monoton bzgl. Substitution und er eine Erweiterungsregel *mit starren Variablen* hat, also die Anwendung von Substitutionen gestattet ist (statt einer Grund-Erweiterungsregel).

Definition 4.2.9 Ein Tableaukalkül mit starren Variablen heiße *gutartig*, wenn er

1. nicht-strukturell ist,
2. monoton ist,
3. monoton ist bzgl. Substitution und
4. eine Erweiterungsregel mit starren Variablen hat.

$\square$

Wie im Grundfall (siehe Lemma 3.3.11) kann die Erweiterungsregel $\mathcal{E}$ eines gutartigen Kalküls mit starren Variablen als Funktion $\tilde{\mathcal{E}}$ auf Prämissen dargestellt werden; und wir identifizieren die Erweiterungsregel $\mathcal{E}$ und die Funktion $\tilde{\mathcal{E}}$.

Lemma 4.2.10 *Sei $\mathcal{C}$ ein Tableaukalkül mit starren Variablen für eine Logik* **L***, der gutartig ist und eine Erweiterungsregel $\mathcal{E}$ mit starren Variablen hat. Dann gibt es für alle Signaturen Σ von* **L** *eine (eindeutig bestimmte) Funktion $\tilde{\mathcal{E}}(\Sigma)$, die jeder (endlichen) Prämisse $\Pi \subset TabForm(\Sigma^*_{\text{fv}})$ eine Menge $\tilde{\mathcal{E}}(\Sigma)(\Pi)$ (möglicher) Konklusionen mit starren Variablen zuweist, so daß*

$$\mathcal{E}(\Sigma)(B) = \tilde{\mathcal{E}}(\Sigma)(Form(B))$$

für alle Tableauäste B über Σ^*_{fv}.

Der Begriff der *minimalen* Prämisse einer Konklusion mit starren Variablen ist wie im Grundfall definiert (Def. 3.3.13).

Erweiterungsregeln gutartiger Kalküle mit starren Variablen können mit Hilfe von Regelschemata beschrieben werden, an die eine Erklärung angefügt ist, wie die Substitution zu berechnen ist, die angewendet wird, wenn eine Instanz des Schemas verwendet wird, um einen Tableauast zu erweitern (siehe Abschnitte 4.2.10 und 4.2.11 für Beispiele).

4.2.6 Eine Subsumtionsrelation auf Konklusionen mit starren Variablen

Die Subsumtionsrelation (oder Ist-allgemeiner-als-Relation) $\leq^W$, die auf Substitutionen definiert ist, kann auf Konklusionen mit starren Variablen wie folgt erweitert werden.

Definition 4.2.11 Sei $\mathcal{C}$ ein gutartiger Kalkül mit starren Variablen für eine Logik **L**; seien $\langle C, \sigma\rangle$ und $\langle C', \sigma'\rangle$ Konklusionen mit starren Variablen über der gleichen Signatur $\Sigma^*_{\text{fv}} \in \mathit{Sig}$; und sei $W \subset \mathit{Var}$ eine endliche Menge starrer Variablen.

Die Konklusion $\langle C, \sigma\rangle$ *subsumiert* die Konklusion $\langle C', \sigma'\rangle$, in Zeichen

$$\langle C, \sigma\rangle \leq^W \langle C', \sigma'\rangle \ ,$$

wenn es eine Substitution $\rho \in \mathit{Subst}(\Sigma^*_{\text{fv}})$ gibt, so daß

1. $\sigma \leq^W \rho \leq^W \sigma'$, wobei $\leq^W$ die Subsumtionsrelation auf Substitutionen aus Definition 2.2.6 ist;
2. $C\rho = C'$.

$\square$

Lemma 4.2.12 *Die Subsumtionsrelation $\leq^W$ auf Konklusionen mit starren Variablen ist transitiv.*

Beweis: Wir nehmen an, daß $\langle C, \sigma\rangle \leq^W \langle C', \sigma'\rangle$ und $\langle C', \sigma'\rangle \leq^W \langle C'', \sigma''\rangle$.

Die Substitutionen ρ und ρ', die gemäß der Definition der Relation $\leq^W$ existieren, erfüllen die Bedingungen $\sigma \leq^W \rho \leq^W \sigma'$ und $\sigma' \leq^W \rho' \leq^W \sigma''$; da die Relation $\leq^W$ auf Substitutionen transitiv ist, impliziert das $\sigma \leq^W \rho' \leq^W \sigma''$.

Außerdem haben wir $C\rho' = (C\rho)\rho' = C'\rho' = C''$. Also gilt $\langle C, \sigma\rangle \leq^W \langle C'', \sigma''\rangle$, denn die Substitution ρ' hat die nötigen Eigenschaften. $\square$

Lemma 4.2.13 *Seien $\langle C, \sigma\rangle$ und $\langle C, \sigma'\rangle$ Konklusionen mit starren Variablen, die sich nur in ihren Substitutionen unterscheiden. Wenn $\sigma \leq^W \sigma'$, dann $\langle C, \sigma\rangle \leq^W \langle C, \sigma'\rangle$.*

Beispiel 4.2.14 Eine Konklusion mit einer allgemeineren Substitution ist allgemeiner, d. h.,

$$\langle\{\{\bot\}\}, id\rangle \ \text{ subsumiert } \ \langle\{\{\bot\}\}, \{X \mapsto a\}\rangle \ ,$$

und wenn $Y \notin W$, dann

$$\langle\{\{\bot\}\}, \{X \mapsto f(Y)\}\rangle \ \text{ subsumiert } \ \langle\{\{\bot\}\}, \{X \mapsto f(a)\}\rangle \ .$$

$\square$

Intuitiv bleibt die Vollständigkeit eines gutartigen Tableaukalküls mit starren Variablen erhalten, wenn aus jeder Prämisse nur bzgl. $\leq^W$ allgemeinste Konklusionen abgeleitet werden. Jedoch muß die Variablenmenge W mit Bedacht gewählt werden. Wenn der Kontext bekannt ist (d. h., ein bestimmtes Tableau), dann reicht es aus, wenn W alle starren Variablen enthält, die in diesem Kontext vorkommen. Andernfalls muß man sicherstellen, daß die Vollständigkeit für jede beliebige Variablenmenge W, die in dem unbekannten Kontext vorkommen könnte, erhalten bleibt.

Beispiel 4.2.15 Die Konklusion

$$C_X = \langle \{\{p(X)\}\}, id \rangle$$

subsumiert die Konklusion

$$C_t = \langle \{\{p(t)\}\}, id \rangle$$

für alle Terme t, falls $X \notin W$; in diesem Fall kann die einzelne Konklusion C_X alle Konklusionen C_t ersetzen.

Wenn jedoch der Kontext unbekannt ist, und also die Menge W die Variable X enthält (oder auch nicht), dann muß man vorsichtig sein, denn C_X subsumiert C_t *nicht*, falls $X \in W$. In diesem Fall kann man die Menge $\{C_X \mid X \in \mathit{Var}\}$ von Konklusionen benutzen, um die Menge $\{C_t \mid t \in \mathit{TabTerm}\}$ zu ersetzen, weil die Zahl der Variablen in jedem beliebigen Kontext immer endlich ist und es daher immer Variablen $X \in \mathit{Var}$ gibt, die im Kontext nicht vorkommen. □

4.2.7 Die Version mit starren Variablen eines Grundkalküls

Wie schon zuvor gesagt, kann ein Tableaukalkül $\mathcal{C}^{\mathrm{rv}}$ mit starren Variablen aus einem Grundkalkül $\mathcal{C}^{\mathrm{gd}}$ durch *Heben* konstruiert werden. Die folgende Definition beschreibt die Beziehung zwischen $\mathcal{C}^{\mathrm{rv}}$ und $\mathcal{C}^{\mathrm{gd}}$.

Definition 4.2.16 Sei $\mathcal{C}^{\mathrm{rv}}$ ein gutartiger Kalkül mit starren Variablen für eine Logik **L**; und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für **L**, so daß für alle Signaturen $\Sigma \in \mathit{Sig}$ die Erweiterte Signatur Σ^*_{gd}, die von $\mathcal{C}^{\mathrm{gd}}$ benutzt wird, diejenige Signatur ist, die gemäß der Definition von Tableaukalkülen mit freien Variablen (Def. 4.2.3) existieren muß, für die

$$\mathit{TabForm}(\Sigma^*_{\mathrm{fv}}) = (\mathit{TabForm}(\Sigma^*_{\mathrm{gd}}))^{\mathrm{fv}}$$

gilt.

Der Kalkül $\mathcal{C}^{\mathrm{rv}}$ ist eine *Version mit starren Variablen* des Kalküls $\mathcal{C}^{\mathrm{gd}}$ (und $\mathcal{C}^{\mathrm{gd}}$ ist eine *Grundversion* von $\mathcal{C}^{\mathrm{rv}}$), wenn für all Prämissen $\Pi_{\mathrm{rv}} \subset \mathit{TabForm}(\Sigma^*_{\mathrm{fv}})$ (mit starren Variablen), alle Prämissen $\Pi_{\mathrm{gd}} \subset \mathit{TabForm}(\Sigma^*_{\mathrm{gd}})$ (ohne starren Variablen) und alle Substitutionen $\tau \in \mathit{Subst}(\Sigma^*_{\mathrm{fv}})$ mit endlichem Definitionsbereich, so daß

$$\Pi_{\mathrm{fv}}\tau = \Pi_{\mathrm{gd}} \quad ,$$

das folgende gilt, wobei $\mathcal{E}^{\text{gd}}$ und $\mathcal{E}^{\text{rv}}$ die Erweiterungsregeln von $\mathcal{C}^{\text{gd}}$ bzw. $\mathcal{C}^{\text{rv}}$ sind und $W = dom(\tau)$:

$$\mathcal{E}^{\text{gd}}(\Pi_{\text{gd}}) = \{C_{\text{gd}} \mid \text{es gibt } \langle C_{\text{rv}}, \sigma\rangle \in \mathcal{E}^{\text{rv}}(\Pi_{\text{rv}}), \text{so daß } \langle C_{\text{rv}}, \sigma\rangle \leq^W \langle C_{\text{gd}}, \tau\rangle\} \ .$$

□

Beispiel 4.2.17 Tabelle 4.1 zeigt Beispiele aus der Prädikatenlogik erster Stufe für die drei Haupttypen von Grundprämissen (und ihre Konklusionen), für die die Erweiterungsregel gehoben werden kann, und die jeweilige Version mit starren Variablen.

In jedem dieser Fälle erlaubt die Grund-Erweiterungsregel (siehe Abschnitt 3.6), die Konklusion C_{gd} aus der Prämisse Π_{gd} abzuleiten, und die Erweiterungsregel mit starren Variablen (siehe Abschnitt 4.2.10) erlaubt, die Konklusion C_{rv} aus der Prämisse Π_{rv} abzuleiten.

Unter Verwendung der Notation aus Definition 4.2.16 ist die Beziehung durch die Substitution τ, für die $\Pi_{\text{rv}}\tau = \Pi_{\text{gd}}$ gilt, und die Substitution ρ, für die $C_{\text{rv}}\rho = C_{\text{gd}}$ gilt, angedeutet.

Im ersten Beispiel reicht es *nicht* aus, wenn $\langle\{\{\mathsf{T}{:}{*}{:}p(X, Y)\}\}, id\rangle$ für nur eine starre Variable $X \in Var$ eine Konklusion von $\{\mathsf{T}{:}{*}{:}(\forall x)(p(x, Y))\}$ ist, weil eine andere Substitution $\tau' = \{Y \mapsto c, X \mapsto d\}$ die Variable X instantiieren könnte; dann würde X ein Element von W sein und also

$$\langle\{\{\mathsf{T}{:}{*}{:}p(X, Y)\}\}, id\rangle \not\leq^W \langle\{\{\mathsf{T}{:}{*}{:}p(t, c)\}\}, \{Y \mapsto c, X \mapsto d\}\rangle \ .$$

□

Eine Grundversion eines beliebigen Kalküls mit starren Variablen kann leicht angegeben werden, indem man festlegt: falls ein Paar $\langle C_{\text{rv}}, \sigma\rangle$ eine Konklusion mit starren Variablen einer Prämisse Π_{rv} ist und ρ eine Substitution, die sowohl C_{rv} als auch Π_{rv} grundiert und für die $\sigma \leq \rho$ gilt, dann sei $C_{\text{gd}} = C_{\text{rv}}\rho$ eine Konklusion (ohne starren Variablen) der Prämisse $\Pi_{\text{gd}} = \Pi_{\text{rv}}$.

Der Vorteil, wenn man einen Kalkül $\mathcal{C}^{\text{rv}}$ mit starren Variablen aus einem Grundkalkül $\mathcal{C}^{\text{gd}}$ durch Heben konstruiert, ist, daß Korrektheit und Vollständigkeit von $\mathcal{C}^{\text{rv}}$ aus der Korrektheit und Vollständigkeit von $\mathcal{C}^{\text{gd}}$ folgen.

Satz 4.2.18 *Sei $\mathcal{C}^{\text{rv}}$ ein gutartiger Kalkül mit starren Variablen für eine Logik $\mathbf{L}$, und sei $\mathcal{C}^{\text{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\text{rv}}$ eine Version mit starren Variablen von $\mathcal{C}^{\text{gd}}$ ist (Def. 4.2.16).*

Dann gilt für alle Signaturen $\Sigma \in Sig$ und alle endlichen Mengen $G \subset Form(\Sigma)$ von Formeln, daß es genau dann einen $\mathcal{C}^{\text{rv}}$-Tableaubeweis für G gibt, wenn es einen $\mathcal{C}^{\text{gd}}$-Tableaubeweis für G gibt.

	Π	C	
gd	$\{\mathsf{T}{:}{*}{:}(\forall x)(p(x,c))\}$	$\{\{\mathsf{T}{:}{*}{:}p(t,c)\}\}$	für alle Terme t
rv	$\{\mathsf{T}{:}{*}{:}(\forall x)(p(x,Y))\}$	$\langle\{\{\mathsf{T}{:}{*}{:}p(X,Y)\}\}, id\rangle$	für alle $X \in \mathit{Var}$
	$\tau = \{Y \mapsto c\}$	$\rho = \{Y \mapsto c,\, X \mapsto t\}$	

	Π	C
gd	$\{\mathsf{T}{:}{*}{:}p(t) \land q(t)\}$	$\{\{\mathsf{T}{:}{*}{:}p(t), \mathsf{T}{:}{*}{:}q(t)\}\}$
rv	$\{\mathsf{T}{:}{*}{:}p(X) \land q(X)\}$	$\langle\{\{\mathsf{T}{:}{*}{:}p(X), \mathsf{T}{:}{*}{:}q(X)\}\}, id\rangle$
	$\tau = \{X \mapsto t\}$	$\rho = \{X \mapsto t\}$

	Π	C
gd	$\{\mathsf{T}{:}{*}{:}p(t), \mathsf{F}{:}{*}{:}p(t)\}$	$\{\{\bot\}\}$
rv	$\{\mathsf{T}{:}{*}{:}p(t'), \mathsf{F}{:}{*}{:}p(t'')\}$	$\langle\{\{\bot\}\}, \sigma\rangle$ wobei σ ein MGU von t', t'' ist
	τ mit $t = t'\tau = t''\tau$	$\rho = \tau$

Tabelle 4.1: Beispiele für die Beziehung zwischen einem Grundkalkül für PL1 und seiner Version mit starren Variablen (siehe Beispiel 4.2.17).

Beweis: *Wenn-Teil:* Angenommen, $T_0^{\mathrm{gd}}, \ldots, T_n^{\mathrm{gd}}$ $(n \geq 0)$ ist ein Tableaubeweis für G, der mit Hilfe des Grundkalküls $\mathcal{C}^{\mathrm{gd}}$ konstruiert ist.

Durch Induktion über i beweisen wir, daß es einen Tableaubeweis $T_0^{\mathrm{rv}}, \ldots, T_n^{\mathrm{rv}}$ mit starren Variablen gibt, so daß $T_i^{\mathrm{rv}}\tau_i = T_i^{\mathrm{gd}}$ für Substitutionen $\tau_i \in \mathit{Subst}(\Sigma^*_{\mathrm{fv}})$.

$i = 0$: Der Kalkül $\mathcal{C}_{\mathrm{rv}}$ ist nicht-strukturell, und in G kommen keine starren Variablen vor. Darum gilt $T_0^{\mathrm{rv}}\tau_0 = T_0^{\mathrm{gd}}$ für $T_0^{\mathrm{rv}} = T_0^{\mathrm{gd}}$ und $\tau_0 = id$.

$i \to i + 1$: Wenn T_{i+1}^{gd} aus T_i^{gd} durch Anwendung der Erweiterungsregel von $\mathcal{C}^{\mathrm{gd}}$ auf eine Prämisse Π^{gd} auf einem Ast B_i^{gd} von T_i^{gd} entstanden ist, wobei die Konklusion C^{gd} verwendet worden ist, dann gibt es eine Prämisse Π^{rv} auf einem Ast B_i^{rv} von T_i^{rv}, so daß $\Pi_i^{\mathrm{rv}}\tau_i = \Pi^{\mathrm{gd}}$ und $B_i^{\mathrm{rv}}\tau_i = B_i^{\mathrm{gd}}$. Also gibt es gemäß der in Definition 4.2.16 festgelegten Beziehung zwischen der Grundversion eines Kalküls und seiner Version mit starren Variablen eine Konklusion $\langle C^{\mathrm{rv}}, \sigma\rangle$ mit starren Variablen, die aus Π^{rv} mit der Erweiterungsregel von $\mathcal{C}^{\mathrm{rv}}$ ableitbar ist, und es gibt eine Substitution ρ_i, so daß $C^{\mathrm{rv}}\rho_i = C^{\mathrm{gd}}$ und $\tau_i \leq^W \rho_i$, wobei W der Definitionsbereich von τ_i ist (der alle in T_i^{rv} auftretenden Variablen enthält); daraus folgt, daß $T_i^{\mathrm{rv}}\rho_i = T_i^{\mathrm{gd}}$. Also erfüllen die Substitution $\tau_{i+1} = \rho_i$ und das Tableau T_{i+1}^{rv}, das aus T_i^{rv} durch Erweiterung des Astes B_i^{rv} von T_i^{rv} mit der Konklusion C^{rv} entsteht, die Bedingung $T_{i+1}^{\mathrm{rv}}\tau_{i+1} = T_{i+1}^{\mathrm{gd}}$.

Genau-dann-Teil: Angenommen, $T_0^{\mathrm{rv}}, \ldots, T_n^{\mathrm{rv}}$ $(n \geq 0)$ ist ein Tableaubeweis für G, der mit Hilfe der Erweiterungsregel mit starren Variablen des Kalküls $\mathcal{C}^{\mathrm{rv}}$ konstruiert ist.

Durch Induktion über i beweisen wir, daß es einen Grund-Tableaubeweis $T_0^{\mathrm{gd}}, \ldots, T_n^{\mathrm{gd}}$ gibt, so daß $T_i^{\mathrm{gd}} = T_i^{\mathrm{rv}}\tau_i$ für Substitutionen $\tau_i \in \mathit{Subst}(\Sigma^*_{\mathrm{rv}})$ gilt $(0 \leq i \leq n)$.

$i = 0$: Der Kalkül $\mathcal{C}_{\mathrm{gd}}$ ist nicht-strukturell, und in G kommen keine starren Variablen vor. Darum gilt $T_0^{\mathrm{gd}} = T_0^{\mathrm{rv}}\tau_0$ für $T_0^{\mathrm{gd}} = T_0^{\mathrm{rv}}$ und $\tau_0 = id$.

$i \to i+1$: Wenn T_{i+1}^{rv} aus T_i^{rv} durch Anwendung der Erweiterungsregel mit starren Variablen des Kalküls $\mathcal{C}^{\mathrm{rv}}$ auf eine Prämisse Π^{rv} auf einem Ast B_i^{rv} von T_i^{fv} entstanden ist, wobei eine Konklusion $\langle C^{\mathrm{rv}}, \sigma\rangle$ mit starren Variablen verwendet worden ist, dann gibt es eine Prämisse Π^{gd} auf einem Ast B_i^{gd} von T_i^{gd}, so daß $\Pi^{\mathrm{gd}} = \Pi_i^{\mathrm{rv}}\tau_i$ und $B_i^{\mathrm{gd}} = B_i^{\mathrm{rv}}\tau_i$. Also gibt es gemäß der in Definition 4.2.16 festgelegten Beziehung zwischen der Grundversion eines Kalküls und seiner Version mit starren Variablen eine Substitution ρ, so daß $\sigma \leq^W \tau_i \leq^W \rho_i$ und die Grund-Konklusion $C^{\mathrm{gd}} = C^{\mathrm{rv}}\rho_i$ kann aus Π^{gd} mit Hilfe der Grund-Erweiterungsregel des Kalküls C^{gd} abgeleitet werden. Da $\sigma \leq^W \tau_i \leq^W \rho_i$, gilt $T_i^{\mathrm{gd}} = T_i\tau_i = T_i\sigma\rho_i$ und $C^{\mathrm{gd}} = C\rho_i = C\sigma\rho_i$. Also erfüllen die Substitution $\tau_{i+1} = \rho_i$ und das Tableau T_{i+1}^{gd}, das aus T_i^{gd} durch Erweiterung des Astes B_i^{gd} von T_i^{gd} mit der Konklusion C^{gd} entsteht, die Bedingung $T_{i+1}^{\mathrm{gd}} = T_{i+1}^{\mathrm{rv}}\tau_{i+1}$.
□

Korollar 4.2.19 *Sei $\mathcal{C}^{\mathrm{rv}}$ ein gutartiger Kalkül mit starren Variablen für eine Logik* $\mathbf{L}$*, und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für* $\mathbf{L}$*, so daß $\mathcal{C}^{\mathrm{rv}}$ eine Version mit starren Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.*

In diesem Fall ist $\mathcal{C}^{\mathrm{rv}}$ genau dann korrekt und vollständig, wenn $\mathcal{C}^{\mathrm{gd}}$ korrekt und vollständig ist.

Aus Satz 4.2.18 folgt auch, daß Korrektheit und Vollständigkeit eines Kalküls $\mathcal{C}_2^{\mathrm{rv}}$ mit starren Variablen aus Korrektheit und Vollständigkeit eines Kalküls $\mathcal{C}_2^{\mathrm{rv}}$ folgen, wenn $\mathcal{C}_1^{\mathrm{rv}}$ und $\mathcal{C}_2^{\mathrm{rv}}$ dieselbe Grund-Version $\mathcal{C}^{\mathrm{gd}}$ haben.

4.2.8 Konstruktion eines Kalküls mit starren Variablen durch Heben

In diesem Abschnitt wird erläutert, auf welche Weise eine Version $\mathcal{C}^{\mathrm{rv}}$ mit starren Variablen eines Grundkalküls $\mathcal{C}^{\mathrm{gd}}$ konstruiert werden kann.

Eine triviale Version $\mathcal{C}^{\mathrm{rv}}$ mit starren Variablen kann leicht wie folgt konstruiert werden: Wenn es eine Grund-Konklusion C_{gd} zu einer Prämisse Π_{gd} gibt, und eine Substitution ρ, die grundierend ist für eine Prämisse Π_{rv} mit starren Variablen, so daß $\Pi_{\mathrm{rv}}\rho = \Pi_{\mathrm{gd}}$, dann sei $\langle C_{\mathrm{gd}}, \rho\rangle$ eine Konklusion mit starren Variablen von Π_{rv}.

Betrachten wir folgendes Beispiel aus der Prädikatenlogik erster Stufe: Da

$$C_{\mathrm{gd}} = \{\{\mathsf{T}{:}p(a)\}, \{\mathsf{T}{:}q(a)\}\}$$

eine Grund-Konklusion von

$$\Pi_{\mathrm{gd}} = \{\mathsf{T}{:}(p(a) \vee q(a))\}$$

ist, und $\Pi_{\mathrm{rv}}\rho = \Pi_{\mathrm{gd}}$ für $\rho = \{X \mapsto a\}$ und

$$\Pi_{\mathrm{rv}} = \{\mathsf{T}{:}(p(X) \vee q(X))\}$$

gilt, folgernd wir, daß

$$\langle C_{\mathrm{rv}}, \rho\rangle = \langle\{\{\mathsf{T}{:}p(a)\}, \{\mathsf{T}{:}q(a)\}\}, \{X \mapsto a\}\rangle$$

eine Konklusion (mit starren Variablen) von Π_{rv} im Kalkül mit starren Variablen ist. Solche eine triviale Konklusion ist natürlich nicht, was wir wollen. Unsere Absicht ist, einen Kalkül mit starren Variablen zu konstruieren, bei dem die allgemeinere Konklusion

$$C'_{\mathrm{rv}} = \langle\{\{\mathsf{T}{:}p(X)\}, \{\mathsf{T}{:}q(X)\}\}, id\rangle$$

eine Konklusion mit starren Variablen der obigen Prämisse Π_{rv} ist.

Die allgemeine Idee des Hebens ist, (wiederholt) Konklusionen (oder Mengen von Konklusionen) der trivialen Version mit starren Variablen durch allgemeinere Konklusionen zu ersetzen – und zwar auf solche Art und Weise, daß Korrektheit und Vollständigkeit erhalten bleiben.

Satz 4.2.20 *Seien $\mathcal{C}$ und $\mathcal{C}'$ gutartige Kalküle mit starren Variablen für eine Logik* $\mathbf{L}$, *die von ihren Erweiterungsregeln $\mathcal{E}$ bzw. $\mathcal{E}'$ abgesehen identisch sind.*

Ferner sei $\mathcal{C}$ korrekt und vollständig, und für alle Signaturen $\Sigma \in Sig$ von $\mathbf{L}$, *alle Prämissen Π mit starren Variablen über Σ^*_{fv} und alle endlichen Menge W starrer Variablen gelte folgendes:*

1. *Wenn es zu einer Konklusion $\langle C', \tau'\rangle \in \mathcal{E}'(\Sigma)(\Pi)$ mit starren Variablen eine Grund-Konklusion C^{gd} über Σ^*_{gd} mit $\langle C', \tau'\rangle \leq^W \langle C^{\mathrm{gd}}, id\rangle$ gibt, dann gibt es eine Konklusion $\langle C, \tau\rangle \in \mathcal{E}(\Sigma)(\Pi)$ mit starren Variablen mit $\langle C, \tau\rangle \leq^W \langle C^{\mathrm{gd}}, id\rangle$* (Korrektheit).

2. *Für jede Konklusion $\langle C, \tau\rangle \in \mathcal{E}(\Sigma)(\Pi)$ gibt es $\langle C', \tau'\rangle \in \mathcal{E}'(\Sigma)(\Pi)$, so daß*

   $$\langle C', \tau'\rangle \leq^W \langle C, \tau\rangle$$

   (Vollständigkeit).

Dann ist auch der Kalkül $\mathcal{C}'$ korrekt und vollständig.

Der obige Satz beschreibt die Eigenschaften, die eine Menge von Konklusionen, die weniger allgemeine Konklusionen ersetzt, haben muß, damit Korrektheit und Vollständigkeit des Kalküls erhalten bleiben. Er beantwortet jedoch nicht die Frage, wie eine solche Menge allgemeinerer Konklusionen zu berechnen ist. Leider gibt es keine allgemeine Methode zur Konstruktion einer optimalen Menge, die nur allgemeinste

Konklusionen enthält. Es gibt allerdings eine Methode, die in den meisten Fällen gute Ergebnisse liefert; sie ist die Methode, die gewöhnlich (in informeller Weise) verwendet wird, wenn ein Grundkalkül in Ad-hoc-Manier gehoben wird. Die Idee ist, alle Vorkommen eines Terms in einer Grund-Prämisse Π^{gd} und einer ihrer Grund-Konklusionen C^{gd} durch (Platzalter für) beliebige Terme zu ersetzen, die starre Variablen enthalten und dann zu überprüfen, welche allgemeinsten Substitutionen angewendet werden müssen und welche andere Eigenschaften die neuen Terme haben müssen, um sicherzustellen, daß, benutzt man die entstehende Konklusion $\langle C^{\mathrm{rv}}, \tau\rangle$ mit starren Variablen als eine mögliche Konklusion für die entstehende Prämisse Π^{rv} mit starren Variablen, eine Erweiterungsregel entsteht, die die Bedingungen des Satzes 4.2.20 erfüllt, und also ein korrekter und vollständiger Kalkül ist.

Beispiel 4.2.21 Wir nehmen an, die Grund-Konklusion $\{\{\mathsf{T}{:}p(f(b))\}\}$ sei aus der Grund-Prämisse $\{\mathsf{T}{:}(a \approx b),\ \mathsf{T}{:}p(f(a))\}$ ableitbar, d. h., die Erweiterungsregel erlaube die „Anwendung“ von Gleichungen auf Termen in anderen Formeln.

Wir ersetzen die Termvorkommen in der Prämisse und der Konklusion durch (Platzhalter für) beliebige Terme mit starren Variablen, nämlich das Vorkommen von $f(a)$ durch t, das Vorkommen von a in $a \approx b$ durch t', das Vorkommen von b in $a \approx b$ durch s und das Vorkommen von $f(b)$ in der Konklusion durch s'.

Nun ist es leicht zu überprüfen, daß, verwendet man $\langle\{\{\mathsf{T}{:}p(s')\}\}, \tau\rangle$ als Konklusion der Prämisse $\{\mathsf{T}{:}(t' \approx s),\ \mathsf{T}{:}p(t)\}$, Korrektheit und Vollständigkeit gemäß Satz 4.2.20 erhalten bleiben, wenn τ ein Unifikator von t' und einem Teilterm t'' von t ist und s' der Term ist, den man erhält, wenn man t'' in t' durch $s\tau$ ersetzt. $\square$

In manchen Fällen kann eine allgemeinere Konklusion $\langle C', \tau'\rangle$ nicht verwendet werden, weil sie die Korrektheitsbedingung des Satzes 4.2.20 nur „ein wenig“ verletzt. Das heißt, für viele oder gar die meisten – aber eben nicht für alle – Grund-Konklusionen C^{gd} mit $\langle C', \tau'\rangle \leq^W \langle C^{\mathrm{gd}}, id\rangle$ gibt es eine (weniger allgemeine) Konklusion $\langle C, \tau\rangle$ mit $\langle C, \tau\rangle \leq^W \langle C^{\mathrm{gd}}, id\rangle$. Diesem Problem kann man beikommen, wenn entscheidbare, symbolische (d. h., syntaktische) *Constraints* formuliert werden können, die die „guten“ von den „schlechten“ Grund-Konklusionen C^{gd} trennen.

Die syntaktischen Objekte, aus denen symbolische Constraints bestehen, müssen in der erweiterten Signatur Σ^* enthalten sein, die benutzt wird, um Tableauformeln zu bilden. Das impliziert sofort, daß keine Constraints an (a) die Formeln eines initialen Tableaus und (b) die spezielle Tableauformel $\perp$ angeheftet werden können. Die Constraints werden zu einem Teil der Markierungen der Tableauformeln gemacht. Eine spezielle Menge von Tableauinterpretationen wird verwendet, um die Semantik eines Grundkalküls mit Constraints zu definieren; in diesen Interpretationen repräsentieren alle Markierungen, die einen Constraint enthalten, der zu false ausgewertet wird, eine spezielle Welt, in der alle Formeln falsch sind (so daß, falls eine Markierung σ zu false

ausgewertet wird, eine Tableauformel der Form $\sigma{:}\mathsf{T}{:}\phi$ von *keiner* Tableauinterpretation erfüllt wird und eine Tableauformel der Form $\sigma{:}\mathsf{F}{:}\phi$ von *jeder* Tableauinterpretation erfüllt wird).

Beispiel 4.2.22 Nehmen wir an, die Grund-Konklusion $\{\{\mathsf{T}{:}\phi(succ(pred(n)))\}\}$ könne aus einer Prämisse $\{\{\mathsf{T}{:}\phi(pred(succ(n)))\}\}$ abgeleitet werden, vorausgesetzt, der Term n kann nicht (syntaktisch) zu 0 reduziert werden. In diesem Fall ist es nicht korrekt, die Konklusion $\{\{\mathsf{T}{:}\phi(succ(pred(X)))\}\}$ mit starren Variablen aus der Prämisse $\{\{\mathsf{T}{:}\phi(pred(succ(X)))\}\}$ abzuleiten.

Wenn jedoch die Bedingung, daß n nicht (syntaktisch) zu 0 reduzierbar ist, als ein symbolischer Constraint $\{n \not\equiv 0\}$ formuliert werden kann, dann ist es für *alle* Terme n korrekt, die Grund-Konklusion $\{\{\mathsf{T}{:}\{n \not\equiv 0\}{:}\phi(succ(pred(n)))\}\}$ aus der Grund-Prämisse $\{\{\mathsf{T}{:}\phi(pred(succ(n)))\}\}$ abzuleiten; und es ist dementsprechend korrekt, die Konklusion $\{\{\mathsf{T}{:}\{X \not\equiv 0\}{:}\phi(succ(pred(X)))\}\}$ mit starren Variablen aus der Prämisse $\{\{\mathsf{T}{:}\phi(pred(succ(X)))\}\}$ mit starren Variablen abzuleiten. □

Symbolische Constraints, die an eine Tableauformel angeheftet werden, um die Korrektheit der Erweiterungsregel sicherzustellen, müssen klar unterschieden werden von solchen Constraints, die verwendet werden, um die Beweissuche zu steuern (wie beispielsweise Ordnungs-Constraints, die Auswahlfunktionen implementieren, siehe Abschnitt 5.5).

4.2.9 Semantik von Tableaus mit starren Variablen

Kalküle mit starren Variablen haben gewöhnlich keine auf Tableauinterpretationen beruhende Modellsemantik, weil die Wahrheitswerte verschiedener Tableauformeln, die dieselbe starre Variable enthalten, miteinander verknüpft ist. Nichtsdestotrotz ist es mögliche, eine Semantik für Tableaus mit starren Variablen zu definieren, basierend auf der Semantik der Grundversion eines Kalküls $\mathcal{C}^{\mathrm{rv}}$ mit starren Variablen; diese Semantik kann eingesetzt werden, um die Korrektheit von $\mathcal{C}^{\mathrm{rv}}$ zu beweisen, indem man zeigt, daß Erweiterungsregelanwendungen die Erfüllbarkeit erhalten. Zum Beweis der Vollständigkeit ist diese Semantik jedoch ungeeignet, weil die Vollständigkeitskriterien aus Definition 3.5.6 wie auch Satz 3.5.7 nur auf nicht-destruktive Kalküle anwendbar sind und Kalküle mit starren Variablen inhärent destruktiv sind.

Definition 4.2.23 Sei $\mathcal{C}^{\mathrm{rv}}$ ein gutartiger Kalkül mit starren Variablen für eine Logik $\mathbf{L}$, und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\mathrm{rv}}$ eine Version mit starren Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.

Ein Tableau T^{rv} mit starren Variablen über einer Signatur Σ^*_{rv} wird genau dann von einer Tableauinterpretation $\langle \mathbf{m}, I\rangle \in \mathit{TabInterp}(\Sigma^*_{\mathrm{gd}})$ von $\mathcal{C}^{\mathrm{gd}}$ *erfüllt*, wenn für alle Substitutionen $\tau \in \mathit{Subst}(\Sigma^*_{\mathrm{fv}})$, die T^{rv} grundieren, das Grund-Tableau $T^{\mathrm{gd}} = T^{\mathrm{rv}}\tau$ von $\langle \mathbf{m}, I\rangle$ erfüllt wird (Def. 3.4.1). □

Definition 4.2.24 Sei $\mathcal{C}^{\mathrm{rv}}$ ein gutartiger Kalkül mit starren Variablen für eine Logik $\mathbf{L}$, und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\mathrm{rv}}$ eine Version mit starren Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.

Der Kalkül $\mathcal{C}^{\mathrm{rv}}$ habe die *starke Korrektheitseigenschaft der Korrektheit der Erweiterung für Kalküle mit starren Variablen*, wenn für alle Signaturen Σ_{fv} und alle Tableaus T, T' über Σ^*_{fv} folgendes gilt: falls T' ein Nachfolgetableau von T ist, dann wird T' von allen Tableauinterpretationen in $TabInterp(\Sigma^*_{\mathrm{gd}})$ erfüllt, die T erfüllen. □

Lemma 4.2.25 *Sei $\mathcal{C}^{\mathrm{rv}}$ ein gutartiger Kalkül mit starren Variablen für eine Logik $\mathbf{L}$, und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\mathrm{rv}}$ eine Version mit starren Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.*

Der Kalkül $\mathcal{C}^{\mathrm{rv}}$ hat genau dann die starke Korrektheitseigenschaft der Korrektheit der Erweiterung für Kalküle mit starren Variablen (Def. 4.2.24), wenn $\mathcal{C}^{\mathrm{gd}}$ die starke Korrektheitseigenschaft der Korrektheit der Erweiterung für Grundkalküle hat (Eigenschaft 2 in Def. 3.5.8).

Beweis: Das Lemma folgt sofort aus den Definitionen der Korrektheitseigenschaften und der Beziehung zwischen einem Kalkül mit starren Variablen und seiner Grundversion. □

Satz 4.2.26 *Sei $\mathcal{C}^{\mathrm{rv}}$ ein gutartiger Kalkül mit starren Variablen für eine Logik $\mathbf{L}$, und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\mathrm{rv}}$ eine Version mit starren Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.*

Hat $\mathcal{C}^{\mathrm{rv}}$ die Korrektheitseigenschaft 1 aus Definition 3.5.8 (Angemessenheit der Menge der Tableauinterpretationen) und die starke Korrektheitseigenschaft der Korrektheit der Erweiterung für Kalküle mit starren Variablen (Def. 4.2.24), dann ist $\mathcal{C}^{\mathrm{rv}}$ korrekt.

Beweis: Weil $\mathcal{C}^{\mathrm{rv}}$ die Korrektheitseigenschaft 1 aus Definition 3.5.8 hat, hat auch $\mathcal{C}^{\mathrm{gd}}$ diese Eigenschaft, denn die initialen Tableaus für eine Menge $\mathfrak{F}$ von Formeln sind in beiden Kalkülen gleich. Da $\mathcal{C}^{\mathrm{rv}}$ die starke Eigenschaft der Korrektheit der Erweiterung für Kalküle mit starren Variablen hat, hat $\mathcal{C}^{\mathrm{gd}}$ die starke Eigenschaft der Korrektheit der Erweiterung für Grundkalküle (Eigenschaft 2 in Def. 3.5.8), was aus Lemma 4.2.25 folgt.

Also ist $\mathcal{C}^{\mathrm{gd}}$ korrekt (Satz 3.5.4), was impliziert, daß auch $\mathcal{C}^{\mathrm{rv}}$ korrekt ist (Korollar 4.2.19). □

4.2.10 Ein gutartiger Kalkül mit starren Variablen für PL1

In diesem Abschnitt wird als Beispiel ein gutartiger Kalkül $\mathcal{C}^{\mathrm{rv}}_{\mathrm{PL1}}$ mit starren Variablen für die Prädikatenlogik erster Stufe definiert, der eine Version mit starren Variablen des

Grundkalküls $\mathcal{C}^{\mathrm{gd}}_{\mathrm{PL1}}$ aus Abschnitt 3.6 ist und aus $\mathcal{C}^{\mathrm{gd}}_{\mathrm{PL1}}$ mit Hilfe der in Abschnitt 4.2.8 beschriebenen Technik des Hebens konstruiert werden kann.

Für jede prädikatenlogische Signatur Σ (siehe Abschnitt 2.3) enthält die erweiterte Signatur Σ^*_{fv} die freien Variablen in Var als Konstanten (siehe Abschnitt 2.3) und zusätzlich die Menge $F^{sko}(\Sigma)$ von Skolem-Funktionssymbolen, die unendlich viele Symbole jeder Stelligkeit $n \geq 0$ enthält. Folglich ist die Menge $TabForm(\Sigma^*_{\mathrm{fv}})$ eine Sprache um die Menge $Term(\Sigma^*_{\mathrm{fv}})$ von Termen (wobei alle freien Variablen und Terme von derselben Sorte sind); und es ist leicht nachzuvollziehen, daß

$$TabForm(\Sigma^*_{\mathrm{fv}}) = (TabForm(\Sigma^*_{\mathrm{gd}})^{\mathrm{fv}} \ ;$$

dabei bezeichnet Σ^*_{gd} die Erweiterung von Σ mit der Menge $F^{sko}(\Sigma)$ von Skolem-Funktionssymbolen.

Beispiel 4.2.27 Angenommen, a ist eine Konstante der Signatur Σ, die aber keine Skolem-Konstante ist, dann ist das Atom $p(a)$ eine Formel über Σ und den erweiterten Signaturen Σ^*_{gd} und Σ^*_{fv}. Ist c ist eine Skolem-Konstante, dann ist $p(c)$ ein Formel über Σ^*_{gd} und Σ^*_{fv} aber nicht über Σ. Die Atome $p(X)$ und $p(c, X)$ sind nur Formeln über Σ^*_{fv}. Man beachte, daß $p(c, x)$ *keine* Formel über irgendeiner der Signaturen ist, da Formeln keine freien Objektvariablen enthalten dürfen. $\square$

Die Menge der Markierungen und die initiale Markierung von $\mathcal{C}^{\mathrm{rv}}$ sind die gleichen wie die des Kalküls $\mathcal{C}^{\mathrm{gd}}$, d. h., die Markierung $*$ repräsentiert die einzelne Welt der PL1-Modelle.

Um die Erweiterungsregel von $\mathcal{C}^{\mathrm{rv}}$ zu definieren, benutzen wir wieder die vereinheitlichende Notation (unifying notation) wie für Grund-Tableauformeln, d. h., Tableauformeln mit starren Variablen werden in α-, β-, δ- und γ-Formeln eingeteilt (siehe Tabelle 3.1).

Die Erweiterungsregel-Schemata von $\mathcal{C}^{\mathrm{rv}}$ sind durch Heben der Regelschemata des Kalküls $\mathcal{C}^{\mathrm{gd}}$ konstruiert. Die Schemata für aus α-, β- und γ-Formeln bestehende Prämissen sind offensichtlich stabil bezüglich der Ersetzung von Termen durch Terme; neue starre Variablen werden durch Regelanwendungen auf Prämissen, die γ-Formeln enthalten, eingeführt.

Das Schema für Prämissen, die aus δ-Formeln bestehen, kann ebenfalls gehoben werden – vorausgesetzt eine geeignete Variante des Schemas wird verwendet. Wie sich leicht nachvollziehen läßt, ist die Variante des Schemas stabil, bei der alle Grundterme, die in der Formel $\delta(x)$ vorkommen, zu Argumenten des Skolem-Terms gemacht werden, der die gebundene Objektvariable x ersetzt. Also ist die Hauptschwierigkeit bei der Gestaltung eines gutartigen Kalküls für PL1 mit starren Variablen schon gemeistert, nämlich durch die Gestaltung eines stabilen Grund-Erweiterungsregel-Schemas für δ-Formeln.

Definition 4.2.28 Sei $\Sigma \in \mathit{Sig}_{\mathrm{PL1}}$ eine PL1-Signatur. Eine *Skolem-Term-Zuweisung mit freien Variablen* ist eine Funktion $\mathit{sko}_{\mathrm{fv}}$, die jeder δ-Formel $\phi \in \mathit{TabForm}_{\mathrm{PL1}}(\Sigma^*_{\mathrm{fv}})$ einen Term

$$\mathit{sko}_{\mathrm{fv}}(\phi) = f(X_1, \ldots, X_k) \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*_{\mathrm{fv}})$$

zuweist, so daß

1. (a) $f \in F^{\mathit{sko}}(\Sigma)$,
   (b) $k = \alpha^{\mathit{sko}}_{\Sigma}(f)$ und
   (c) $X_1, \ldots, X_k$ die in ϕ vorkommenden freien Variablen sind;
2. wenn ein $f' \in F^{\mathit{sko}}(\Sigma)$ in ϕ vorkommt, dann gilt $f > f'$, wobei $>$ eine beliebige aber feste Ordnung auf $F^{\mathit{sko}}(\Sigma)$ ist; und
3. für alle δ-Formeln $\psi \in \mathit{TabForm}_{\mathrm{PL1}}(\Sigma^*)$ gilt: wenn $\mathit{sko}(\psi) = f(X'_1, \ldots X'_k)$, dann sind die Formeln ϕ und ψ gleich bis auf Umbenennung gebundener Objektvariablen und die Ersetzung von Vorkommen freier Variablen X_i durch X'_i $(1 \leq i \leq k)$. □

Die Erweiterungsregel des Grundkalküls $\mathcal{C}^{\mathrm{gd}}$ ist nur für solche Prämissen instabil, die es erlauben einen Ast abzuschließen. Denn es wird vorausgesetzt, daß die Formeln in den komplementären Atomen $\mathsf{T}{:}{*}{:}G$ und $\mathsf{F}{:}{*}{:}G$ identisch sind, aus denen die minimale Prämisse für die Ableitung von $\perp$ besteht. Darum erlaubt das Regelschema mit starren Variablen für den Abschluß von Ästen, aus einer Prämisse $\Pi = \{\mathsf{T}{:}{*}{:}G,\ \mathsf{F}{:}{*}{:}G'\}$ die Konklusion $\langle\{\{\perp\}\}, \mu\rangle$ abzuleiten, wenn μ ein Unifikator von G und G' ist. Da es ausreicht, nur allgemeinste Konklusionen zu verwenden (Abschnitt 4.2.8), bleibt die Vollständigkeit des Kalküls erhalten, wenn man nur solche Konklusionen $\langle\{\{\perp\}\}, \mu\rangle$ zuläßt, bei denen μ ein *allgemeinster* Unifikator von G und G' ist.

In Tabelle 4.2 ist die Erweiterungsregel $\mathcal{E}^{\mathrm{rv}}_{\mathrm{PL1}}$ mit starren Variablen des Kalküls $\mathcal{C}^{\mathrm{rv}}_{\mathrm{PL1}}$ schematisch dargestellt; und im folgenden ist sie formal definiert.

Definition 4.2.29 Für alle Signaturen $\Sigma \in \mathit{Sig}_{\mathrm{PL1}}$ und alle Prämissen

$$\Pi \subset \mathit{TabForm}_{\mathrm{PL1}}(\Sigma^*_{\mathrm{rv}})$$

bestehe $\mathcal{E}(\Sigma)^{\mathrm{rv}}_{\mathrm{PL1}}(\Pi)$ aus den folgenden Konklusionen:

– $\{\langle\{\alpha_1, \alpha_2\}, \mathit{id}\rangle\}$ für alle $\alpha \in \Pi$,
– $\{\langle\{\beta_1\}, \{\beta_2\}, \mathit{id}\rangle\}$ für alle $\beta \in \Pi$,
– $\{\langle\{\gamma_1(X)\}, \mathit{id}\rangle\}$ für alle $\gamma \in \Pi$ und alle starren Variablen $X \in \mathit{Var}$,
– $\{\langle\{\delta(t)\}, \mathit{id}\rangle\}$ für alle $\delta \in \Pi$, wobei $t = \mathit{sko}_{\mathrm{fv}}(\delta)$ (Def. 4.2.28),
– $\{\langle\{\perp\}, \mu\rangle\}$ falls es $\mathsf{T}{:}\sigma{:}G$ und $\mathsf{F}{:}\sigma{:}G'$ in Π gibt, so daß G und G' unifizierbare Atome sind und μ ein MGU von G und G' ist.

□

$$\frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}} \qquad \frac{\beta}{\beta_1 \mid \beta_2} \qquad \frac{\gamma(x)}{\gamma_1(X)} \qquad \frac{\delta(x)}{\delta_1(t)}$$

für jede starre Variable Xt

wobei $t = sko_{\mathrm{fv}}(\delta)$ (siehe Def. 4.2.28)

$$\begin{array}{c}\phi\\ \overline{\psi}\\ \hline \bot\end{array}$$

wenn ϕ und ψ unifizierbare Atome sind
ein MGU von ϕ und ψ ist auf das Tableau anzuwenden

Tabelle 4.2: Regelschemata mit starren Variablen für PL1.

Der Kalkül $\mathcal{C}^{\mathrm{rv}}_{\mathrm{PL1}}$ ist eine Version mit starren Variablen des korrekten und vollständigen Grundkalküls $\mathcal{C}_{\mathrm{PL1}}$, der in Abschnitt 3.6 definiert wurde; daraus folgt, daß $\mathcal{C}^{\mathrm{rv}}_{\mathrm{PL1}}$ korrekt und vollständig ist (Korollar 4.2.19).

Satz 4.2.30 *Der Kalkül $\mathcal{C}^{\mathrm{rv}}_{\mathrm{PL1}}$ ist korrekt und vollständig.*

4.2.11 Ein gutartiger Kalkül mit starren Variablen für die Modallogik K

Als ein weiteres Beispiel definieren wir eine Version $\mathcal{C}^{\mathrm{rv}}_{\mathrm{K}}$ mit starren Variablen des Grundkalküls $\mathcal{C}^{\mathrm{con}}_{\mathrm{K}}$ für die Modallogik K aus Abschnitt 3.7.4, der eine Erweiterungsregel hat, die bezüglich ν-Formeln enthaltenden Prämissen stetig ist, und die darum für diese Prämissen hebbar ist.

Da $Form_{\mathrm{mod}}(\Sigma)$ keine Sprache mit Termen ist, werden starre Variablen nicht im Formelteil einer Tableauformel sondern in ihrer Markierung benutzt; die Menge der Markierungen von $\mathcal{C}^{\mathrm{con}}_{\mathrm{K}}$ ist eine Sprache mit Termen (wobei Terme natürliche Zahlen sind). Also ist $Lab_{\mathrm{fv}} = CondLab(\mathbb{N} \cup Var)$ die Menge der Markierungen von $\mathcal{C}^{\mathrm{rv}}_{\mathrm{K}}$ (wobei $CondLab(\mathbb{N} \cup Var)$ analog zu $CondLab(\mathbb{N})$ definiert ist, siehe Definition 3.7.1) mit der initialen Markierung 1. Mit dieser Menge Lab_{fv} von Markierungen gilt trivialerweise $TabForm_{\mathrm{fv}}(\Sigma_{\mathrm{mod}}) = (TabForm_{\mathrm{gd}}(\Sigma_{\mathrm{mod}}))^{\mathrm{fv}}$ (die Signaturen werden nicht durch zusätzliche [Skolem-]Symbole erweitert).

Wie in Kalkülen für die Prädikatenlogik erster Stufe ist das Erweiterungsregel-Schema des Grundkalküls offensichtlich stabil bzgl. Prämissen, die aus α- und β-Formeln bestehen. Die Rolle des Schemas für γ-Formeln spielt nun aber das Schema für ν-Formeln. Dieses Schema erlaubt es beispielsweise, aus einer Prämisse, die $\mathsf{T}{:}1{:}\Box G$ enthält, die Konklusion $\{\{\mathsf{T}{:}1.(n){:}p\}\}$ für *alle* $n \in \mathbb{N}$ abzuleiten. Das Schema kann gehoben werden, und seine Version mit starren Variablen erlaubt, aus Π die Konklusion $\{\{\mathsf{T}{:}1.(X){:}p\}\}$ für alle starren Variablen X abzuleiten.

Wenn ein Ast abgeschlossen wird, dann werden alle starren Variablen, die in den Markierungen der beteiligten komplementären Atome vorkommen, instantiiert, weil die Markierungen durch andere auf dem Ast vorkommende Markierungen gerechtfertigt werden müssen; und alle unbedingten Positionen in Markierungen sind grund (d. h., sie bestehen aus natürlichen Zahlen und nicht aus Variablen). Beispielsweise können die komplementären Atome $\phi_1 = \mathsf{T}{:}1.(X){:}p$ und $\phi_2 = \mathsf{F}{:}1.(Y){:}p$ nur zum Abschluß eines Astes verwendet werden, wenn der Ast Formeln wie z. B. $\psi = \mathsf{T}{:}1.1{:}q$ enthält, durch deren Markierungen (Instanzen der) Markierungen $1.(X)$ und $1.(Y)$ gerechtfertigt werden. Also wird die Konklusion $\langle\{\{\bot\}\}, \{X \mapsto 1, Y \mapsto 1\}$ aus der Prämisse $\{\phi_1, \phi_2, \psi\}$ abgeleitet.

In Tabelle 4.2 ist die Erweiterungsregel $\mathcal{E}^{\mathrm{rv}}_{\mathrm{mod}}$ mit starren Variablen des Kalküls $\mathcal{C}^{\mathrm{rv}}_{\mathrm{mod}}$ schematisch dargestellt; und sie ist im folgenden formal definiert.

Definition 4.2.31 Für alle Signaturen $\Sigma \in \mathit{Sig}_{\mathrm{mod}}$ und alle Prämissen

$$\Pi \subset \mathit{TabForm}_{\mathrm{mod}}(\Sigma)$$

bestehe $\mathcal{E}^{\mathrm{rv}}_{\mathrm{K}}(\Sigma)(\Pi)$ aus den folgenden Konklusionen (wobei $\lceil\cdot\rceil$ eine beliebige aber feste Bijektion von der Menge $\mathit{Form}_{\mathrm{mod}}(\Sigma)$ der modallogischen Formeln in die natürlichen Zahlen ist):

- $\langle\{\{\alpha_1, \alpha_2\}\}$ für alle $\alpha \in \Pi$,
- $\langle\{\{\beta_1\}, \{\beta_2\}\}, \mathit{id}\rangle$ für alle $\beta \in \Pi$,
- $\langle\{\{\mathsf{T}{:}\sigma.(X){:}G\}\}, \mathit{id}\rangle$ für alle $\mathsf{T}{:}\sigma{:}\Box G \in \Pi$ und alle $X \in \mathit{Var}$,
- $\langle\{\{\mathsf{F}{:}\sigma.(X){:}G\}\}, \mathit{id}\rangle$ für alle $\mathsf{F}{:}\sigma{:}\Diamond G \in \Pi$ und alle $X \in \mathit{Var}$,
- $\langle\{\{\mathsf{F}{:}\sigma.n{:}G\}\}, \mathit{id}\rangle$ für alle $\mathsf{F}{:}\sigma{:}\Box G \in \Pi$, wobei $n = \lceil\neg G\rceil$,
- $\langle\{\{\mathsf{T}{:}\sigma.n{:}G\}\}, \mathit{id}\rangle$ für alle $\mathsf{T}{:}\sigma{:}\Diamond G \in \Pi$, wobei $n = \lceil G\rceil$,
- $\langle\{\{\bot\}\}, \mu\rangle$ falls es $\mathsf{T}{:}\sigma{:}G, \mathsf{F}{:}\sigma'{:}G \in \Pi$ gibt und μ eine allgemeinste Substitution ist, so daß $[\sigma\mu] = [\sigma'\mu]$ und $\sigma\mu, \sigma'\mu$ durch $\Pi\mu$ gerechtfertigt sind.

□

Es kann verschiedene Möglichkeiten geben, die Variablen in einer Markierung zu instantiieren, so daß sie auf einem Tableauast gerechtfertigt ist.

Beispiel 4.2.32 Betrachten wir die Prämisse

$$\Pi = \{\mathsf{T}{:}1.1.1{:}q,\ \mathsf{T}{:}1.2{:}r,\ \mathsf{T}{:}1.(X){:}p\ \mathsf{F}{:}1.(X){:}p\}\ .$$

Die Menge $\mathcal{E}^{\mathrm{rv}}_{\mathrm{mod}}(\Pi)$ besteht aus den Konklusionen

$$\langle\{\{\bot\}\}, \{X \mapsto 1\}\rangle \text{ and } \langle\{\{\bot\}\}, \{X \mapsto 2\}\rangle\ .$$

□

$$\frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}} \qquad \frac{\beta}{\beta_1 \mid \beta_2} \qquad \frac{\mathsf{T}{:}\sigma{:}\Box G}{\mathsf{T}{:}\sigma.(X){:}G} \qquad \frac{\mathsf{F}{:}\sigma{:}\Diamond G}{\mathsf{F}{:}\sigma.(X){:}G}$$

$$\text{für alle } X \in \mathit{Var}$$

$$\frac{\mathsf{T}{:}\sigma{:}\Diamond F}{\mathsf{T}{:}\sigma.n{:}G} \qquad \frac{\mathsf{F}{:}\sigma{:}\Box G}{\mathsf{F}{:}\sigma.n{:}G} \qquad \frac{\mathsf{T}{:}\sigma{:}\neg G}{\mathsf{F}{:}\sigma{:}G} \qquad \frac{\mathsf{F}{:}\sigma{:}\neg G}{\mathsf{T}{:}\sigma{:}G}$$

$$\text{wobei } n = \lceil G \rceil \qquad \text{wobei } n = \lceil \neg G \rceil$$

$$\frac{\begin{array}{c}\mathsf{T}{:}\sigma{:}G\\ \mathsf{F}{:}\sigma'{:}G\end{array}}{\bot}$$

falls es eine Substitution μ gibt, so daß $[\sigma\mu] = [\sigma'\mu]$ und
diese Markierung durch Formeln auf dem Ast
nach Anwendung von μ gerechtfertigt ist;
eine allgemeinste solche Substitution μ ist auf das Tableau anzuwenden

Tabelle 4.3: Erweiterungsregel-Schemata des Kalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{rv}}$ mit starren Variablen für die Modallogik K

Der Kalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{rv}}$ ist korrekt und vollständig, denn er ist eine Version mit starren Variablen des Grundkalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ aus Abschnitt 3.7.4, der korrekt und vollständig ist.

Satz 4.2.33 *Der Kalkül $\mathcal{C}_{\mathrm{K}}^{\mathrm{rv}}$ für die Logik* K *ist korrekt und vollständig.*

Beispiel 4.2.34 Als ein Beispiel beweisen wir wieder die Unerfüllbarkeit der Formel G aus Beispielen 3.7.16 und 3.7.22, nun allerdings mit dem oben definierten Kalkül $\mathcal{C}_{\mathrm{K}}^{\mathrm{rv}}$ mit starren Variablen. Abbildung 4.1 zeigt ein geschlossenes Tableau $T_{\mathrm{rv}}^{\mathrm{con}}$ für G, das mit der Erweiterungsregel von $\mathcal{C}_{\mathrm{K}}^{\mathrm{con}}$ konstruiert ist (in der Abbildung sind die Tableauformeln mit uninstantiierten starren Variablen gezeigt; die Substitutionen, die während der Konstruktion des Tableaubeweises anzuwenden sind, sind separat aufgelistet).

Die Beweis hat die gleiche Struktur wie der in Abbildung 3.2 gezeigte, der mit Hilfe der Grundversion des Kalküls konstruiert ist. Der Unterschied ist, daß bei der Anwendung des Erweiterungsregel-Schemas für ν-Formeln, um die Formeln 6, 7 und 19 zum Tableau hinzuzufügen, die neu eingeführten Markierungen nicht „geraten" werden müssen. Statt dessen werden die starre Variablen enthaltenden Markierungen $1.(X_1)$, $1.(X_2)$ bzw.$1.(X_3)$ eingeführt. Die starren Variablen werden später instantiiert, wenn die Erweiterungsregel angewendet wird, um den linken Ast des Tableaus abzuschließen.

Beim Abschluß des linken Astes reicht es nicht aus, die Substitution $\{X_1 \mapsto X_2\}$ anzuwenden, d. h., $\langle\{\{\bot\}\}, \{X_1 \mapsto X_2\}\rangle$ ist keine gültige Konklusion irgendeiner Prämis-

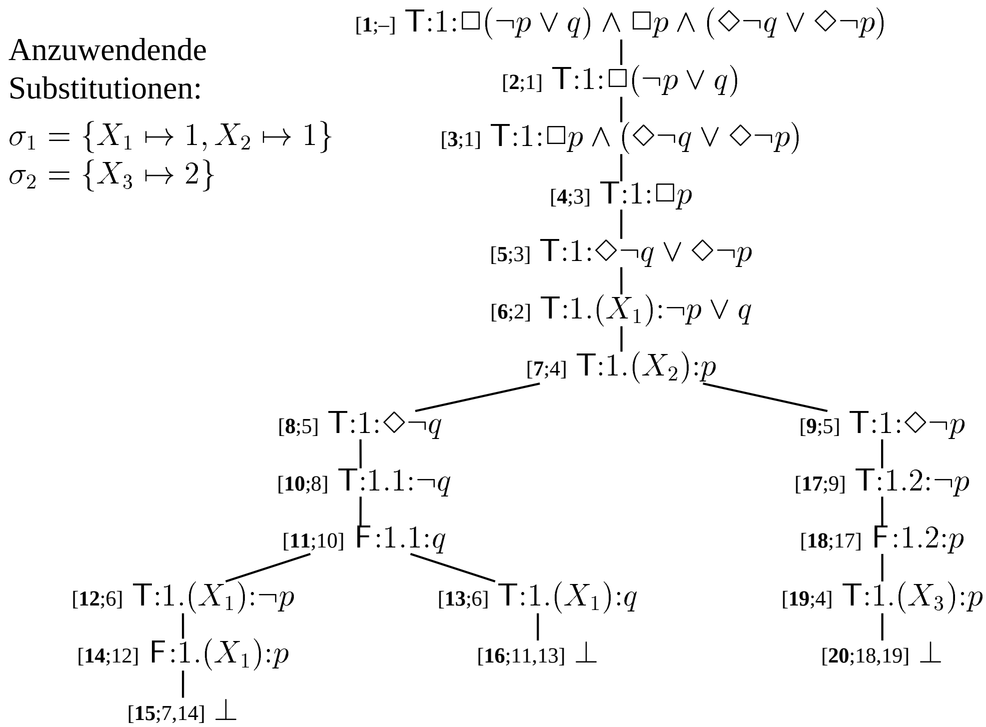


Abbildung 4.1: Das Tableau $T^{\mathrm{con}}_{\mathrm{rv}}$ aus Beispiel 4.2.34 ist das Ergebnis der Anwendung der aufgeführten Substitutionen auf den obigen Baum.

se auf dem linken Ast des Tableaus, – denn die Markierung $1.(X_2)$ ist nicht gerechtfertig. Sowohl X_1 als auch X_2 müssen mit 1 instantiiert werden, so daß die gerechtfertigte Markierung $1.(1)$ entsteht.

Nachdem $\sigma_1 = \{X_1 \mapsto 1, X_2 \mapsto 1\}$ auf das Tableau angewendet worden ist, erfordert es keine weiteren Instantiierungen starrer Variablen, um den mittleren Ast abzuschließen. Um den rechten Ast abzuschließen muß dann allerdings die Substitution σ_2 angewendet werden, die X_3 mit 2 instantiiert.

Dieses Beispiel zeigt die Vorteile der Verwendung starrer Variablen. Wenn die Äste geschlossen werden, gibt es (in diesem Beispiel) jeweils nur eine allgemeinste Substitution, die angewendet werden kann, d. h., in jedem Fall ist die Wahl der Instantiierungen der freien Variablen deterministisch. Im Gegensatz dazu müssen die Markierungen tatsächlich geraten werden, wenn die Grundversion des Kalküls verwendet wird; es ist nicht offensichtlich, ob die Markierung $1.(1)$ oder die Markierung $1.(2)$ eingeführt werden sollte, was der Wahlmöglichkeit entspricht, jede der starren Variablen entweder mit 1 oder 2 zu instantiieren. $\Box$

4.3 Kalküle mit universellen Variablen

4.3.1 Die Idee eines Kalküls mit universellen Variablen

Unter bestimmten Bedingungen gibt es noch eine andere Möglichkeit als die Verwendung starrer Variablen, um einen Kalkül zu verstärken. Statt eine freie Variable zu verwenden, um einen einzelnen, unbekannten Term zu repräsentieren, kann man sie auch dazu verwenden, als Platzhalten für *alle* Terme zu stehen. Dann repräsentiert eine Formel, die eine solche freie Variable $\boldsymbol{x}$ enthält, für *alle* Formeln, die man dadurch erhält, daß man $\boldsymbol{x}$ durch irgendeinen Term ersetzt. Intuitiv kann man diese Variablen als auf de Meta-Eben universell quantifiziert betrachten; darum heißen sie *universelle Variablen*. Um die starren und die universellen Variablen klar zu trennen, verwenden wird im folgenden die Variablen in der Menge Var ausschließlich als starre Variablen; universelle Variablen stammen immer aus der separaten Menge $UVar = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots\}$, die von $Var = \{X_1, X_2, \ldots\}$ disjunkt ist. Universelle Variablen werden niemals instantiiert; auf Tableaus, die universelle (und keine starren) Variablen enthalten werden keine Substitutionen angewendet.

Definition 4.3.1 Sei $\mathbf{L}$ eine Logik. Ein Kalkül $\mathcal{C}^{\mathrm{uv}}$ für $\mathbf{L}$ mit freien Variablen heiße ein *Kalkül mit universellen Variablen* (für $\mathbf{L}$), wenn sein Tableauformeln nur freie Variablen aus der Menge $UVar$ und keine aus der Menge Var enthalten. □

Definition 4.3.2 Sei $\mathcal{C}^{\mathrm{uv}}$ ein Kalkül mit universellen Variablen für eine Logik $\mathbf{L}$; sei $\Sigma \in Sig$ eine Signatur von $\mathbf{L}$; und sei $\Phi \subset TabForm(\Sigma^*_{\mathrm{fv}})$ eine Menge von Tableauformeln.

Die Menge $Inst(\Phi) \subset TabForm(\Sigma^*_{\mathrm{gd}})$ der *Instanzen* der Formeln in Φ ist wie folgt definiert:

$$Inst(\Phi) = \{\phi\tau \mid \tau \in Subst^{\mathrm{fv}}(\Sigma^*) \text{ ist für } \phi \text{ grundierend}\} \ .$$

□

Gutartigkeit und andere syntaktische Begriffe wie Konklusion und Erweiterungsregel sind für Kalküle mit universellen Variablen in gleicher Weise wie für Grundkalküle definiert.

Der Vorteil universeller Variablen ist folgender: Häufig ist es notwendig, mehrere verschiedene Instanzen einer freie Variablen enthaltenden Tableauformel zu verwenden, um einen Ast oder ein Teiltableau abschließen zu können. In Kalkülen mit starren Variablen ist der Mechanismus dafür, die Erweiterungsregel mehrfach auf die Prämissen anzuwenden, die die Einführung neuer starrer Variablen gestatten, um so Varianten der Tableauformeln zu erzeugen. Starre Variablen sind *nicht* implizit universell quantifiziert (wie beispielsweise die Variablen in den Klauseln eines Resolutionskalküls).

Nehmen wir an, ein Tableauast enthalte eine Formel $\phi(X)$; und nehmen wir weiter an, die Erweiterung des Tableaus werde mit der Erzeugung neuer Äste fortgesetzt. Einige dieser neuen (Teil-)Äste enthalten Vorkommen der starren Variablen X; wenn X instantiiert wird, muß die gleiche Ersetzung für X auf all diesen Ästen vorgenommen werden. In bestimmten Situationen ist es jedoch möglich – ohne die Korrektheit des Kalküls zu beeinträchtigen –, die Formel $\phi(\boldsymbol{x})$ zu B hinzuzufügen. In solch einem Fall können verschiedene Instanzen $\phi(t)$ von $\phi(\boldsymbol{x})$ verwendet werden, um den Ast B zu erweitern – ohne zuerst Varianten $\phi(X'), \phi(X''), \ldots$ von $\phi(X)$ zu generieren. Wenn man solche Situationen erkennt und ausnutzt, führt das zu kürzeren Tableaubeweisen und in den meisten Fällen zu einem kleineren Suchraum. Wenn sowohl universelle als auch starre Variablen in einem Kalkül verwendet werden, dann können in den Konklusionen mit starren Variablen Substitutionen benutzt werden, die allgemeiner sind als in dem entsprechenden Kalkül, der nur starre Variablen verwendet.

Wenn ein Ast B eine Tableauformel $\phi(\boldsymbol{x})$ enthält, bedeutet das intuitiv, daß man für beliebige Terme t die Formel $\phi(t)$ zu B hinzufügen könnte, ohne dabei neue offene Äste zu erzeugen (diese intuitive Interpretation ist allerdings nur richtig, wenn keine Information in der Struktur eines Tableauastes verborgen ist, d. h., wenn der Kalkül gutartig ist).

Beispiel 4.3.3 Abbildung 4.2 zeigt ein Beispiel für die Nützlichkeit universellerer Variablen. Das Tableau T_1^{rv} (oben links in der Abbildung) für die Menge

$$\mathfrak{F} = \{(\forall x)(p(x)),\ \neg p(a) \vee \neg p(b)\}$$

von PL1-Formeln kann nicht sofort geschlossen werden, da es keine einzelne Ersetzung für X gibt, die es erlauben würde, $\perp$ zu beiden Ästen hinzuzufügen. Um einen Beweis zu finden, muß die Erweiterungsregel noch einmal auf die γ-Formel $\mathsf{T}{:}(\forall x)(p(x))$ angewendet werden, um eine Variante $\phi(X') = \mathsf{T}{:}p(X')$ der Formel $\phi(X) = \mathsf{T}{:}p(X)$ zu erzeugen. Dann kann das geschlossene Tableau T_2^{rv} (oben rechts in der Abbildung) abgeleitet werden.

Das Tableau T_1^{uv} (unten links in der Abbildung), das die Formel $\mathsf{T}{:}p(\boldsymbol{x})$ statt der Formel $\mathsf{T}{:}p(X)$ enthält, kann dagegen zu dem geschlossenen Tableau T_2^{uv} (unten rechts in der Abbildung) erweitert werden, ohne daß eine Substitution angewendet wird, denn $\mathsf{T}{:}p(\boldsymbol{x})$ steht für alle Formeln der Form $\mathsf{T}{:}p(t)$, einschließlich $\mathsf{T}{:}p(a)$ und $\mathsf{T}{:}p(b)$. $\square$

Ein zusätzlicher Vorteil universeller Variablen ist, daß sie helfen, Redundanzen zu vermeiden, die in Kalkülen mit starren Variablen inhärent vorhanden sind. Wenn beispielsweise ein Tableauast mit starren Variablen die Formeln

$$\begin{aligned} \phi(X_1) &= \mathsf{T}{:}p(X_1) \\ \phi(X_2) &= \mathsf{T}{:}p(X_2) \\ \psi_1 &= \mathsf{F}{:}p(a) \\ \psi_2 &= \mathsf{F}{:}p(b) \end{aligned}$$

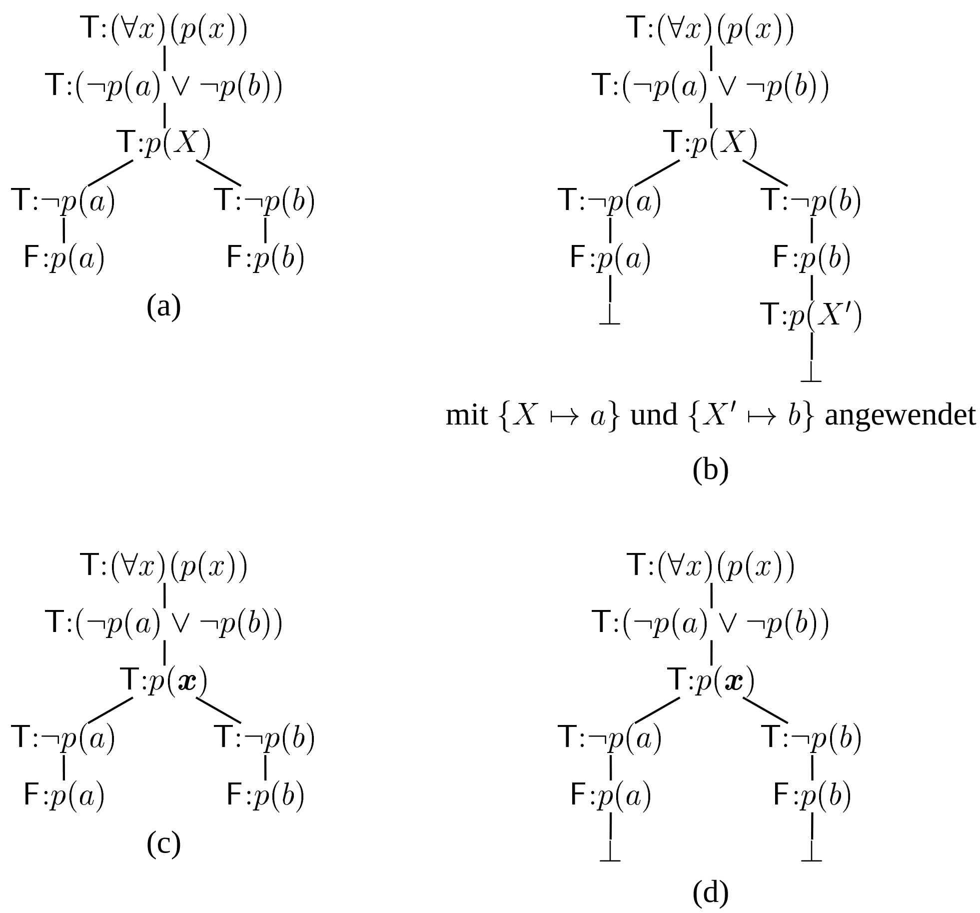


Abbildung 4.2: Beispiel für die Nützlichkeit universeller Variablen; die Tableaus (a) T_1^{rv}, (b) T_2^{rv}, (c) T_1^{uv} und (d) T_2^{uv} aus Beispiel 4.3.3.

enthält, dann gibt es vier verschiedene Möglichkeiten, den Ast zu schließen. Enthält der Ast jedoch die Formel $\phi(\boldsymbol{x}) = \mathsf{T}{:}p(\boldsymbol{x})$ mit einer universellen Variablen statt $\phi(X_1)$ und $\phi(X_2)$, dann gibt es nur eine Konklusion, die den Ast schließt, nämlich $\{\{\bot\}\}$, und es ist nicht nötig, dafür eine Variable zu instantiieren.

Universelle Variablen können in Tableaus beliebig umbenannt werden, vorausgesetzt alle Vorkommen einer Variablen in derselben Tableauformel werden durch die gleiche neue Variable ersetzt. Darum ist es möglich, mit einer Prämisse, die die beiden komplementären Atome $\mathsf{T}{:}p(\boldsymbol{x})$ und $\mathsf{F}{:}p(f(\boldsymbol{x}))$ enthält, einen Ast abzuschließen, weil die universelle Variable $\boldsymbol{x}$ in einem der beiden Atome umbenannt werden kann.

Die Technik, in Tableaukalkülen universelle Variablen zu verwenden, wurde für die klassische Prädikatenlogik erster Stufe zuerst in (Beckert & Hähnle, 1992) beschrieben und ist in (Beckert & Hähnle, 1998) weiter verbessert worden. Ein Tableaukalkül mit universellen Variablen für Modallogiken ist in (Beckert & Goré, 1997) beschrieben. In (Bibel, 1982) ist eine *Splitting by need* genannte Technik für die Konnektionsmethode vorgeschlagen; sie basiert – wie die Methode der universellen Variablen – auf der Idee, das Kopieren universell quantifizierter Formeln in Fällen zu vermeiden, in denen es korrekt ist, eine einzelne Kopie mit verschiedenen Instantiierungen für ihre Variablen zu verwenden.

4.3.2 Die universelle Variablen verwendende Version eines Grundkalküls

Kalküle $\mathcal{C}^{\mathrm{uv}}$ mit universellen Variablen werden gewöhnlich durch Heben eines Grundkalküls $\mathcal{C}^{\mathrm{gd}}$ konstruiert (ähnlich wie Kalküle mit starren Variablen), so daß Korrektheit und Vollständigkeit von $\mathcal{C}^{\mathrm{uv}}$ aus Korrektheit und Vollständigkeit von $\mathcal{C}^{\mathrm{gd}}$ folgen.

Definition 4.3.4 Sei $\mathcal{C}^{\mathrm{uv}}$ ein gutartiger universelle Variablen verwendender Kalkül für eine Logik $\mathbf{L}$; und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß für alle Signaturen $\Sigma \in \mathit{Sig}$ die erweiterte Signatur Σ^*_{gd}, die von $\mathcal{C}^{\mathrm{gd}}$ benutzt wird, diejenige Signatur ist, die gemäß der Definition von Tableaukalkülen mit freien Variablen (Def. 4.2.3) existieren muß, für die

$$\mathit{TabForm}(\Sigma^*_{\mathrm{fv}}) = (\mathit{TabForm}(\Sigma^*_{\mathrm{gd}}))^{\mathrm{fv}}$$

gilt.

Der Kalkül $\mathcal{C}^{\mathrm{uv}}$ ist eine *Version mit universellen Variablen* von $\mathcal{C}^{\mathrm{gd}}$ (und $\mathcal{C}^{\mathrm{gd}}$ ist eine *Grundversion* von $\mathcal{C}^{\mathrm{uv}}$), wenn für alle Prämissen $\Pi_{\mathrm{uv}} \subset \mathit{TabForm}(\Sigma^*_{\mathrm{fv}})$ mit universellen Variablen die Mengen

$$\bigcup \{\mathcal{E}^{\mathrm{gd}}(\Pi_{\mathrm{gd}}) \mid \Pi_{\mathrm{gd}} \text{ ist eine endliche Teilmenge von } \mathit{Inst}(\Pi_{\mathrm{uv}})\}$$

und

$$\{\{E_1\tau_1, \ldots, E_n\tau_n\} \mid \{E_1, \ldots, E_n\} \in \mathcal{E}^{\mathrm{uv}}(\Pi_{\mathrm{uv}}), \text{ und für } 1 \le i \le n, \\ \text{ist } \tau_i \in \mathit{Subst}^{\mathrm{fv}}(\Sigma^*_{\mathrm{fv}}) \text{ eine für } E_i \text{ grundierende Substitution}\}$$

gleich sind. □

Der folgende Satz stellt Korrektheit und Vollständigkeit eines gutartigen Kalküls mit universellen Variablen zu Korrektheit und Vollständigkeit seiner Grundversion in Beziehung. Der Beweis des Satzes ist konstruktiv und stellt also einen Algorithmus zur Verfügung, mit dem ein Grund-Tableaubeweis aus einem Tableaubeweis mit universellen Variablen berechnet werden kann. Da der Grund-Tableaubeweis exponentiell größer sein kann, ist eine Beweisrichtung nicht ganz einfach; es gibt keine eindeutige Zuordnung zwischen Erweiterungsregelanwendungen im Grundkalkül und im Kalkül mit universellen Variablen (in Beispiel 4.3.7 wird die Transformation veranschaulicht).

Satz 4.3.5 *Sei $\mathcal{C}^{\mathrm{uv}}$ ein gutartiger Kalkül mit universellen Variablen für eine Logik* $\mathbf{L}$*; und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für* $\mathbf{L}$*, so daß $\mathcal{C}^{\mathrm{uv}}$ eine Version mit universellen Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist (Def. 4.3.4).*

Es gibt für eine Formelmenge $G \subset \mathit{Form}(\Sigma)$ über einer Signatur $\Sigma \in \mathit{Sig}$ genau dann einen $\mathcal{C}^{\mathrm{uv}}$-Tableaubeweis, wenn es einen $\mathcal{C}^{\mathrm{gd}}$-Tableaubeweis für G gibt.

Beweis: *Wenn-Teil:* Sei $T_1^{\mathrm{gd}}, \ldots, T_n^{\mathrm{gd}}$ $(n \geq 1)$ ein mit dem Grundkalkül $\mathcal{C}^{\mathrm{gd}}$ konstruierter Tableaubeweis für G.

Durch Induktion über i zeigen wir, daß es eine Folge $T_1^{\mathrm{uv}}, \ldots, T_n^{\mathrm{uv}}$ von Tableaus mit universellen Variablen für G gibt, so daß für jeden Ast B^{uv} von T_i^{uv} ein Ast B^{gd} von T_i^{gd} mit

$$\mathit{Form}(B^{\mathrm{gd}}) \subset \mathit{Inst}(\mathit{Form}(B^{\mathrm{uv}}))$$

existiert. Daraus folgt dann, daß alle Äste des geschlossenen Tableaus T_n^{gd} die Tableauformel $\perp$ enthalten. Also ist auch T_n^{uv} geschlossen.

$i = 1$: Ein (beliebiges) initiales Tableau T_1^{uv} für G enthält keine universellen Variablen; darum gilt $\mathit{Form}(B^{\mathrm{gd}}) = \mathit{Form}(B^{\mathrm{uv}}) = \mathit{Inst}(\mathit{Form}(B^{\mathrm{uv}}))$.

$i \to i+1$: Sei B_i^{gd} der Ast von T_i^{gd}, der unter Verwendung einer Prämisse Π und einer Konklusion $C^{\mathrm{gd}} = \{E_1^{\mathrm{gd}}, \ldots, E_k^{\mathrm{gd}}\}$ erweitert worden ist. Gemäß der Definition der Beziehung zwischen einem Kalkül mit universellen Formeln und seiner Grundversion gibt es auf jedem Ast B_i^{uv} mit $\mathit{Form}(B_i^{\mathrm{gd}}) \subset \mathit{Inst}(\mathit{Form}(B_i^{\mathrm{uv}}))$ eine Prämisse Π^{uv} aus der eine Konklusion $C^{\mathrm{uv}} = \{E_1^{\mathrm{uv}}, \ldots, E_k^{\mathrm{uv}}\}$ mit universellen Variablen abgeleitet werden kann, so daß $E_j^{\mathrm{gd}} \subset \mathit{Inst}(E_j^{\mathrm{uv}})$ $(1 \leq j \leq k)$. Das Tableau T_{i+1}^{uv} sei aus T_i^{uv} konstruiert, indem jeder solche Ast unter Verwendung der Prämisse Π^{uv} und der Konklusion C^{uv} erweitert wird.

Nun sei B_{i+1}^{uv} ein beliebiger Ast in T_{i+1}^{uv}. Der einzige interessante Fall ist, wenn

$$\mathit{Form}(B_{i+1}^{\mathrm{uv}}) = \mathit{Form}(B_i^{\mathrm{uv}}) \cup E_j^{\mathrm{uv}}$$

für ein $j \in \{1, \ldots, k\}$ gilt. Dann erfüllt der Ast B^{gd}_{i+1} in T^{gd}_{i+1}, der durch Erweiterung von B^{gd}_i mit E^{gd}_j entstanden ist, die Bedingung der Induktionsannahme, weil $\mathit{Form}(B^{\mathrm{gd}}_i) \subset \mathit{Inst}(\mathit{Form}(B^{\mathrm{uv}}_i))$ und also

$$\begin{aligned}
\mathit{Form}(B^{\mathrm{gd}}_{i+1}) &= \mathit{Form}(B^{\mathrm{gd}}_i) \cup E^{\mathrm{gd}}_j \\
&\subset \mathit{Inst}(\mathit{Form}(B^{\mathrm{uv}}_i)) \cup \mathit{Inst}(E^{\mathrm{uv}}_j) \\
&= \mathit{Inst}(\mathit{Form}(B^{\mathrm{uv}}_i) \cup E^{\mathrm{uv}}_j) \\
&= \mathit{Inst}(\mathit{Form}(B^{\mathrm{uv}}_{i+1})) \ .
\end{aligned}$$

Genau-dann-Teil: Sei $T^{\mathrm{uv}}_1, \ldots, T^{\mathrm{uv}}_n$ $(n \geq 1)$ ein Tableaubeweis für G, der mit Hilfe des universelle Variablen verwendenden Kalküls $\mathcal{C}^{\mathrm{uv}}$ konstruiert ist.

Durch Induktion über i – mit Induktionsanfang $i = n$ und bei jedem Schritt abnehmendem i – zeigen wir:

Induktionsannahme: Für jeden Ast B^{uv}_i in T^{uv}_i existiert eine Menge $\Phi^{\mathrm{gd}} \subset \mathit{Inst}(B^{\mathrm{uv}}_i)$ von Grund-Tableauformeln, so daß jeder Tableauast B^{gd}, der diese Formeln enthält, zu einem geschlossenen (Teil-)Tableau erweitert werden kann.

Daß die Induktion mit $i = n$ beginnt und i in jedem Schritt verringert wird, spiegelt die Tatsache wieder, daß man als allererstes wissen muß, welche Instanzen von universelle Variablen enthaltenden Formeln man als Prämissen benötigt, um die Blätter des Grund-Tableaus abzuleiten, was dann erlaubt, die Instanzen in den Prämissen der Prämissen zu bestimmen, usw.

Nachdem bewiesen ist, daß die Induktionsannahme für $i = 1$ gilt, kann man folgern, daß der einzelne Ast B^{gd}_1 eines initialen Grundtableaus zu einem geschlossenen Grundtableau erweitert werden kann, denn es gilt

$$\mathit{Form}(B^{\mathrm{gd}}_1) = \mathit{Form}(B^{\mathrm{uv}}_1) = \mathit{Inst}(\mathit{Form}(B^{\mathrm{uv}}_1)) \ ,$$

wobei B^{uv}_1 der einzelne Ast des initialen Tableaus T^{uv}_1 mit universellen Variablen ist.

$i = n$: Da das Tableau T^{uv}_n geschlossen ist, enthält jeder seiner Äste B^{uv}_n die Tableauformel $\bot$; darum kann Φ^{gd} als die Menge $\{\bot\}$ festgesetzt werden. Jeder Grund-Tableauast, der $\bot$ enthält, kann trivialerweise zu einem geschlossenen Teiltableau erweitert werden.

$i{+}1 \to i$: Sei B^{uv}_i ein beliebiger Ast von T^{uv}_i. Falls B^{uv}_i auch ein Ast von T^{uv}_{i+1} ist, sind wir fertig. Andernfalls ist T^{uv}_{i+1} aus T^{uv}_i durch Erweiterung des Astes B^{uv}_i konstruiert worden – unter Verwendung einer Prämisse $\Pi^{\mathrm{uv}} \subset \mathit{Form}(B^{\mathrm{uv}}_i)$ und einer Konklusion $C^{\mathrm{uv}} = \{E^{\mathrm{uv}}_1, \ldots, E^{\mathrm{uv}}_k\}$ von Π^{uv}.

Sei $\Phi^{\mathrm{gd}}_{i+1,j} \subset \mathit{Form}(B^{\mathrm{uv}}_{i+1,j})$ diejenige Menge von Tableauformeln, die gemäß der Induktionsannahme für den Ast $B^{\mathrm{uv}}_{i+1,j}$ von T^{uv}_{i+1} existiert, der durch Erweiterung von B^{uv}_i

mit E_j^{uv} entstanden ist ($1 \leq j \leq k$). Weiterhin seien für alle $j \in \{1, \dots, k\}$ Substitutionen $\tau_1^j, \dots, \tau_{l_j}^j$, die für E_j^{uv} grundierend sind, so gewählt, daß alle Tableauformeln in $Inst(E_j^{\mathrm{uv}})$, die in $\Phi_{i+1,j}^{\mathrm{gd}}$ vorkommen, Elemente der Vereinigung $\bigcup_{r=1}^{l_j} E_j^{\mathrm{uv}} \tau_r^j$ sind.

Und weiter sei $\Phi_i^{\mathrm{gd}} \subset Inst(Form(B_i^{\mathrm{uv}}))$ die kleinste Menge, die die folgenden Grund-Tableauformeln enthält:

1. alle Grund-Tableauformeln in $Inst(Form(B_i^{\mathrm{uv}}))$, die in einer der Mengen $\Phi_{i+1,j}^{\mathrm{gd}}$ enthalten sind ($1 \leq j \leq k$),

2. alle Grund-Tableauformeln in $Inst(Form(B_i^{\mathrm{uv}}))$, die in solchen minimalen Prämissen $\Pi^{\mathrm{gd}} \subset Inst(Form(B_i^{\mathrm{uv}}))$ vorkommen, die erlauben, eine der Konklusionen

$$C^{\mathrm{gd}} = \{E_1^{\mathrm{uv}} \tau_{r_1}^1, \dots, E_k^{\mathrm{uv}} \tau_{r_k}^1\}$$

   abzuleiten, wobei $r_j \in \{1, \dots, l_j\}$ für $1 \leq j \leq k$. Solche Prämissen Π^{gd} existieren als Teilmengen von $Inst(Form(B_i^{\mathrm{uv}}))$ gemäß der Definition der Beziehung zwischen einem Kalkül mit universellen Variablen und seiner Grundversion.

Nun kann jeder Grund-Tableauast B_i^{gd} mit $\Phi_i^{\mathrm{gd}} \subset Form(B_i^{\mathrm{gd}})$ wie folgt zu einem geschlossenen Teiltableau erweitert werden: Der Ast B_i^{gd} werde (wiederholt) unter Verwendung *aller* der Konklusionen $C^{\mathrm{gd}} = \{E_1^{\mathrm{uv}} \tau_{r_1}^1, \dots, E_k^{\mathrm{uv}} \tau_{r_k}^1\}$ in solcher Weise erweitert, daß in der Konstruktion jedes der neu entstehenden Teiläste jede dieser Konklusionen einmal verwendet worden ist. Die Anzahl der Konklusionen ist $\prod_{j=1}^k l_j$; damit ist die Zahl neuer Teiläste exponentiell in $\prod_{j=1}^k l_j$. Aus dem Schubfachprinzip folgt, daß jeder der neuen Teiläste für zumindest ein $j \in \{1, \dots . k\}$ die Formeln *aller* Extensionen $E_j^{\mathrm{uv}} \tau_r^j$ enthält (also für *alle* $r \in \{1, \dots, l_j\}$); denn sonst müßte es einen Ast geben, der für jedes $j \in \{1, \dots, k\}$ zumindest eine Extensionen $E_j^{\mathrm{uv}} \tau_{r_j}^j$ *nicht* enthält – im Widerspruch zur Konstruktionsvorschrift, daß alle Konklusionen $\{E_1^{\mathrm{uv}} \tau_{r_1}^1, \dots, E_k^{\mathrm{uv}} \tau_{r_k}^1\}$ an der Erweiterung aller neuen Teiläste beteiligt sind.

Es folgt, daß jeder der neuen Teiläste für zumindest ein $j \in \{1, \dots, k\}$ alle Tableauformeln in $\Phi_{i+1,j}^{\mathrm{gd}}$ enthält, denn eine Formel

$$\Phi_{i+1,j}^{\mathrm{gd}} \subset Inst(Form(B_{i+1,j}^{\mathrm{uv}})) = Inst(Form(B_i^{\mathrm{uv}})) \cup Inst(E_j^{\mathrm{uv}})$$

ist entweder Element von $Inst(Form(B_i^{\mathrm{uv}}))$ – dann kommt sie gemäß Bedingung 1 in der Definition von Φ_i^{gd} in Φ_i^{gd} und also in B_i^{gd} vor –, oder sie ist ein Element von $Inst(E_j^{\mathrm{uv}})$ und kommt daher gemäß Konstruktion auf dem neuen Teilast vor.

Die Induktionsannahme trifft auf jeden der neuen Teiläste zu, denn jeder von ihnen enthält die Formeln einer der Mengen $\Phi_{i+1,j}^{\mathrm{gd}}$; sie können daher alle zu einem geschlossenen Teiltableau erweitert werden. Also kann der Ast B_i^{gd} zu einem geschlossenen Teiltableau erweitert werden. $\square$

Korollar 4.3.6 *Sei $\mathcal{C}^{\mathrm{uv}}$ ein gutartiger Kalkül mit universellen Variablen für eine Logik $\mathbf{L}$; und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\mathrm{uv}}$ eine Version mit universellen Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist (Def. 4.3.4).*

Der Kalkül $\mathcal{C}^{\mathrm{uv}}$ ist genau dann korrekt und vollständig, wenn $\mathcal{C}^{\mathrm{gd}}$ korrekt und vollständig ist.

Beispiel 4.3.7 Da die aus Beweisen mit universellen Variablen konstruierten Grund-Tableaubeweise exponentiell größer sind, können wir nur ein kleines Beispiel für einen Schritt in der Konstruktion vorstellen.

Wir betrachten den in Abbildung 4.3 gezeigten Tableaubeweis, der universelle Variablen verwendet (er ist mit dem Kalkül $\mathcal{C}^{\mathrm{uv}}_{\mathrm{PL1}}$ für PL1 aus Abschnitt 4.3.6 konstruiert); und wir nehmen an, die Teiltableaus T^{uv}_p und T^{uv}_q seien geschlossen (die Menge Γ der Tableauformeln des initialen Tableaus könnte beispielsweise die Formeln $\mathsf{T}{:}(\neg p(a) \lor \neg p(b))$ und $\mathsf{T}{:}(\neg q(c) \lor \neg q(d))$ enthalten).

Wir nehmen weiter an, daß die Methode zur Konstruktion eines Grund-Tableaubeweises aus dem Beweis von Satz 4.3.5 schon auf alle Erweiterungsschritte angewendet worden ist, die notwendig sind, um die Teiltableaus T^{uv}_p und T^{uv}_q zu konstruieren. Nun betrachten wir den Erweiterungsschritt, in dem die universelle Variablen enthaltende Konklusion $\{\{\mathsf{T}{:}p(\boldsymbol{x})\}, \{\mathsf{T}{:}p(\boldsymbol{y})\}\}$ aus der Prämisse $\{\mathsf{T}{:}(p(\boldsymbol{x}) \lor p(\boldsymbol{y}))\}$ abgeleitet wird. Sei B^{uv} der einzelne Ast des Tableaus vor der Erweiterungsregelanwendung, und seien B^{uv}_p und B^{uv}_q die beiden Äste des Tableaus danach. Die Konstruktion des Grund-Tableaubeweises ist schon soweit fortgeschritten, daß geschlossene Teiltableaus T^{gd}_p und T^{gd}_q vorliegen, die aus jedem beliebigen Grund-Tableauast erzeugt werden können, der bestimmte Instanzen der Formeln auf B^{uv}_p bzw. B^{uv}_q enthält. Wir nehmen an, daß $\Phi^{\mathrm{gd}}_p = \{\mathsf{T}{:}p(a), \mathsf{T}{:}p(b)\}$ und $\Phi^{\mathrm{gd}}_q = \{\mathsf{T}{:}q(c), \mathsf{T}{:}q(d)\}$ diese Menge von Instanzen seien.

Abbildung 4.4 zeigt das komplette geschlossene Grundtableau. Wie sich leicht überprüfen läßt, enthält jeder der neuen Teiläste entweder sowohl $\mathsf{T}{:}p(a)$ als auch $\mathsf{T}{:}p(b)$ oder sowohl $\mathsf{T}{:}q(c)$ als auch $\mathsf{T}{:}q(d)$.[2] $\square$

Für viele Logiken kann die Reduktion in der Größe kürzester Beweise, die sich mit der Verwendung universeller Variablen erzielen läßt, auch durch eine zusätzliche nicht-analytische Schnittregel erreicht werden. In obigem Beispiel könnte man die Schnittformel

$$G = (\forall x)(\forall y)(p(x) \lor q(y)) \to (\forall x)(p(x)) \lor (\forall y)(q(y))$$

verwenden. Dann könnte derjenige der entstehenden Äste, der $\mathsf{F}{:}G$ enthält, mit einem Teil-Tableaubeweis konstanter Größe abgeschlossen werden, während der $\mathsf{T}{:}G$

[2] Man beachte, daß im Grundkalkül wie auch im universelle Variablen verwendenden Kalkül ein Regelschema für γ-Formeln benutzt wird, das erlaubt, $\mathsf{T}{:}(p(t) \lor q(t'))$ in einem Schritt aus $\mathsf{T}{:}(\forall x)(\forall y)(p(x) \lor q(y))$ abzuleiten, ohne daß zuerst $\mathsf{T}{:}(\forall y)(p(t) \lor q(y))$ abgeleitet wird.

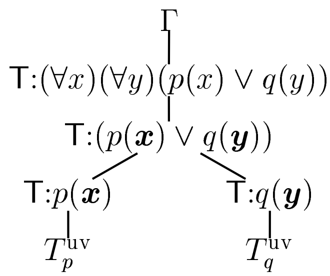


Abbildung 4.3: Ein Tableaubeweis mit universellen Variablen (Beispiel 4.3.7).

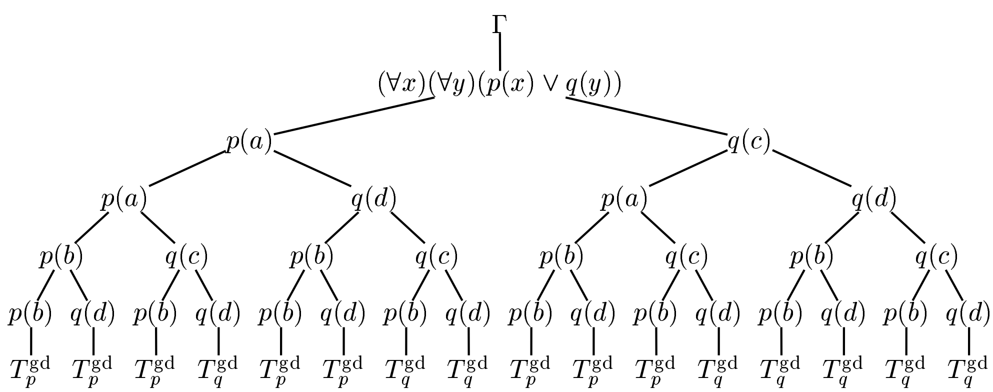


Abbildung 4.4: Ein Grund-Tableaubeweis, der aus dem Tableaubeweis mit universellen Variablen in Abbildung 4.3 konstruiert ist (siehe Beispiel 4.3.7); um die Lesbarkeit zu verbessern, sind die Markierungen und Vorzeichen nicht dargestellt (die in allen Fällen $*$ bzw. T sind).

enthaltende Ast mit einem Teil-Tableaubeweis von gleicher Größe wie der Tableaubeweis mit universellen Variablen geschlossen werden könnte. Also kann die Technik der universellen Variablen in gewisser Weise als eine eingeschränkte Anwendung der nicht-analytischen Schnittregel angesehen werden (eine unbeschränkte Verwendung der Schnittregel führt zu einer Suchraumexplosion).

In (Stenz, 1997) ist eine Transformation von Tableaubeweisen, die mit Hilfe eines universelle Variablen verwendenden Kalküls für PL1 konstruiert sind, in Grund-Tableaubeweise beschrieben. Diese Transformation kann jedoch nur angewendet werden, falls universelle Variablen in dem Ausgangsbeweis niemals in einer Konklusion mit mehr als einer Extension auftreten (also niemals in einer verzweigenden Konklusion). In diesem Fall kann ein Grund-Tableaubeweis konstruiert werden, dessen Größe polynomiell in der Größe des Beweises mit universellen Variablen ist.

4.3.3 Konstruktion eines universelle Variablen verwendenden Kalküls

In diesem Abschnitt diskutieren wir die Frage, wie eine universelle Variablen verwendende Version $\mathcal{C}^{\mathrm{uv}}$ eines Grund-Tableaukalküls $\mathcal{C}^{\mathrm{gd}}$ zu konstruieren ist.

Eine Möglichkeit ist, eine starre Variablen verwendende Version $\mathcal{C}^{\mathrm{rv}}$ von $\mathcal{C}^{\mathrm{gd}}$, die mit Hilfe der in Abschnitt 4.2 beschriebenen Technik des Hebens konstruiert worden ist, in einen Kalkül mit universellen Variablen zu verwandeln. Das kann wie folgt geschehen: Sei eine Prämisse Π_{uv} mit universellen Variablen gegeben. Dann werde jede universelle Variable in Π_{uv} durch eine starre Variable ersetzt, so daß eine nur starre Variablen enthaltende Prämisse entsteht. Dabei werden mehrerer Vorkommen der gleichen universellen Variablen $\boldsymbol{x}$ in einer Formel durch die gleiche starre Variable ersetzt. Vorkommen von $\boldsymbol{x}$ in anderen Formeln werden durch andere starre Variablen ersetzt; und verschiedene universelle Variablen werden immer durch verschiedene starre Variablen ersetzt. Dann besteht die Menge möglicher Konklusionen mit universellen Variablen für Π_{uv} aus allen C_{uv}, die wie folgt aus Konklusionen $\langle C_{\mathrm{rv}}, \tau\rangle$ mit starren Variablen von Π_{rv} konstruiert werden können:

1. Jede starre Variable, die in nur einer Extension von C_{rv} vorkommt, wird durch eine universelle Variable ersetzt.

2. Jede starre Variable, die in mehr als einer Extension von C_{rv} vorkommt, wird durch einen beliebigen Grundterm $t \in \mathit{TabTerm}$ ersetzt.

Alle Vorkommen einer starren Variablen werden durch die gleiche universelle Variable bzw. den gleichen Term ersetzt. Man beachte, daß die Substitution τ in der Konklusion mit starren Variablen keine Rolle in der Konstruktion der Konklusion mit universellen Variablen spielt. Der so entstehende Kalkül mit universellen Variablen ist per Konstruktion gutartig.

Beispiel 4.3.8 In Tabelle 4.4 sind Beispiele für die wichtigsten Phänomene aufgeführt, die auftreten können, wenn die oben beschriebene Methode zur Berechnung von Konklusionen mit universellen Variablen verwendet wird.

(a) Eine neue universelle Variable wird eingeführt.

(b) Eine Variable wird über zwei Extensionen verteilt; sie verliert ihre Universalität und muß durch einen Grundterm ersetzt werden.

(c) Eine Variable tritt in nur einer Extension auf und bleibt darum universell.

(d) Eine Kombination der Fälle (b) und (c); die Variable $\boldsymbol{x}$ muß durch einen Grundterm ersetzt werden, während die Variable $\boldsymbol{y}$ universell bleibt.

(e) In der Version mit starren Variablen muß eine Substitution auf das Tableau angewendet werden; in der Version mit universellen Variablen wird sie nicht angewendet, da die Variablen, die zu instantiieren wären, universell sind.

(f) In diesem Fall führt die Methode zur Konstruktion einer Konklusion mit universellen Variablen nicht zu einem optimalen Ergebnis. Jeder der beiden universellen Variablen wird über die beiden Extensionen verteilt, und sie werden darum durch Grundterme ersetzt. Jedoch beschreibt auch das alternative Schema

$$\frac{\mathsf{T}{:}p(\boldsymbol{x}) \leftrightarrow q(\boldsymbol{y})}{\begin{array}{c|c} \mathsf{T}{:}p(\boldsymbol{x}) & \mathsf{F}{:}p(\boldsymbol{x}) \\ \mathsf{T}{:}q(\boldsymbol{y}) & \mathsf{F}{:}q(\boldsymbol{y}) \end{array}}$$

für PL1-Formeln der Form $p(\boldsymbol{x}) \leftrightarrow q(\boldsymbol{y})$ eine korrekte Erweiterungsregel; die beiden universellen Variablen bleiben universell, obwohl sie über die Extensionen verteilt werden. Dies Schema ist korrekt, vorausgesetzt, daß die universellen Variablen $\boldsymbol{x}$ und $\boldsymbol{y}$ in jeweils nur einer Teilformel der Prämisse auftreten.[3] □

Wie Fälle (a) und (b) in obigem Beispiel zeigen, werden in Kalkülen mit universellen Variablen verschiedene Varianten (oder Instanzen) einer Konklusion durch das verzweigende Regelschemata erzeugt, und nicht wie in Kalkülen mit starren Variablen durch die Schemata, die neue Variablen einführen.

Für bestimmte Formelklassen kann die Verwendung eines Kalküls $\mathcal{C}^{\mathrm{uv}}$ mit universellen Variablen, der aus einem Kalkül $\mathcal{C}^{\mathrm{fv}}$ mit starren Variablen konstruiert worden ist, Tableaubeweise erlauben, die exponentiell kleiner sind als die kürzesten Beweise, die mit $\mathcal{C}^{\mathrm{fv}}$ oder seiner Grundversion $\mathcal{C}^{\mathrm{gd}}$ geführt werden können. Nichtsdestotrotz ist das Ergebnis dieser Konstruktion nicht immer ein optimaler Kalkül mit universellen Variablen (wie Fall (f) in Beispiel 4.3.8 zeigt). Tatsächlich ist die Frage, ob eine in der

[3] Intuitiv ist dieses liberalere Schema korrekt, weil, falls $p(t)$ und $q(t')$ für alle $t, t' \in \mathit{Term}$ äquivalent sind, die Formeln $p(t)$ und $q(t)$ entweder beide für alle $t \in \mathit{Term}$ wahr sind oder beide für alle $t \in \mathit{Term}$ falsch sind.

$$\begin{array}{rc}
\Pi_{\mathrm{uv}} = & \mathsf{T}{:}(\forall x)(p(x,\boldsymbol{y})) \\ \cline{2-2}
\Pi_{\mathrm{rv}} = & \mathsf{T}{:}(\forall x)(p(x,Y)) \\ \cline{2-2}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \mathsf{T}{:}p(X,Y)\ ,\ id\rangle \\ \cline{2-2}
C_{\mathrm{uv}} = & \mathsf{T}{:}p(\boldsymbol{x},\boldsymbol{y})
\end{array}$$

(a)

$$\begin{array}{rc|c}
\Pi_{\mathrm{uv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(\boldsymbol{x}) \vee q(\boldsymbol{x})} \\ \cline{2-3}
\Pi_{\mathrm{rv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(X) \vee q(X)} \\ \cline{2-3}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \mathsf{T}{:}p(X) & \mathsf{T}{:}q(X)\ ,\ id\rangle \\ \cline{2-3}
C_{\mathrm{uv}} = & \mathsf{T}{:}p(t) & \mathsf{T}{:}q(t)
\end{array}$$

für alle $t \in \mathit{Term}$

(b)

$$\begin{array}{rc}
\Pi_{\mathrm{uv}} = & \mathsf{T}{:}p(\boldsymbol{x}) \wedge q(\boldsymbol{x}) \\ \cline{2-2}
\Pi_{\mathrm{rv}} = & \mathsf{T}{:}p(X) \wedge q(X) \\ \cline{2-2}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \begin{array}{c}\mathsf{T}{:}p(X) \\ \mathsf{T}{:}q(X)\end{array}\ ,\ id\rangle \\ \cline{2-2}
C_{\mathrm{uv}} = & \begin{array}{c}\mathsf{T}{:}p(\boldsymbol{x}) \\ \mathsf{T}{:}q(\boldsymbol{x})\end{array}
\end{array}$$

(c)

$$\begin{array}{rc|c}
\Pi_{\mathrm{uv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(\boldsymbol{x}) \vee q(\boldsymbol{x},\boldsymbol{y})} \\ \cline{2-3}
\Pi_{\mathrm{rv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(X) \vee q(X,Y)} \\ \cline{2-3}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \mathsf{T}{:}p(X) & \mathsf{T}{:}q(X,Y)\ ,\ id\rangle \\ \cline{2-3}
C_{\mathrm{uv}} = & \mathsf{T}{:}p(t) & \mathsf{T}{:}q(t,\boldsymbol{y})
\end{array}$$

für alle $t \in \mathit{Term}$

(d)

$$\begin{array}{rc}
\Pi_{\mathrm{uv}} = & \begin{array}{c}\mathsf{T}{:}p(f(a)) \\ \mathsf{T}{:}f(\boldsymbol{x}) \approx \boldsymbol{x}\end{array} \\ \cline{2-2}
\Pi_{\mathrm{rv}} = & \begin{array}{c}\mathsf{T}{:}p(f(a)) \\ \mathsf{T}{:}f(X) \approx X\end{array} \\ \cline{2-2}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \mathsf{T}{:}p(a)\ ,\ \{X \mapsto a\}\rangle \\ \cline{2-2}
C_{\mathrm{uv}} = & \mathsf{T}{:}p(a)
\end{array}$$

(e)

$$\begin{array}{rc|c}
\Pi_{\mathrm{uv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(\boldsymbol{x}) \leftrightarrow q(\boldsymbol{y})} \\ \cline{2-3}
\Pi_{\mathrm{rv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(X) \leftrightarrow q(Y)} \\ \cline{2-3}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \begin{array}{c}\mathsf{T}{:}p(X) \\ \mathsf{T}{:}q(Y)\end{array} & \begin{array}{c}\mathsf{F}{:}p(X) \\ \mathsf{F}{:}q(Y)\end{array}\ ,\ id\rangle \\ \cline{2-3}
C_{\mathrm{uv}} = & \begin{array}{c}\mathsf{T}{:}p(t) \\ \mathsf{T}{:}q(t')\end{array} & \begin{array}{c}\mathsf{F}{:}p(t) \\ \mathsf{F}{:}q(t')\end{array}
\end{array}$$

für alle $t, t' \in \mathit{Term}$

(f)

Tabelle 4.4: Beispiele für die Konstruktion von Konklusionen mit universellen Variablen (siehe Beispiel 4.3.8).

Prämisse auftretenden universelle Variable in der Konklusion universell bleiben kann oder durch Grundterme ersetzt werden muß, im allgemeinen unentscheidbar (für den Fall der Prädikatenlogik erster Stufe wird dieses Problem in (Beckert & Hähnle, 1998) diskutiert).

4.3.4 Semantik von universelle Variablen enthaltenden Tableaus

Es ist im allgemeinen nicht möglich, eine Modellsemantik für Kalküle mit universellen Variablen zu definieren, wenn beide Vorzeichen F und T in Tableaus vorkommen können. Der Grund dafür ist der folgende: Definiert man (was naheliegend ist), daß $\phi(\boldsymbol{x})$ in $I(\sigma)$ wahr sei, falls $\phi(t)$ in $I(\sigma)$ für alle Terme t wahr ist, dann ist $\phi(\boldsymbol{x})$ falsch in $I(\sigma)$, falls es auch nur einen Term t gibt, so daß $\phi(t)$ in $I(\sigma)$ falsch ist. Das heißt, $\mathsf{F}{:}\sigma{:}\phi(\boldsymbol{x})$ ist erfüllt, wenn es einen Term t gibt, so daß $\mathsf{F}{:}\sigma{:}\phi(t)$ erfüllt ist. Das ist aber nicht die beabsichtigte Semantik der universellen Variablen $\boldsymbol{x}$.

Darum gründen wir die Semantik der Tableaus eines Kalküls $\mathcal{C}^{\mathrm{uv}}$ mit universellen Variablen auf die Semantik der Tableaus der Grundversion $\mathcal{C}^{\mathrm{gd}}$ von $\mathcal{C}^{\mathrm{uv}}$; wobei allerdings die Beziehung eine andere ist als zwischen der Semantik von Tableaus mit *starren* Variablen und Grundtableaus.

Definition 4.3.9 Sei $\mathcal{C}^{\mathrm{uv}}$ ein gutartiger Kalkül mit universellen Variablen für eine Logik $\mathbf{L}$; und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\mathrm{uv}}$ eine Version mit universellen Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.

Eine Tableauformel $\phi \in \mathit{TabForm}(\Sigma^*_{\mathrm{uv}})$ mit universellen Variablen wird genau dann von einer Tableauinterpretation $\langle \mathbf{m}, I\rangle \in \mathit{TabInterp}(\Sigma^*)$ von $\mathcal{C}^{\mathrm{gd}}$ erfüllt, wenn alle Grund-Tableauformeln in $\mathit{Inst}(\{\phi\})$ von $\langle \mathbf{m}, I\rangle$ erfüllt werden. □

Wie im Fall von Grundtableaus, wird ein Ast B_{uv} eines Tableaus mit universellen Variablen von einer Tableauinterpretation $\langle \mathbf{m}, I\rangle$ erfüllt, wenn sie alle Formeln auf B_{uv} erfüllt (oder, was äquivalent ist, wenn sie alle Formeln in $\mathit{Inst}(B_{\mathrm{uv}})$ erfüllt); und die Tableauinterpretation $\langle \mathbf{m}, I\rangle$ erfüllt ein Tableau T_{uv} mit universellen Variablen, wenn sie einen Ast von T_{uv} erfüllt.

Lemma 4.3.10 *Sei $\mathcal{C}^{\mathrm{uv}}$ ein gutartiger Kalkül mit universellen Variablen für eine Logik $\mathbf{L}$; und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\mathrm{uv}}$ eine Version mit universellen Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.*

Der Kalkül $\mathcal{C}^{\mathrm{uv}}$ hat genau dann die Eigenschaft der starken Korrektheit der Erweiterung (Eigenschaft 2 in Def. 3.5.8) bzgl. der Tableauinterpretationen von $\mathcal{C}^{\mathrm{gd}}$, wenn $\mathcal{C}^{\mathrm{gd}}$ diese Eigenschaft hat.

Beweis: Das Lemma folgt sofort aus den Definitionen der Korrektheitseigenschaft und der Beziehung zwischen einem Kalkül mit universellen Variablen und seiner Grundversion. □

Satz 4.3.11 *Sei $\mathcal{C}^{\mathrm{uv}}$ ein gutartiger Kalkül mit universellen Variablen für eine Logik* $\mathbf{L}$*; und sei $\mathcal{C}^{\mathrm{gd}}$ ein gutartiger Grundkalkül für* $\mathbf{L}$*, so daß $\mathcal{C}^{\mathrm{uv}}$ eine Version mit universellen Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.*

Wenn $\mathcal{C}^{\mathrm{uv}}$ die starken Korrektheitseigenschaften aus Definition 3.5.8 hat (Angemessenheit der Menge der Tableauinterpretationen und Korrektheit der Erweiterung), dann ist $\mathcal{C}^{\mathrm{uv}}$ korrekt.

Beweis: Da $\mathcal{C}^{\mathrm{uv}}$ die Eigenschaft 1 (starke Angemessenheit der Menge der Tableauinterpretationen) aus Definition 3.5.8 hat, hat auch $\mathcal{C}^{\mathrm{gd}}$ diese Eigenschaft, denn die initialen Tableaus für eine Menge $\mathfrak{F}$ von Formeln sind in beiden Kalkülen gleich.

Weil $\mathcal{C}^{\mathrm{uv}}$ die Eigenschaft 2 hat (starke Korrektheit der Erweiterung), hat auch $\mathcal{C}^{\mathrm{gd}}$ diese Eigenschaft (gemäß Lemma 4.3.10).

Also ist $\mathcal{C}^{\mathrm{gd}}$ korrekt (Satz 3.5.4), woraus folgt, daß auch $\mathcal{C}^{\mathrm{uv}}$ korrekt ist (Korollar 4.3.6). □

4.3.5 Mischen Starrer und Universeller Variablen

Ein Kalkül mit universellen Variablen kann syntaktisch als Grundkalkül aufgefaßt werden (da er keine starren Variablen enthält) und kann also gehoben und zusätzlich um starre Variablen angereichert werden.

Die Technik aus Abschnitt 4.3.3, mit der eine universelle Variablen verwendende Version eines Grundkalküls $\mathcal{C}^{\mathrm{gd}}$ aus einer starre Variablen verwendenden Version $\mathcal{C}^{\mathrm{rv}}$ von $\mathcal{C}^{\mathrm{gd}}$ konstruiert werden kann, kann so angepaßt werden, daß man einen Kalkül $\mathcal{C}^{\mathrm{mv}}$ mit gemischt starren und universellen Variablen konstruieren kann, der eine Version mit starren Variablen einer Version mit universellen Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.

Die Konstruktion beginnt genauso wie die eines ausschließlich universelle Variablen verwendenden Kalküls: Sei eine Prämisse Π_{mv} mit gemischten Variablen gegeben (d. h., eine Prämisse, die sowohl starre als auch universelle Variablen enthält). Zunächst wird eine ausschließlich starre Variablen enthaltende Prämisse Π_{rv} konstruiert, indem die universellen Variablen in Π_{mv} durch starre Variablen ersetzt werden, wobei – wie zuvor – Vorkommen einer universellen Variablen $\boldsymbol{x}$ in derselben Formel durch die gleiche starre Variable ersetzt werden, aber Vorkommen von $\boldsymbol{x}$ in verschiedenen Formeln durch verschiedene starre Variablen; und verschiedene universelle Variablen werden immer durch verschiedene starre Variablen ersetzt. Zusätzliche verlangen wir jetzt, daß die neu eingeführten starren Variablen von den starren Variablen verschieden sind, die schon in Π_{mv} vorkommen. Dann besteht die Menge der Konklusionen mit gemischten Variablen für die Prämisse Π_{mv} aus allen Konklusionen C_{mv}, die wie folgt aus einer Konklusion $\langle C_{\mathrm{rv}}, \tau\rangle$ mit starren Variablen von Π_{rv} konstruiert werden können:

1. Alle starren Variablen in C_{rv}, die als ein Ersatz für eine universelle Variable $\boldsymbol{x}$ eingeführt worden sind und in nur einer Extension von C_{rv} vorkommen, werden wieder durch die ursprüngliche Variable $\boldsymbol{x}$ ersetzt.

2. Alle starren Variablen in C_{rv}, die als ein Ersatz für eine universelle Variable $\boldsymbol{x}$ eingeführt worden sind und in mehr als einer Extension von C_{rv} vorkommen, werden durch eine beliebige starre Variable ersetzt.

3. Die Substitution τ wird auf die starren Variablen eingeschränkt, die in der ursprünglichen Prämisse Π_{mv} mit gemischten Variablen vorkommen.

Der zweite der obigen Schritte mag redundant erscheinen; er ist aber notwendig, weil beispielsweise $\{\{\mathsf{T}{:}p(X)\},\ \mathsf{T}{:}q(X)\}\}$ für *alle* starren Variablen X eine Konklusion von $\{\{\mathsf{T}{:}p(\boldsymbol{x}) \vee q(\boldsymbol{x})\}\}$ sein muß. Intuitiv müssen Erweiterungsregelanwendungen, die die Universalität einer Variablen $\boldsymbol{x}$ zerstören, die Ableitung einer beliebigen Zahl von Varianten der Konklusion gestatten, die verschiedene starre Variablen X anstelle der universellen Variablen $\boldsymbol{x}$ enthalten.

Beispiel 4.3.12 Tabellen 4.5 und 4.6 zeigen Beispiele für Prämissen mit gemischten Variablen und die Konklusionen mit gemischten Variablen, die aus diesen Konklusionen mit der oben beschriebenen Methode konstruiert werden (vgl. Tabelle 4.4 und Beispiel 4.3.8 wo ähnliche Beispiele benutzt werden, um die Konstruktion einer Konklusion mit ausschließlich universellen Variablen zu demonstrieren).

(a) Eine neue universelle Variable $\boldsymbol{x}$ wird eingeführt. Die starre Variable Z bleibt unverändert.

(b) Die universelle Variable $\boldsymbol{x}$ wird über zwei Extensionen verteilt; sie verliert ihre Universalität und wird durch eine beliebige starre Variable X ersetzt. Die starre Variable Y bleibt unverändert.

(c) Die universelle Variable $\boldsymbol{x}$ tritt in nur einer Extension auf; darum bleibt sie universell. Die starre Variable Y bleibt unverändert.

(d) Ein Kombination der Fälle (b) und (c); die universelle Variable $\boldsymbol{x}$ muß durch eine beliebige starre Variable X ersetzt werden, während die Variable $\boldsymbol{y}$ universell bleibt. Die starre Variable Z bleibt unverändert.

(e) In der Version mit ausschließlich starren Variablen muß eine Substitution auf das Tableau angewendet werden; eine der instantiierten Variablen (die starre Variable X) ist als Ersatz für die universelle Variable $\boldsymbol{x}$ eingeführt worden, die andere instantiierte Variable (die starre Variable Y) kommt schon in der Prämisse mit gemischten Variablen vor. In der Konklusion mit gemischten Variablen wird nur die Variable Y, die schon in der Prämisse vorkommt, instantiiert.

(f) Dieses Beispiel zeigt, daß eine universelle Variable ihre Universalität verliert und durch eine beliebige starre Variable ersetzt werden muß (bzw. ersetzt bleibt), wenn sie in einem Term im Wertebereich der Substitution auftritt, die Teil der Konklusion mit ausschließlich starre Variablen ist. □

Wie Fall (f) im obigen Beispiel zeigt, muß eine universelle Variable durch eine starre Variable ersetzt werden, wenn sie im Wertebereich einer Substitution auftritt, die auf das Tableau anzuwenden ist. Darum ist es besser, wenn man die Wahl hat, entweder $\boldsymbol{x}$ mit Y oder Y mit (einem starren Ersatz für) $\boldsymbol{x}$ zu instantiieren, die erstgenannte Möglichkeit zu wählen.

Beim Entwurf eines Kalküls mit gemischten Variablen spielt eine allgemeinerer Version der Unifikation eine wichtige Rolle, die wir *UV-Unifikation* nennen. Im Fall, daß ausschließlich starre Variablen auftreten, wie auch im Fall, daß ausschließlich universelle Variablen auftreten, reicht die Standard-Unifikation, wie sie in Abschnitt 2.2.3 definiert ist, aus (der Unterschied ist nur, daß im universellen Fall der Unifikator nicht auf das Tableau angewendet wird). Im Fall, daß starre und universelle Variablen gemischt auftreten, muß jedoch UV-Unifikation verwendet werden, die die unterschiedliche Natur der beiden Typen von Variablen berücksichtigt.

Definition 4.3.13 Seien ϕ und ψ Tableauformeln über einer Signatur Σ^*_{mv} mit gemischt starren und universellen Variablen; und seien

$$\pi = \{\boldsymbol{x}_1 \mapsto X_1, \ldots, \boldsymbol{x}_k \mapsto X_k\} \text{ und } \rho = \{\boldsymbol{y}_1 \mapsto Y_1, \ldots, \boldsymbol{y}_k \mapsto Y_l\}$$

Substitutionen, die alle universellen Variablen in ϕ bzw. ψ durch neue starre Variablen ersetzen, d. h.,

1. $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ sind alle universellen Variablen, die in ϕ vorkommen, und $\boldsymbol{y}_1, \ldots, \boldsymbol{y}_l$ sind alle universellen Variablen, die in ψ vorkommen,

2. $X_1, \ldots, X_k, Y_1, \ldots, Y_l$ sind paarweise verschiedene starre Variablen, die weder in ϕ noch in ψ vorkommen.

Eine Substitution $\mu \in \mathit{Subst}(\Sigma^*_{\mathrm{mv}})$ heiße ein *UV-Unifikator* von ϕ und ψ, wenn sie die Einschränkung eines Unifikators von $\phi\pi$ und $\psi\rho$ auf die in ϕ und/oder ψ auftretenden starren Variablen ist. □

Man beachte, daß Definitions- und Wertebereich eines UV-Unifikators nur starre und keine universellen Variablen enthalten.

Beispiel 4.3.14 Die Substitution $\{Y \mapsto b\}$ ist ein UV-Unifikator der Tableauformeln $\mathsf{T}{:}p(\boldsymbol{x}, Y)$ und $\mathsf{T}{:}p(a, b)$; die leere Substitution id ist ein UV-Unifikator von $\mathsf{T}{:}p(\boldsymbol{x})$ und $\mathsf{T}{:}p(f(\boldsymbol{x}))$; und $\{X \mapsto f(Y)\}$ ist ein UV-Unifikator von $\mathsf{T}{:}p(X)$ und $\mathsf{T}{:}p(f(\boldsymbol{y}))$. □

$$
\begin{array}{r c}
\Pi_{\mathrm{mv}} = & \mathsf{T}{:}(\forall x)(p(x,\boldsymbol{y},Z)) \\ \cline{2-2}
\Pi_{\mathrm{rv}} = & \mathsf{T}{:}(\forall x)(p(x,Y,Z)) \\ \cline{2-2}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \mathsf{T}{:}p(X,Y,Z)\ ,\ id\rangle \\ \cline{2-2}
\langle C_{\mathrm{mv}},\tau\rangle = & \langle \mathsf{T}{:}p(\boldsymbol{x},\boldsymbol{y},Z)\ ,\ id\rangle
\end{array}
$$

(a)

$$
\begin{array}{r c|c}
\Pi_{\mathrm{mv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(\boldsymbol{x},Y)\vee q(\boldsymbol{x},Y)} \\ \cline{2-3}
\Pi_{\mathrm{rv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(X,Y)\vee q(X,Y)} \\ \cline{2-3}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \mathsf{T}{:}p(X,Y) & \mathsf{T}{:}q(X,Y)\ ,\ id\rangle \\ \cline{2-3}
\langle C_{\mathrm{mv}},\tau\rangle = & \langle \mathsf{T}{:}p(X,Y) & \mathsf{T}{:}q(X,Y)\ ,\ id\rangle \\
& \multicolumn{2}{c}{\text{für alle } X \in \mathit{Var}}
\end{array}
$$

(b)

$$
\begin{array}{r c}
\Pi_{\mathrm{mv}} = & \mathsf{T}{:}p(\boldsymbol{x},Y)\wedge q(\boldsymbol{x},Y) \\ \cline{2-2}
\Pi_{\mathrm{rv}} = & \mathsf{T}{:}p(X,Y)\wedge q(X,Y) \\ \cline{2-2}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \begin{matrix}\mathsf{T}{:}p(X,Y)\\ \mathsf{T}{:}q(X,Y)\end{matrix}\ ,\ id\rangle \\ \cline{2-2}
\langle C_{\mathrm{mv}},\tau\rangle = & \langle \begin{matrix}\mathsf{T}{:}p(\boldsymbol{x},Y)\\ \mathsf{T}{:}q(\boldsymbol{x},Y)\end{matrix}\ ,\ id\rangle
\end{array}
$$

(c)

$$
\begin{array}{r c|c}
\Pi_{\mathrm{mv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(\boldsymbol{x},Z)\vee q(\boldsymbol{x},\boldsymbol{y},Z)} \\ \cline{2-3}
\Pi_{\mathrm{rv}} = & \multicolumn{2}{c}{\mathsf{T}{:}p(X,Z)\vee q(X,Y,Z)} \\ \cline{2-3}
\langle C_{\mathrm{rv}},\tau\rangle = & \langle \mathsf{T}{:}p(X,Z) & \mathsf{T}{:}q(X,Y,Z)\ ,\ id\rangle \\ \cline{2-3}
\langle C_{\mathrm{mv}},\tau\rangle = & \langle \mathsf{T}{:}p(X,Z) & \mathsf{T}{:}q(X,\boldsymbol{y},Z)\ ,\ id\rangle \\
& \multicolumn{2}{c}{\text{für alle } X \in \mathit{Var}}
\end{array}
$$

(d)

Tabelle 4.5: Beispiele für die Konstruktion von Konklusionen mit gemischten Variablen (erster Teil), siehe Beispiel 4.3.12.

$$\begin{array}{rc}
\Pi_{\mathrm{mv}} = & \begin{array}{c}\mathsf{T}{:}p(f(a,b)) \\ \mathsf{T}{:}f(\boldsymbol{x},Y) \approx \boldsymbol{x}\end{array} \\ \hline
\Pi_{\mathrm{rv}} = & \begin{array}{c}\mathsf{T}{:}p(f(a,b)) \\ \mathsf{T}{:}f(X,Y) \approx X\end{array} \\ \hline
\langle C_{\mathrm{rv}}, \tau\rangle = & \langle \mathsf{T}{:}p(a)\ ,\ \{X \mapsto a, Y \mapsto b\}\rangle \\ \hline
\langle C_{\mathrm{mv}}, \tau\rangle = & \langle \mathsf{T}{:}p(a)\ ,\ \{Y \mapsto b\}\rangle
\end{array}$$

(e)

$$\begin{array}{rc}
\Pi_{\mathrm{mv}} = & \begin{array}{c}\mathsf{T}{:}p(f(\boldsymbol{x})) \\ \mathsf{F}{:}p(Y)\end{array} \\ \hline
\Pi_{\mathrm{rv}} = & \begin{array}{c}\mathsf{T}{:}p(f(X)) \\ \mathsf{F}{:}p(Y)\end{array} \\ \hline
\langle C_{\mathrm{rv}}, \tau\rangle = & \langle \bot\ ,\ \{Y \mapsto f(X)\}\rangle \\ \hline
\langle C_{\mathrm{mv}}, \tau\rangle = & \langle \bot\ ,\ \{Y \mapsto f(X)\}\rangle \\
& \text{für alle } X \in \mathit{Var}
\end{array}$$

(f)

Tabelle 4.6: Beispiele für die Konstruktion von Konklusionen mit gemischten Variablen (zweiter Teil), siehe Beispiel 4.3.12.

Die Definitionen der Semantik ausschließlich starre bzw. ausschließlich universelle Variablen enthaltender Tableaus (Def. 4.2.23 und Def. 4.3.9) können leicht kombiniert werden, um eine Semantik für Tableaus mit gemischten Variablen zu definieren:

Definition 4.3.15 Sei $\mathcal{C}^{\mathrm{mv}}$ ein gutartiger Kalkül mit gemischten Variablen für eine Logik $\mathbf{L}$; und sei $\mathcal{C}^{\mathrm{gd}}$ ein Grundkalkül für $\mathbf{L}$, so daß $\mathcal{C}^{\mathrm{mv}}$ eine Version mit starren Variablen einer Version mit universellen Variablen von $\mathcal{C}^{\mathrm{gd}}$ ist.

Ein Tableau mit gemischten Variablen wird genau dann von einer Tableauinterpretation $\langle \mathbf{m}, I\rangle$ des Grundkalküls $\mathcal{C}^{\mathrm{gd}}$ erfüllt, wenn für alle Substitutionen $\tau \in \mathit{Subst}(\Sigma^*_{\mathrm{rv}})$, für die $T^{\mathrm{mv}}\tau$ keine starren Variablen enthält, das Tableau $T^{\mathrm{mv}}\tau$ von $\langle \mathbf{m}, I\rangle$ erfüllt wird (gemäß der Erfüllbarkeit eines Tableaus mit ausschließlich universellen Variablen, Def. 4.3.9), d. h., wenn es einen Ast B von $T^{\mathrm{mv}}\tau$ gibt, so daß $\langle \mathbf{m}, I\rangle$ alle (Grund-) Tableauformeln in $\mathit{Inst}(\mathit{Form}(B))$ erfüllt. □

4.3.6 Ein gutartiger Kalkül mit gemischten Variablen für PL1

In diesem Abschnitt definieren wir einen Kalkül $\mathcal{C}^{\mathrm{mv}}_{\mathrm{PL1}}$ mit gemischten Variablen für die Prädikatenlogik erster Stufe. Er ist mit Hilfe der in Abschnitt 4.2.10 beschriebenen Methode konstruiert, und zwar aus dem Kalkül $\mathcal{C}^{\mathrm{rv}}_{\mathrm{PL1}}$ mit ausschließlich starren Variablen, der in Abschnitt 4.3.5 definiert ist.

Für jede prädikatenlogische Signatur Σ (siehe Abschnitt 2.3) enthält die erweiterte Signatur Σ^*_{mv} nur die starren Variablen aus Var und die universellen Variablen aus $UVar$ als Konstanten (siehe Abschnitt 2.3); zusätzlich enthält sie die Menge $F^{sko}(\Sigma)$ von Skolem-Funktionssymbolen, in der es unendlich viele Symbole jeder Stelligkeit gibt. Alle freien (starren und universellen) Variablen sind von der gleichen Sorte.

Die Menge der Markierungen und die initiale Markierung von $\mathcal{C}^{\mathrm{mv}}$ sind die gleichen wie die von $\mathcal{C}^{\mathrm{gd}}$ und $\mathcal{C}^{\mathrm{rv}}$, d. h., die Markierung $*$ repräsentiert die einzelne Welt in PL1-Modellen.

Wie zuvor verwenden wir die vereinheitlichende Notation, um die Erweiterungsregel von $\mathcal{C}^{\mathrm{mv}}$ zu beschreiben, teilen also die Menge der Tableauformeln gemäß Tabelle 3.1 in α-, β-, γ- und δ-Formeln ein.

Wie schon zu Beginn dieses Abschnittes gesagt, sind die Erweiterungsregel-Schemata von $\mathcal{C}^{\mathrm{mv}}$ mit Hilfe der im letzten Abschnitt beschriebenen Methode aus den Schemata von $\mathcal{C}^{\mathrm{rv}}$ konstruiert. Die Schemata von $\mathcal{C}^{\mathrm{mv}}$ unterscheiden sich besonders in zwei Punkten von den entsprechenden Schemata des Kalküls $\mathcal{C}^{\mathrm{rv}}$, der ausschließlich starre Variablen verwendet: Erstens ist es nicht mehr das Schema für γ-Formeln, das neue Varianten einer Formel mit verschiedenen starren Variablen einführt, sondern das Schema für β-Formeln. Zweitens verlangt das Schema zum Abschluß von Ästen nur die Instantiierung von starren Variablen (und also nicht von universellen Variablen); das heißt, ein allgemeinster UV-Unifikator (Def. 4.3.13) der komplementären Atome wird angewendet.

Es werden die gleichen Skolem-Terme wie im Fall ausschließlich starrer Variablen verwendet (Def. 4.2.28); jedoch müssen *alle* freien Variablen, also starre und universelle, zu Argumenten des Skolem-Terms gemacht werden.

In Tabelle 4.7 ist die Erweiterungsregel schematisch für die verschiedenen Arten von Prämissen dargestellt, wobei nun starre und universelle Variablen gemischt auftreten können.

Das folgende ist eine formale Definition der Erweiterungsregel $\mathcal{E}^{\mathrm{mv}}_{\mathrm{PL1}}$ des Kalküls $\mathcal{C}^{\mathrm{mv}}_{\mathrm{PL1}}$ mit gemischt starren und universellen Variablen.

Definition 4.3.16 Für alle Signaturen $\Sigma \in Sig_{\mathrm{PL1}}$ und alle Prämissen

$$\Pi \subset TabForm_{\mathrm{PL1}}(\Sigma^*_{\mathrm{mv}})$$

bestehe $\mathcal{E}(\Sigma)^{\mathrm{mv}}_{\mathrm{PL1}}(\Pi)$ aus den folgenden Konklusionen mit gemischten Variablen:

$$\frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}}$$

$$\frac{\beta}{\beta_1\rho \mid \beta_2\rho}$$

für alle Substitutionen $\rho = \{\boldsymbol{x}_1 \mapsto X_1, \ldots, \boldsymbol{x}_k \mapsto X_k\}$,
so daß $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ diejenigen universellen Variablen sind,
die sowohl in β_1 als auch in β_2 auftreten
und $\{X_1, \ldots, X_k\}$ beliebige starre Variablen sind

$$\frac{\gamma(x)}{\gamma_1(\boldsymbol{x})}$$

wobei $\boldsymbol{x}$ eine
universelle Variable ist,
die nicht in γ vorkommt

$$\frac{\delta(x)}{\delta_1(t)}$$

wobei $t = sko_{\mathrm{fv}}(\delta)$
(siehe Def. 4.2.28)

$$\frac{\dfrac{\phi}{\psi}}{\bot}$$

falls ϕ und ψ unifizierbare atomare Formeln sind;
ein allgemeinster UV-Unifikator von ϕ, ψ ist auf das Tableau anzuwenden

Tabelle 4.7: Erweiterungsregel-Schemata für PL1 mit gemischt starren und universellen Variablen.

– $\{\langle\{\alpha_1, \alpha_2\}, id\rangle\}$ für alle $\alpha \in \Pi$,
– $\{\langle\{\beta_1\rho\}, \{\beta_2\rho\}, id\rangle\}$ für alle $\beta \in \Pi$ und alle $\rho = \{\boldsymbol{x}_1 \mapsto X_1, \ldots, \boldsymbol{x}_k \mapsto X_k\}$, so daß $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ die universellen Variablen sind, die sowohl in β_1 als auch in β_2 vorkommen, und $\{X_1, \ldots, X_k\}$ beliebige starre Variablen sind,
– $\{\langle\{\gamma_1(\boldsymbol{x})\}, id\rangle\}$ für alle $\gamma \in \Pi$, wobei $\boldsymbol{x}$ eine universelle Variable ist, die nicht in γ vorkommt,
– $\{\langle\{\delta(t)\}, id\rangle\}$ für alle $\delta \in \Pi$, wobei $t = sko_{\mathrm{fv}}(\delta)$ (Def. 4.2.28),
– $\{\langle\{\bot\}, \mu\rangle\}$ falls es Tableauformeln $\mathsf{T}{:}\sigma{:}G$ und $\mathsf{F}{:}\sigma{:}G'$ in Π gibt, so daß G und G' unifizierbare Atome sind; μ ist ein UV-Unifikator von G und G'.

□

Satz 4.3.17 *Der Kalkül $\mathcal{C}^{\mathrm{mv}}_{\mathrm{PL1}}$ ist eine Version mit starren Variablen einer Version mit universellen Variablen des in Abschnitt 3.6 definierten Kalküls $\mathcal{C}_{\mathrm{PL1}}$.*

Korollar 4.3.18 *Der Kalkül $\mathcal{C}^{\mathrm{mv}}_{\mathrm{PL1}}$ ist korrekt und vollständig.*

4.3.7 Ein gutartiger Kalkül mit gemischten Variablen für die Modallogik K

In diesem Abschnitt definieren wir eine Version $\mathcal{C}^{\mathrm{mv}}_{\mathrm{K}}$ mit gemischten Variablen des Kalküls $\mathcal{C}^{\mathrm{con}}_{\mathrm{K}}$ für die Modallogik K aus Abschnitt 3.7.4, d. h., einen Kalkül mit stetigem Erweiterungsregel-Schema für ν-Formeln. Variablen werden wieder in Markierungen verwendet (und nicht im Formelteil einer Tableauformel). Die Menge der Markierungen des Kalküls mit gemischten Variablen ist

$$Lab_{\mathrm{mv}} = CondLab(\mathbb{N} \cup Var \cup UVar)$$

mit der initialen Markierung 1.

Die Beziehung zwischen dem Kalkül $\mathcal{C}^{\mathrm{mv}}_{\mathrm{K}}$ mit gemischten Variablen zu dem Kalkül $\mathcal{C}^{\mathrm{rv}}_{\mathrm{K}}$ mit ausschließlich starren Variablen ist ähnlich wie die zwischen den Kalkülen $\mathcal{C}^{\mathrm{mv}}_{\mathrm{PL1}}$ und $\mathcal{C}^{\mathrm{rv}}_{\mathrm{PL1}}$ für PL1: In $\mathcal{C}^{\mathrm{rv}}_{\mathrm{PL1}}$ werden Varianten einer Formel durch Anwendung der Erweiterungsregel auf γ-Formeln generiert – während solche Regelanwendungen in $\mathcal{C}^{\mathrm{mv}}_{\mathrm{PL1}}$ universelle Variablen einführen und Varianten durch Erweiterungsregelanwendungen auf β-Formeln erzeugt werden. In analoger Weise werden in $\mathcal{C}^{\mathrm{rv}}_{\mathrm{K}}$ Varianten von Tableauformeln durch Erweiterungsregelanwendungen auf ν-Formeln erzeugt – während in $\mathcal{C}^{\mathrm{mv}}_{\mathrm{K}}$ durch solche Regelanwendungen universelle Variablen generiert werden und Variante durch Erweiterungsregelanwendungen auf β-Formeln erzeugt werden, wenn universelle Variablen über verschiedene Äste verteilt werden und dabei ihre Universalität verlieren.

In dem Kalkül $\mathcal{C}_{\mathrm{K}}^{\mathrm{rv}}$ mit ausschließlich starren Variablen werden wegen Tests auf Rechtfertigung alle Variablen, die in einem Paar komplementärer Atome vorkommen, instantiiert, wenn das Paar zum Abschluß eines Astes verwendet wird. Beispielsweise kann ein Paar $\mathsf{T}{:}1.(X){:}p$, $\mathsf{F}{:}1.(Y){:}p$ nur verwendet werden, um einen Ast abzuschließen, wenn Formeln wie $\mathsf{T}{:}1.1{:}q$ zur Verfügung stehen, die die Markierungen des komplementären Paars rechtfertigen. Also muß in diesem Fall die Substitution $\{X \mapsto 1, Y \mapsto 1\}$ angewendet werden. Wenn jedoch komplementäre Atome benutzt werden, die universelle Variablen enthalten, wie z. B. $\mathsf{T}{:}1.(\boldsymbol{x}){:}p$ und $\mathsf{F}{:}1.(\boldsymbol{y}){:}p$, dann muß man zwar noch immer testen, ob es Formeln auf dem Ast gibt, die eine Instanz $1.(n)$ von $1.(\boldsymbol{x})$ und $1.(\boldsymbol{y})$ rechtfertigen, aber es ist nicht mehr notwendig, die universellen Variablen $\boldsymbol{x}$ und $\boldsymbol{y}$ zu instantiieren, wenn der Ast geschlossen wird.

Anders als die Erweiterungsregel des gemischte Variablen verwendenden Kalküls $\mathcal{C}_{\mathrm{PL1}}^{\mathrm{mv}}$ für PL1 macht die Erweiterungsregel des Kalkül $\mathcal{C}_{\mathrm{K}}^{\mathrm{mv}}$ für K keinen Gebrauch von UV-Unifikation. Der Grund ist, daß freie Variablen in Markierungen niemals mit komplexen Termen instantiiert werden, sondern nur mit einer anderen freien Variablen oder einer natürlichen Zahl. Markierungen mit gemischten Variablen sind unifizierbar, wenn es Substitution gibt, die ihre jeweiligen universellen Variablen instantiiert, und eine weitere gemeinsame Substitution, die ihre starren Variablen instantiiert, so daß die Kombination dieser Substitutionen ein Unifikator der Markierungen ist. Die Substitutionen universeller und die Substitution starrer Variablen können sich nicht gegenseitig beeinflussen, und nur die starre Variablen instantiierende Substitution wird auf das Tableau angewendet.

Die Erweiterungsregel $\mathcal{E}_{\mathrm{K}}^{\mathrm{mv}}$ von $\mathcal{C}_{\mathrm{K}}^{\mathrm{mv}}$ ist in Tabelle 4.8 schematisch dargestellt; und sie ist im folgenden formal definiert.

Definition 4.3.19 Für alle Signaturen $\Sigma \in \mathit{Sig}_{\mathrm{mod}}$ und alle Prämissen

$$\Pi \subset \mathit{TabForm}_{\mathrm{mod}}(\Sigma)$$

bestehe $\mathcal{E}_{\mathrm{K}}^{\mathrm{rv}}(\Sigma)(\Pi)$ aus den folgenden Konklusionen mit gemischten Variablen (wobei $\lceil\cdot\rceil$ eine beliebige aber feste Bijektion von $\mathit{Form}_{\mathrm{mod}}(\Sigma)$ in die Menge der natürlichen Zahlen ist):

– $\langle\{\{\alpha_1, \alpha_2\}\}$ für alle $\alpha \in \Pi$,

– $\langle\{\{\beta_1\rho\}, \{\beta_2\rho\}\}, id\rangle$ für alle $\beta \in \Pi$ und alle $\rho = \{\boldsymbol{x}_1 \mapsto X_1, \ldots, \boldsymbol{x}_k \mapsto X_k\}$, so daß $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ diejenigen universellen Variablen sind, die sowohl in β_1 als auch in β_2 vorkommen, und $\{X_1, \ldots, X_k\}$ beliebige starre Variablen sind,

– $\langle\{\{\mathsf{T}{:}\sigma.(\boldsymbol{x}){:}G\}\}, id\rangle$ für alle $\mathsf{T}{:}\sigma{:}\Box G \in \Pi$, wobei $\boldsymbol{x} \in \mathit{UVar}$ eine universelle Variable ist, die nicht σ in vorkommt,

– $\langle\{\{\mathsf{F}{:}\sigma.(\boldsymbol{x}){:}G\}\}, id\rangle$ für alle $\mathsf{F}{:}\sigma{:}\Diamond G \in \Pi$, wobei $\boldsymbol{x} \in \mathit{UVar}$ eine universelle Variable ist, die nicht σ in vorkommt,

$$\frac{\alpha}{\begin{array}{c}\alpha_1\\ \alpha_2\end{array}}$$

$$\frac{\beta}{\beta_1\rho \mid \beta_2\rho}$$

für alle Substitutionen $\rho = \{\boldsymbol{x}_1 \mapsto X_1, \ldots, \boldsymbol{x}_k \mapsto X_k\}$,
so daß $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_k$ diejenigen universellen Variablen sind,
die sowohl in β_1 als auch in β_2 auftreten
und $\{X_1, \ldots, X_k\}$ beliebige starre Variablen sind

$$\frac{\mathsf{T}{:}\sigma{:}\Box G}{\mathsf{T}{:}\sigma.(\boldsymbol{x}){:}G} \qquad \frac{\mathsf{F}{:}\sigma{:}\Diamond G}{\mathsf{F}{:}\sigma.(\boldsymbol{x}){:}G}$$

wobei $\boldsymbol{x}$ eine universelle Variable ist, die nicht in σ vorkommt

$$\frac{\mathsf{T}{:}\sigma{:}\Diamond G}{\mathsf{T}{:}\sigma.n{:}G}$$

wobei $n = \lceil G \rceil$

$$\frac{\mathsf{F}{:}\sigma{:}\Box G}{\mathsf{F}{:}\sigma.n{:}G}$$

wobei $n = \lceil \neg G \rceil$

$$\frac{\mathsf{T}{:}\sigma{:}\neg G}{\mathsf{F}{:}\sigma{:}G} \qquad \frac{\mathsf{F}{:}\sigma{:}\neg G}{\mathsf{T}{:}\sigma{:}G}$$

$$\frac{\begin{array}{c}\mathsf{T}{:}\sigma{:}G\\ \mathsf{F}{:}\sigma'{:}G\end{array}}{\bot}$$

falls es eine Substitution $\mu \in \mathit{Subst}_{\mathrm{rv}}$ und Substitutionen $\rho, \rho' \in \mathit{Subst}_{\mathrm{uv}}$ gibt,
so daß $[\sigma\rho\mu] = [\sigma'\rho'\mu]$ und
die Markierungen $\sigma\rho\mu$ und $\sigma'\rho'\mu$ von Formeln in $\mathit{Inst}(B\mu)$ gerechtfertigt sind,
wobei B der erweiterte Ast ist;
eine allgemeinste solche Substitution μ ist auf das Tableau anzuwenden

Tabelle 4.8: Erweiterungsregel-Schemata des Kalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{mv}}$ mit gemischten Variablen für die Modallogik K.

– $\langle\{\{\mathsf{F}{:}\sigma.n{:}G\}\}, id\rangle$ für alle $\mathsf{F}{:}\sigma{:}\Box G \in \Pi$, wobei $n = \lceil\neg G\rceil$,
– $\langle\{\{\mathsf{T}{:}\sigma.n{:}G\}\}, id\rangle$ für alle $\mathsf{T}{:}\sigma{:}\Diamond G \in \Pi$, wobei $n = \lceil G\rceil$,
– $\langle\{\{\bot\}\}, \mu\rangle$ falls $\mathsf{T}{:}\sigma{:}G, \mathsf{F}{:}\sigma'{:}G \in \Pi$ und μ eine allgemeinste Substitution in $Subst_{\mathrm{rv}}$ ist, für die es Substitutionen $\rho, \rho' \in Subst_{\mathrm{uv}}$ gibt, so daß $[\sigma\rho\mu] = [\sigma'\rho'\mu]$ und die Markierungen $\sigma\rho\mu$ und $\sigma'\rho'\mu$ durch Formeln in $Inst(\Pi\mu)$ gerechtfertigt sind.

□

Satz 4.3.20 *Der Kalkül $\mathcal{C}_{\mathrm{K}}^{\mathrm{mv}}$ ist eine Version mit starren Variablen einer Version mit universellen Variablen des Kalküls $\mathcal{C}_{\mathrm{K}}$ aus Abschnitt 3.7.4.*

Korollar 4.3.21 *Der Kalkül $\mathcal{C}_{\mathrm{K}}^{\mathrm{mv}}$ ist korrekt und vollständig.*

Beispiel 4.3.22 Wir setzen Beispiele 3.7.22 und 4.2.34 fort und beweisen wieder, daß die Formel

$$G = \Box(\neg p \vee q) \wedge \Box p \wedge (\Diamond\neg q \vee \Diamond\neg p)$$

K-unerfüllbar ist, nun aber mit Hilfe des oben definierten Kalküls $\mathcal{C}_{\mathrm{K}}^{\mathrm{mv}}$ mit gemischten Variablen. Abbildung 4.5 zeigt ein geschlossenes Tableau $T_{\mathrm{mv}}^{\mathrm{con}}$ mit gemischten Variablen für G.[4]

Bei der Anwendung der Erweiterungsregel auf die ν-Formeln 2 und 4, durch die die Formeln 6 bzw. 7 zum Tableau hinzugefügt werden, werden die universellen Variablen $\boldsymbol{x}_1$ bzw. $\boldsymbol{x}_2$ eingeführt (statt starrer Variablen).

Wenn die Erweiterungsregel auf die Formel 6 angewendet wird, und die Formeln 12 und 13 zum Tableau hinzugefügt werden, verliert die in der Prämisse auftretende universelle Variable $\boldsymbol{x}$ ihre Universalität, da sie über beide Extensionen verteilt wird; in 12 und 13 ist sie durch die starre Variable X_1 ersetzt.

In der Version des Beweises, in der ausschließlich starre Variablen verwendet werden, muß die Substitution $\{X_1 \mapsto 1, X_2 \mapsto 1\}$ beim Abschluß des linken Astes auf das Tableau angewendet werden. Nun enthält das Tableau jedoch die universelle Variable $\boldsymbol{x}_2$ statt der starren Variablen X_2. Darum reicht es aus, zu überprüfen, daß eine Instantiierung von $\boldsymbol{x}_2$ *existiert*, die erlaubt, den Ast abzuschließen. Es genügt, die Substitution $\{X_1 \mapsto 1\}$, die die starre Variable X_1 instantiiert, auf das Tableau anzuwenden.

Wie im Beweis mit ausschließlich starren Variablen erfordert der Abschluß des mittleren Astes keine weiteren Instantiierungen, nachdem $\{X_1 \mapsto 1\}$ angewendet worden ist. Nun ist allerdings die Variable $\boldsymbol{x}_2$ universell und ist daher beim Abschluß des linken Astes *nicht* mit 1 instantiiert worden. Daher ist es nicht notwendig, eine zweite Variante der Formel 7 zu erzeugen; der rechte Ast kann mit den komplementären Atomen 7 und 18 abgeschlossen werden, weil $\boldsymbol{x}_2$ mit 2 instantiiert werden *könnte*.

[4] Die Abbildung zeigt die Tableauformeln mit uninstantiierten starren Variablen; die während der Konstruktion des Tableaus anzuwendende Substitution $\{X_1 \mapsto 1\}$ ist separat aufgeführt.

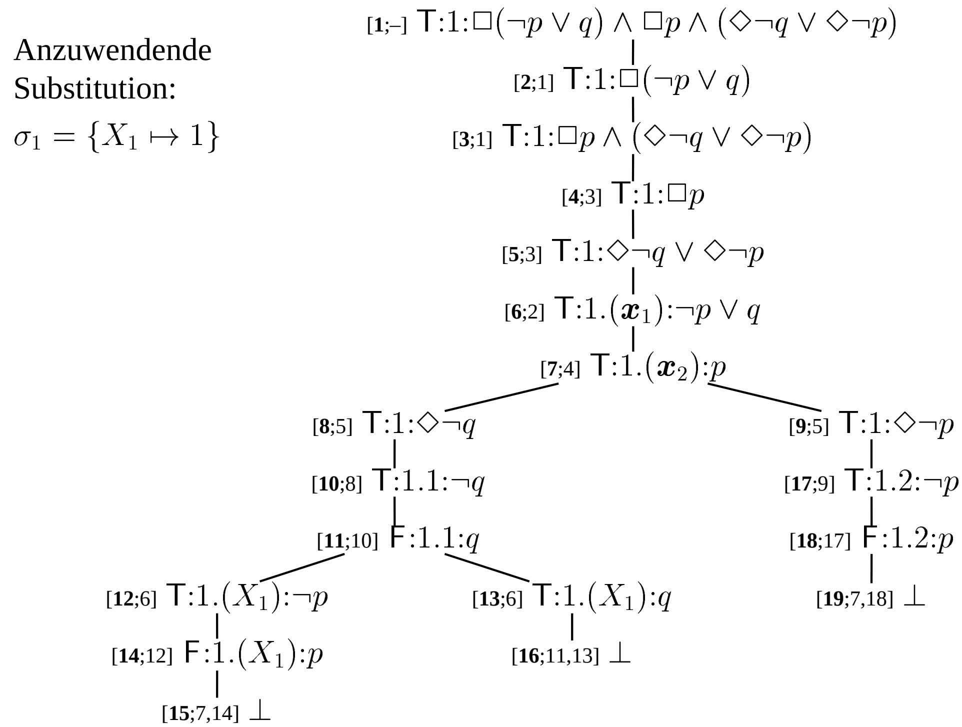


Abbildung 4.5: Das Tableau $T_{\mathrm{rv}}^{\mathrm{con}}$ aus Beispiel 4.2.34 entsteht durch Anwendung der Substitution σ_1 auf den obigen Baum.

Dann wäre die Markierung $1.(\boldsymbol{x}_2)$ der Formel 7 durch die Formeln des rechten Astes gerechtfertigt (nämlich durch Formel 18); es ist wiederum nicht notwendig, die Substitution $\{\boldsymbol{x}_2 \mapsto 2\}$, die eine universelle Variable instantiiert, tatsächlich auf das Tableau anzuwenden – ihre Existenz genügt. $\Box$

4.4 Verbesserte Skolemisierung

Skolemisierung ist eine erfüllbarkeitserhaltende Deduktion der folgenden allgemeinen Form: Wenn eine Formel bzw. eine Prämisse Π gegeben ist, die die Existenz eines Objekts mit bestimmten Eigenschaften in allen Modellen von Π impliziert, dann wird ein Skolem-Symbol, ein Skolem-Term oder ein anderes syntaktisches Konstrukt eingeführt, daß ein beliebiges dieser Objekte repräsentieren soll, deren Existenz bekannt ist; und eine Formel wird abgeleitet, die die Tatsache ausdrückt, daß das von dem Skolem-Symbol repräsentierte Objekt die bekannten Eigenschaften hat.

Beispiel 4.4.1 Eine Formel $\delta = (\exists x)(\phi(x))$ der Prädikatenlogik erster Stufe impliziert die Existenz eines Elementes d im Universum jeder δ erfüllenden prädikatenlogischen Struktur $\langle D, \mathcal{I}\rangle$ mit der Eigenschaft, daß $val_{\mathcal{I},[x\mapsto d]}(\phi(x)) = true$. Eine

Skolem-Konstante c (oder ein Skolem-Term t) wird eingeführt, die d repräsentiert; und die Formel $\phi(c)$ (bzw. $\phi(t)$) wird abgeleitet. □

Beispiel 4.4.2 Eine Ungleichung $\mathsf{F}{:}(s \approx t)$ impliziert, wenn s und t als Mengen interpretiert werden, die Existenz eines Elementes d, das in nur einer der beiden Mengen vorkommt, nicht aber in der anderen.

Darum führt die Erweiterungsregel des Kalküls $\mathcal{C}_{\mathrm{MLSS}}$ aus Abschnitt 3.8 für das Fragment MLSS der Mengentheorie, wenn sie auf eine Ungleichung angewendet wird, eine Konstante c ein, die das existierende Element d darstellt. □

Die Objekte, deren Existenz bekannt ist, müssen keine Elemente eines Universums sein. Sie können auch Funktionen, Relationen oder mögliche Welten sein (wie es bei Kalkülen für Modallogiken der Fall ist).

Tableaukalküle für viele Logiken verwenden die eine oder andere Art von Skolemisierung. Oft jedoch liefert die „Standard-Skolemisierung“ keine optimalen Ergebnisse. Diese Kalküle können wie folgt verbessert werden: Statt ein beliebiges neues Skolem-Symbol einzuführen, wird jeder Prämisse, aus der die Existenz eines Objektes mit bestimmte Eigenschaften abgeleitet werden kann, ihr eigenes Symbol zugewiesen (bzw. eine Konstante oder ein Term). Wenn die gleiche Prämisse noch einmal für eine Erweiterung verwendet wird, dann wird auch wieder das gleiche Skolem-Symbol benutzt. Da die Menge aller Prämissen aufzählbar ist, können sie nur endlich viele verschiedene Eigenschaften ausdrücken, so daß eine aufzählbare Menge von Skolem-Symbolen ausreicht.

Diese verbesserte Art der Skolemisierung wird von den gutartigen Kalkülen für Pl1 und für die Modallogik K verwendet, die in Kapitel 3 und den Abschnitten 4.2 und 4.3 beschrieben sind. Sie wurde zuerst für den Fall der Prädikatenlogik erster Stufe in (Beckert *et al.*, 1993) eingeführt, nämlich in Form eines liberalisierten Erweiterungsregel-Schemas für δ-Formeln. Eine verbesserte Version des Erweiterungsregel-Schemas für π-Formeln in Modallogiken wurde zuerst in (Beckert & Goré, 1997) vorgestellt. Die verbesserte Skolemisierung in Prädikatenlogik erster Stufe ist keine neue Idee; die Verwendung eines Skolem-Terms, der der zu skolemisierenden Formel eindeutig zugeordnet ist, entspricht in gewisser Weise den ϵ-Termen, die zuerst in (Hilbert & Bernays, 1939) definiert wurden (eine gute Einführung in ϵ-Logik findet sich in (Meyer Viol, 1995)).

Diese verbesserte Skolemisierung in Tableaukalkülen für das Automatische Beweisen zu verwenden, hat mehrere wichtige Vorteile. Erstens bewahrt sie die Monotonie der Erweiterungsregeln, während eine Regel, die Symbole einführt, die *neu* sein müssen, nicht monoton ist (Kalküle, die solch eine Regel verwenden, sind nicht gutartig). Zweitens wird durch die verbesserte Skolemisierung der Suchraum eingeschränkt, weil die Zahl der verschiedenen Skolem-Symbole, die in einem Beweis verwendet werden, kleiner ist. Und drittens haben Kalküle, die die verbesserte Skolemisierung nicht verwenden, in der Regel nicht die Eigenschaft der *starken* Korrektheit der Erweiterung (Eigenschaft 2 in Def. 3.5.8), d. h., sie erhalten nicht notwendig die Erfüllbarkeit durch *dieselben* Tableauinterpretationen. Der Grund hierfür ist, daß eine mit Standard-Skolemisierung konstruierte Konklusion nur durch eine Tableauinterpretation erfüllt wird, in der das neue Skolem-Symbol in der richtigen Weise interpretiert wird, nämlich durch eines der Objekte, deren Existenz bekannt ist. Wird dagegen die verbesserte Skolemisierung verwendet, dann weiß man von vorneherein, was die Eigenschaften des Objektes sind, das durch ein bestimmtes Skolem-Symbol repräsentiert wird. Also ist es möglich, die Menge der Tableauinterpretationen so zu wählen, daß sie nur *kanonische* Interpretationen enthält, in denen die Skolem-Symbole durch Objekte mit den richtigen Eigenschaften interpretiert werden. Wenn solche kanonischen Interpretationen benutzt werden, dann erhält (die verbesserte) Skolemisierung nicht nur die Erfüllbarkeit, sondern Prämisse und Konklusion werden durch *dieselben* (kanonischen) Tableauinterpretationen erfüllt. Um die Korrektheit des Kalküls zu beweisen, muß man dann natürlich zusätzlich zeigen, daß jedes initiale Tableau für eine erfüllbare Formelmenge von einer *kanonischen* Tableauinterpretation erfüllt wird. Das kann in der Regel nur dann gelingen, wenn man sicher sein kann, daß die Skolem-Symbole nicht in initialen Tableaus vorkommen können – was der Hauptgrund dafür ist, daß wir die Verwendung einer erweiterten Signatur für die Konstruktion von Tableaubeweisen zulassen, denn die zusätzlichen Symbole der erweiterten Signatur können nicht in initialen Tableaus vorkommen.

Oft ist es auch möglich – in Abhängigkeit von der Ausdrucksstärke der jeweiligen Logik –, die Korrektheit einer Erweiterungsregel, die die verbesserte Skolemisierung benutzt, *syntaktisch* zu beweisen (basierend auf der Korrektheit der entsprechenden Erweiterungsregel mit Standard-Skolemisierung). Beispielsweise beweist Egly (1998), daß es in einem Tableaukalkül für PL1 mit Standard-Skolemisierung und nicht-analytischer Schnittregel – da die Schnittregel trivialerweise korrekt ist, beeinträchtigt ihre Verwendung das Argument nicht – möglich ist, für alle Formeln G (in mehreren Schritten) aus der leeren Prämisse die Tableauformel

$$\mathsf{T}{:}(\forall x)((\exists y)(G(x,y) \to G(x, f(x))))$$

abzuleiten, wobei $f = sko_{\mathrm{fv}}((\exists y)(G(X,y))$. Daraus folgt die Korrektheit einer Erweiterungsregel für PL1, die die verbesserten Skolemisierung verwendet; denn die obige Formel erlaubt es, an beliebiger Stelle im Tableaubeweis die Tableauformel $\mathsf{T}{:}G(X, f(X))$ aus der Tableauformel $\mathsf{T}{:}(\exists y)(G(X,y))$ abzuleiten.

Neben der Frage, welches Skolem-Symbol zu verwenden ist, besteht das Problem, die Korrektheit der Skolemisierung im Zusammenhang mit freien (sowohl starren als auch universellen) Variablen sicherzustellen. Man muß dafür sorgen, daß die Instantiierung von starren Variablen mit Skolem-Symbolen (oder mit Termen, die Skolem-Symbole enthalten) nicht zu unerwünschten Ergebnissen führt.

Beispiel 4.4.3 Für alle Instantiierungen $\{Y \mapsto t\}$ der starren Variablen Y impliziert die Formel $(\exists x)(p(x,Y))\{Y \mapsto t\}$ die Existenz eines Objektes d, für das

$$val_{\mathcal{I},[x \mapsto d]}(p(x,Y)\{Y \mapsto t\}) = true$$

gilt. Im allgemeinen gibt es jedoch kein einzelnes d, das diese Eigenschaft für alle t hat. □

Das im obigen Beispiel dargestellte Problem wird zumeist dadurch gelöst, daß man einen komplexen Skolem-Term statt einer Skolem-Konstanten verwendet, wobei die freien Variablen, von denen abhängt, welche Objekte d die gewünschten Eigenschaften haben, zu Argumenten des Skolem-Terms gemacht werden, d. h., im Beispiel wird die Formel $p(c(Y), Y)$ statt der Formel $p(c, Y)$ abgeleitet.

Die Schwierigkeit beim Entwurf eines korrekten Regelschemas mit Skolemisierung ist, herauszufinden, wodurch genau beeinflußt wird, welches die Objekte sind, deren Existenz bekannt ist. Alle freien Variablen, deren Instantiierung einen Einfluß haben kann, müssen als Argumente des Skolem-Terms verwendet werden. Auf der anderen Seite wird der Kalkül unnötigerweise abgeschwächt und der Suchraum vergrößert, wenn zu viele Variablen in den Skolem-Term eingehen.

Beispielsweise verwendeten frühere Versionen von Kalkülen mit starren Variablen für PL1 eine Skolemisierung, bei der alle starren Variablen, die auf dem erweiterten Ast vorkommen, zu Argumenten des Skolem-Terms werden. Später erkannte man, daß nur die Instantiierung der Variablen, die in einer δ-Formeln δ vorkommen (und nicht aller Variablen auf dem Ast) einen Einfluß darauf haben, welche Objekt im Universum einer δ erfüllenden Tableauinterpretation die von δ ausgedrückte Eigenschaft haben (Hähnle & Schmitt, 1994); es ist also ausreichend, die in δ vorkommenden Variablen zu Argumenten des Skolem-Terms zu machen, der bei einer Regelanwendung auf δ eingeführt wird.

Man kann natürlich alle Schwierigkeit, die durch die Verwendung freier Variablen im Zusammenhang mit Skolemisierung entstehen, dadurch beseitigen, daß man die Eingabe-Formeln in einem Vorverarbeitungsschritt skolemisiert, wie es in allen Kalkülen geschieht, die Klauselnormalform verwenden. Allerdings ist das für manche Logiken nicht möglich, so z. B. für intuitionistische Prädikatenlogik. Außerdem führt Skolemisierung zu einem Informationsverlust (und frühe Skolemisierung zu frühem Informationsverlust). Betrachten wir beispielsweise die Formel $\phi = \mathsf{T}{:}(\exists x)(p(x))$ und ihre skolemisierte Version $\phi' = \mathsf{T}{:}p(c)$. Wenn ein lokales Lemma $\overline{\phi}$ bzw. $\overline{\phi'}$ erzeugt wird (siehe Abschnitt 4.5), dann ist das Lemma $\overline{\phi}$, das zu $\mathsf{T}{:}(\forall x)(\neg p(x))$ äquivalent ist, sehr viel nützlicher als das Lemma $\overline{\phi'}$, das zu $\mathsf{T}{:}\neg p(c)$ äquivalent ist. Tatsächlich haben Tableaukalküle für PL1 mit Lemma-Generierung und *dynamischer* Skolemisierung nicht-elementar kürzere Beweise für bestimmte Formelklassen als Kalküle mit Lemma-Generierung und Skolemisierung als *Vorverarbeitungsschritt* (Egly, 1998).

4.5 Lokale Lemmata

Eine einfache und in vielen Fällen nützliche Methode zum Stärken von Kalkülen mit Erweiterungsregel ist, sicherzustellen, daß die Extensionen einer Konklusion sich nicht *semantisch überschneiden*:

Definition 4.5.1 Sei $\mathcal{C}$ ein (Grund-)Tableaukalkül für eine Logik $\mathbf{L}$; und sei $\Sigma \in \mathit{Sig}$ eine Signatur von $\mathbf{L}$.

Zwei Extensionen $E_1, E_2 \subset \mathit{TabForm}(\Sigma^*)$ *überschneiden sich semantisch*, wenn es eine Tableauinterpretation in $\mathit{TabInterp}(\Sigma^*)$ gibt, die sowohl E_1 als auch E_2 erfüllt. □

Beispiel 4.5.2 Betrachten wir den Tableaukalkül $\mathcal{C}_{\mathrm{PL1}}$ für PL1 aus Abschnitt 3.6. Die zwei Extensionen $\{\mathsf{T}{:}p\}$ und $\{\mathsf{T}{:}q\}$ in der Konklusion der Prämisse $\{\mathsf{T}{:}(p \vee q)\}$ überschneiden sich semantisch, da $\mathsf{T}{:}p$ und $\mathsf{T}{:}q$ beide von ein und derselben Tableauinterpretation von $\mathcal{C}_{\mathrm{PL1}}$ erfüllt werden können. □

Wenn eine Erweiterungsregelanwendung eine Konklusion verwendet, die sich semantisch überschneidende Extensionen enthält, dann wird in gewissem Sinne weniger Information zum Tableau hinzugefügt. Intuitiv muß man, um ein Tableau T abzuschließen, zeigen, daß keine Tableauinterpretation existiert, die irgendeinen Ast von T erfüllt; und wenn es Äste gibt, die (teilweise) von denselben Tableauinterpretation erfüllt werden, dann müssen diese Tableauinterpretationen, die mehr als einen Ast erfüllen, mehrfach behandelt werden.

Extensionen können durch zusätzliche Tableauformeln überschneidungsfrei gemacht werden; dabei können aber auch zusätzliche Extensionen erforderlich werden, und man muß vorsichtig vorgehen, damit die Korrektheit des Kalküls erhalten bleibt.

Satz 4.5.3 *Sei $\mathcal{C}$ ein gutartiger (Grund-)Tableaukalkül für eine Logik* $\mathbf{L}$*, der (1) die Korrektheitseigenschaften aus Definition 3.5.3 hat (und also korrekt ist) und (2) vollständig ist; und sei Σ eine Signatur von* $\mathbf{L}$*.*

Für jede Konklusion $C = \{E_1, \dots, E_k\}$ ($k \geq 1$) über Σ^*, wobei $E_i = \{\phi^i_1, \dots, \phi^i_{r_i}\}$ ($1 \leq i \leq k$), sei die Konklusion $C^!$ wie folgt definiert:

$$C^! = \{E_i \cup \bigcup_{j=1}^{i-1} \{\overline{\phi^j_{l_j}}, \phi^j_1, \dots, \phi^j_{l_j - 1}\} \mid 1 \leq i \leq k, \textit{ und } 1 \leq l_j \leq r_j \textit{ für } 1 \leq j \leq i\} \ .$$

Sei der Kalkül $\mathcal{C}^!$ wie folgt definiert: Die Erweiterungsregel $\mathcal{E}^!$ von $\mathcal{C}^!$ ist für alle Signaturen $\Sigma \in \mathit{Sig}$ und alle Prämissen $\Pi \in \mathit{TabForm}(\Sigma^*)$ durch

$$\mathcal{E}^!(\Sigma)(\Pi) = \{C^! \mid C \in \mathcal{E}(\Sigma)(\Pi)\}$$

gegeben, wobei $\mathcal{E}$ die Erweiterungsregel von $\mathcal{C}$ ist; und $\mathcal{C}^!$ stimmt mit $\mathcal{C}$ von der Erweiterungsregel abgesehen überein.

Dann gilt folgendes:

1. *Der Kalkül $\mathcal{C}^!$ hat die Korrektheitseigenschaften aus Definition 3.5.3 (und ist also korrekt);*

2. *der Kalkül $\mathcal{C}^!$ ist vollständig;*

3. *für alle Signaturen $\Sigma \in Sig$ und alle Prämissen $\Pi \in TabForm(\Sigma^*)$ sind die Extensionen in allen Konklusionen in $\mathcal{E}^!(\Sigma)(\Pi)$ semantisch überschneidungsfrei.*

Beweis: *Korrektheitseigenschaften:* Die Korrektheitseigenschaft 1 aus Definition 3.5.3 überträgt sich trivialerweise, weil sich $\mathcal{C}$ und $\mathcal{C}^!$ nur in ihrer Erweiterungsregel unterscheiden.

Um zu beweisen, daß die zweite Korrektheitseigenschaft (Korrektheit der Erweiterung) erhalten bleibt, benutzen wir Lemma 3.5.9. Danach genügt es zu zeigen, daß, wenn eine der Extension in einer Konklusion $C = \{E_1, \ldots, E_k\}$ von einer Tableauinterpretation $\langle \mathbf{m}, I \rangle$ erfüllt wird, auch eine der Extensionen in $C^!$ von $\langle \mathbf{m}, I \rangle$ erfüllt wird. Sei E_i die erste Extension in C, die von $\langle \mathbf{m}, I \rangle$ erfüllt wird, d. h., $\langle \mathbf{m}, I \rangle$ erfüllt keine der Extensionen $E_1, \ldots, E_{i-1}$; und sei jeweils $\phi^j_{l_j}$ die erste Tableauformel in E_j, die nicht erfüllt ist ($1 \leq j \leq i-1$), d. h., die Tableauformeln $\phi^j_1, \ldots, \phi^j_{l_j-1}$ werden von $\langle \mathbf{m}, I \rangle$ erfüllt. Es folgt, daß die Extension

$$E_i \cup \bigcup_{j=1}^{i-1} \{\overline{\phi^j_{l_j}}, \phi^j_1, \ldots, \phi^j_{l_j-1}\} \;\;,$$

die ein Element von $C^!$ ist, von $\langle \mathbf{m}, I \rangle$ erfüllt wird.

Vollständigkeit: Die Vollständigkeit von $\mathcal{C}^!$ folgt sofort aus der Vollständigkeit von $\mathcal{C}$, weil beide Kalküle gutartig sind und also die zusätzlichen Tableauformeln in den Extensionen von $\mathcal{C}^!$ die Konstruktion eines $\mathcal{C}^!$-Tableaubeweises nicht beeinträchtigen können, der die gleichen Regelanwendungen benutzt, die zur Konstruktion eines $\mathcal{C}$-Tableaubeweises verwendet werden – man beachte, daß es für jede Extension $E^!$ in einer Konklusion $C^!$ eine Extension $E \in C$ gibt, so daß $E \subset E^!$.

Keine semantische Überschneidung: Seien E_i und $E_{i'}$ zwei verschiedene Extensionen in $C^!$; dann gibt es per Konstruktion von $C^!$ eine Tableauformel ϕ mit $\phi \in E_i$ und $\overline{\phi} \in E_{i'}$; daraus folgt sofort, daß sich E_i und $E_{i'}$ nicht semantisch überschneiden können. $\square$

Beispiel 4.5.4 Wir nehmen an, p, q, r, s, t seien Tableauformeln und

$$\begin{array}{c|c|c}
\multicolumn{3}{c}{\Pi} \\
\hline
p & r & t \\
q & s &
\end{array}$$

sei ein Erweiterungsregel-Schema eines Tableaukalküls $\mathcal{C}$. Dann ist

$$\begin{array}{c|c|c|c|c|c|c}
\multicolumn{7}{c}{\Pi} \\
\hline
p & r & r & t & t & t & t \\
q & s & s & & & & \\
 & \overline{p} & \overline{q} & \overline{p} & \overline{p} & \overline{q} & \overline{q} \\
 & & p & & & p & p \\
 & & & \overline{r} & \overline{s} & \overline{r} & \overline{s} \\
 & & & & r & & r
\end{array}$$

das entsprechende Erweiterungsregel-Schema ohne semantische Überschneidungen des Kalküls $\mathcal{C}^!$ (Satz 4.5.3). $\square$

Die Formeln, die zu Extensionen hinzugefügt werden, um sie überschneidungsfrei zu machen, kann man als *lokale Lemmata* ansehen. Wenn im obigen Beispiel ein Ast mit $C^!$ erweitert worden ist, und der neue Teilast, der p und q enthält, abgeschlossen worden ist, kann man folgern, daß in allen Tableauinterpretationen, die den Ast B erfüllen, mindestens eines des Atome p und q falsch ist und – darüberhinaus – daß entweder (a) p nicht erfüllt ist oder (b) p erfüllt ist aber q nicht. Also können diese Formeln als „Lemmata" zu den weiteren Extensionen hinzugenommen werden. Die Lemmata sind „lokal“, weil sie nur in der lokalen Konklusion verwendet werden.

Wie Beispiel 4.5.4 zeigt, kann es zu einem drastischen Anwachsen der Zahl der Extensionen führen, wenn man sicherstellen will, daß die Extensionen aller Konklusionen überschneidungsfrei sind – was also nicht immer von Vorteil ist. Nichtsdestotrotz ist die Methode nützlich, denn es steht einem frei, entweder die ursprüngliche Konklusion C oder die überschneidungsfreie Konklusion $C^!$ (Satz 4.5.3) zu verwenden und die Entscheidung von der Zahl der Extensionen in C bzw. $C^!$ abhängig zu machen.

Zudem kann man das Verfahren auch auf Konklusionen in Regelschemata anwenden statt nur auf einzelne Instanzen von Schemata. In vielen Fällen ist es dann möglich, das Schema für die Konklusion $C^!$ zu vereinfachen, da einige ihrer Extensionen inkonsistent sind.

Beispiel 4.5.5 Wir betrachten das folgende Erweiterungsregel-Schema, das typischerweise von einem Kalkül für die dreiwertige Łukasiewicz-Logik $\mathcal{L}_3$ verwendet wird:

$$\begin{array}{c|c}
\multicolumn{2}{c}{\mathsf{T}{:}\{\frac{1}{2}\}{:}F \lor G} \\
\hline
\mathsf{T}{:}\{0, \frac{1}{2}\}{:}F & \mathsf{T}{:}\{\frac{1}{2}\}{:}F \\
\mathsf{T}{:}\{\frac{1}{2}\}{:}G & \mathsf{T}{:}\{0, \frac{1}{2}\}{:}G
\end{array}$$

Die Markierungen sind Teilmengen der Menge $\{0, \frac{1}{2}, 1\}$ der Wahrheitswerte. Passende Tableauinterpretationen für diesen Kalkül können folgendermaßen definiert werden: Für jedes Modell der Łukasiewicz-Logik gebe es eine Tableauinterpretation $\langle \mathbf{m}, I \rangle$ mit je einer Welt $I(\sigma)$ für jede mögliche Markierung; und es gelte $I(\sigma) \models F$ genau dann, wenn der Wahrheitswert von F in dem entsprechenden mehrwertigen Modell ein Element von σ ist.

Wendet man die Konstruktion aus Satz 4.5.3 an, so erhält man das neue überschneidungsfreie Schema

$$\begin{array}{c|c|c}
\multicolumn{3}{c}{\mathsf{T}{:}\{\frac{1}{2}\}{:}F \vee G} \\
\hline
\mathsf{T}{:}\{0, \frac{1}{2}\}{:}F & \mathsf{T}{:}\{\frac{1}{2}\}{:}F & \mathsf{T}{:}\{\frac{1}{2}\}{:}F \\
\mathsf{T}{:}\{\frac{1}{2}\}{:}G & \mathsf{T}{:}\{0, \frac{1}{2}\}{:}G & \mathsf{T}{:}\{0, \frac{1}{2}\}{:}G \\
 & \mathsf{F}{:}\{0, \frac{1}{2}\}{:}F & \mathsf{F}{:}\{\frac{1}{2}\}{:}G \\
 & & \mathsf{T}{:}\{0, \frac{1}{2}\}{:}F
\end{array}$$

Die zweite Extension dieses Schemas ist unerfüllbar, da der Wahrheitswerte von F nicht zugleich ein Element von $\{\frac{1}{2}\}$ und *kein* Element von $\{0, \frac{1}{2}\}$ sein kann. Die dritte Extension kann vereinfacht werden, da die erste ihrer Formeln die letzte subsumiert; und die zweite und dritte Formel implizieren zusammen, daß 0 der Wahrheitswert von G ist. Das Ergebnis ist das folgende Schema, das überschneidungsfrei ist und die gleiche Zahl von Extensionen hat, wie das ursprüngliche Schema:

$$\begin{array}{c|c}
\multicolumn{2}{c}{\mathsf{T}{:}\{\frac{1}{2}\}{:}F \vee G} \\
\hline
\mathsf{T}{:}\{0, \frac{1}{2}\}{:}F & \mathsf{T}{:}\{\frac{1}{2}\}{:}F \\
\mathsf{T}{:}\{\frac{1}{2}\}{:}G & \mathsf{T}{:}\{0\}{:}G
\end{array}$$

□

Das obige Beispiel stammt aus (Hähnle, 1993), wo ein Algorithmus zur Konstruktion überschneidungsfreier Erweiterungsregeln für beliebige mehrwertige Logiken beschrieben wird.

Extensionen überschneidungsfrei zu machen, ist auch für Kalküle mit starren und universellen Variablen nützlich. Da diese aber keine durch Tableauinterpretationen definierte Semantik haben, ist die Konstruktion aus Satz 4.5.3 nicht direkt anwendbar. Statt dessen muß man sicherstellen, daß die Extensionen in der Grundversion des Kalküls überschneidungsfrei sind, bevor der Kalkül gehoben wird, um eine Version mit freien Variablen zu konstruieren

Beispiel 4.5.6 Die Erweiterungsregel des Kalküls $\mathcal{C}_{\mathrm{PL1}}$ für PL1 erlaubt die Ableitung von Konklusionen aus β-Formeln, die nicht überschneidungsfrei sind. Eine alternative Version des Schemas

$$\begin{array}{c|c}
\multicolumn{2}{c}{\beta} \\
\hline
\beta_1 & \beta_2
\end{array}$$

für β-Formeln ist das Schema

$$\begin{array}{c|c} \multicolumn{2}{c}{\beta} \\ \hline \beta_1 & \dfrac{\beta_2}{\beta_1} \end{array}$$

das überschneidungsfreie Konklusionen erzeugt; es kann auch für Versionen von $\mathcal{C}_{\mathrm{PL1}}$ mit starren Variablen verwendet werden.

Die Verwendung dieses Schemas führt für bestimmte Klassen von PL1-Formeln zu einer nicht-elementaren Reduktion der Länge der kürzesten existierenden Tableaubeweise (Egly, 1998). □

Das folgende Beispiel zeigt, daß man Vorsicht walten lassen muß bei der Erzeugung lokaler Lemmata, die universelle Variablen enthalten, und daß das Ergebnis der uniformen Konstruktion aus Satz 4.5.3 nicht immer optimal ist.

Beispiel 4.5.7 Wir betrachten das Erweiterungsregel-Schema

$$\begin{array}{c|c} \multicolumn{2}{c}{\mathsf{T}{:}(F(t) \vee G)} \\ \hline \mathsf{T}{:}F(t) & \mathsf{T}{:}G \end{array}$$
für alle Terme t

Es kann gehoben werden, um das universelle Variablen verwendende Schema

$$\begin{array}{c|c} \multicolumn{2}{c}{\mathsf{T}{:}(F(\boldsymbol{x}) \vee G)} \\ \hline \mathsf{T}{:}F(\boldsymbol{x}) & \mathsf{T}{:}G \end{array}$$
falls $\boldsymbol{x}$ nicht in G vorkommt

zu konstruieren. Wenn das Grund-Schema überschneidungsfrei gemacht wird, indem das Komplement von $\mathsf{T}{:}F(t)$ als Lemma zur rechten Extension hinzugefügt wird, dann hat das resultierende Schema

$$\begin{array}{c|c} \multicolumn{2}{c}{\mathsf{T}{:}(F(t) \vee G)} \\ \hline \mathsf{T}{:}F(t) & \mathsf{T}{:}G \\ & \mathsf{F}{:}F(t) \end{array}$$
für alle Terme t

unglücklicherweise keine universelle Variablen verwendende Version, weil der Term t nun in beiden Extensionen vorkommt.

Es ist jedoch möglich, das Lemma $\mathsf{F}{:}(\forall x)(F(x))$ anstelle des Lemmas $\mathsf{F}{:}F(t)$ zu verwenden. Dann produziert das Regelschema nicht völlig überschneidungsfreie Konklusionen, aber es gibt weniger Tableauinterpretationen, die beide Extensionen erfüllen; und das Schema hat eine universelle Variablen verwendende Version:

$$\begin{array}{c|c} \multicolumn{2}{c}{\mathsf{T}{:}(F(\boldsymbol{x}) \vee G)} \\ \hline \mathsf{T}{:}F(\boldsymbol{x}) & \mathsf{T}{:}G \\ & \mathsf{F}{:}(\forall x)(F(x)) \end{array}$$
falls $\boldsymbol{x}$ nicht in G vorkommt

□

4.6 Die Pruning-Methode

Die Pruning-Methode[5], die mit der *Condensing* genannten Technik aus (Oppacher & Suen, 1988) eng verwandt ist, erlaubt, sowohl die Größe des Suchraumes als auch die Länge der generierten Tableaubeweise zu reduzieren.

Nehmen wir an, ein Ast B eines Tableaus werde mit einer aus Extensionen $E_1, \ldots, E_k$ bestehenden Konklusion C erweitert, und im weiteren Verlauf des Beweises werde ein geschlossenes Teiltableau unter einer der Extensionen E_i konstruiert, wobei *keine* der Tableauformeln in E_i bei der Konstruktion des Teiltableaus in einer Prämisse Verwendung findet. Dann kann das Teiltableau auch an jeden anderen Teilast unterhalb einer der Extensionen E_j ($j \neq i$) oder sogar direkt an den Ast B angefügt werden.

Es gibt mehrere mögliche Wege, eine solche Situation auszunutzen:

1. Der Kalkül kann verstärkt werden, indem man die Tableauregel so verändert, daß sie erlaubt, $\perp$ zu allen Teilästen unterhalb der Extensionen E_j hinzuzufügen ($j \neq i$); dadurch wird der Kalkül jedoch nicht-gutartig, weil $\perp$ in diesem Fall nicht aus Prämissen auf den erweiterten Ästen abgeleitet wird.

2. Man kann den Kalkül so verändern, daß es möglich wird, die Entscheidung aufzuschieben, welches die nächste Konklusion sein soll, die zur Erweiterung eines Astes verwendet wird. Die Konklusion C wird dann nur „vorläufig" zur Erweiterung von B verwendet. Nur wenn später Formeln aus allen ihren Extensionen zum Abschluß von Ästen verwendet werden, wird die Entscheidung getroffen, daß C tatsächlich für die Erweiterung von B verwendet werden soll; andernfalls wird die Entscheidung getroffen, daß C nicht verwendet werden soll, die vorläufige Erweiterung mit C wird rückgängig gemacht, und das geschlossene Teiltableau unterhalb E_i wird direkt an B angefügt. Aber auch dies macht den Kalkül nicht-gutartig, weil in einem gutartigen Kalkül eine Tableauformeln nicht wieder entfernt werden kann, wenn sie erst einmal zum Tableau hinzugefügt worden ist.

3. Man läßt den Kalkül unverändert; die Information, daß das geschlossene Teiltableau unterhalb E_i konstruiert werden kann, ohne Formeln aus E_i in Prämissen zu verwenden, wird ausgenutzt, indem dieses Teiltableau an jeden noch offenen Ast unterhalb der Extensionen E_j ($j \neq i$) angefügt wird. Dann bleibt der Kalkül gutartig; und Pruning wird als eine Technik zur deterministischen Konstruktion geschlossener Teiltableaus aufgefaßt. Eine Implementierung muß das geschlossene Teiltableau natürlich nicht wirklich wiederholt konstruieren; die Information, daß (und wie) das möglich wäre, genügt.[6]

[5] Von engl. *to prune*: abschneiden, zurückschneiden.

[6] Man beachte jedoch, daß man annehmen muß, das Teiltableau sei (zumindest implizit) unter jedem offenen Teilast unterhalb einer der Extensionen E_j vorhanden, wenn die Regularität einer Erweiterungsregelanwendung überprüft wird (siehe Abschnitt 5.2).

Definition 4.6.1 Sei $\mathcal{C}$ ein gutartiger Kalkül für eine Logik **L**; sei $\Sigma \in Sig$ eine Signatur; und sei $T_0, \ldots, T_n$ eine Folge von Tableaus für eine Menge $G \subset Form(\Sigma)$ von Formeln, so daß T_{i+1} ein Nachfolgetableau von T_i ist ($1 \leq i < n$).

Ein Vorkommen einer Tableauformel ϕ in einem Knoten N eines Tableau T_i ist *benutzt* für die Konstruktion von $T_{i+1}, \ldots, T_n$, falls

1. $\phi = \bot$ (d. h., alle Vorkommen der Tableauformel $\bot$ gelten als benutzt); oder
2. es ein Tableau T_j ($i \leq j \leq n-1$) und einen Ast B_j durch den N entsprechenden Knoten N_j in T_j gibt, so daß T_{j+1} aus T_j durch Erweiterung des Astes B_j konstruiert ist, wobei eine ϕ enthaltende Prämisse und eine Konklusion
$$C_j = \{E_1, \ldots, E_k\}$$
verwendet ist und ein Vorkommen in T_{j+1} von zumindest je einer Formel in jeder der Extensionen $E_1, \ldots, E_k$ für die Konstruktion von $T_{j+1}, \ldots, T_n$ benutzt ist. □

Wenn es mehr als ein Vorkommen derselben Formel auf einem Ast B gibt, von denen eines zur Erweiterung von B benutzt ist, dann gelten sie gemäß Definition alle als benutzt. Man könnte statt dessen auch nur eines der Vorkommen als benutzt ansehen (bevorzugt dasjenige, das der Wurzel des Tableaus am nächsten ist); jedoch sollten zu viele Vorkommen derselben Formel auf einem Ast ohnehin vermieden werden (siehe Abschnitt 5.2).

Beispiel 4.6.2 Wir betrachten das in Abbildung 4.6 (a) dargestellt Tableau; wir nehmen an, zur Konstruktion des Tableaus sei zunächst die Konklusion $\{E_1, E_2\}$ der Prämisse Π verwendet worden, dann die Konklusion $\{E'_1, E'_2, E'_3\}$ der Prämisse $\Pi' = E_1$ und schließlich seien das geschlossene Teiltableau T_1 an den Ast unterhalb E'_1 und das geschlossene T_2 an den Ast unterhalb E'_2 angefügt worden; die Äste, die die Extensionen E'_3 und E_2 enthalten, seien noch offen.

Weiter nehmen wir an, daß keine der Formeln in E_1 für die Konstruktion des Teiltableaus T_1 benutzt worden ist und daß keine der Formeln in E_1 und E'_2 für die Konstruktion von T_2 benutzt worden ist. Dann gelten per Definition die Formeln in Π als unbenutzt. Ob Formeln in E'_1 für die Konstruktion von T_1 benutzt sind, ist irrelevant.

Wenn die erste der Möglichkeit, Pruning auszuführen, verwendet wird, dann kann $\bot$ zu den verbleibenden offenen Ästen hinzugefügt werden (Abbildung 4.6 (b)). Wird die zweite Möglichkeit verwendet, werden also alle Tableauerweiterungen, die sich als unnötig erweisen, zurückgenommen, dann entsteht das in Abbildung 4.6 (c) dargestellt Tableau. Und das Tableau aus Abbildung 4.6 (c) ist das Ergebnis der dritten Möglichkeit, Pruning auszuführen, die die Gutartigkeit des Kalküls erhält. Man beachte, daß die verschiedenen Pruning-Techniken alle zweimal ausgeführt werden müssen, nämlich einmal für jede der beiden Erweiterungsregelanwendungen, mit denen die unbenutzten Extensionen E_1 bzw. E'_2 zum Tableau hinzugefügt wurden. □

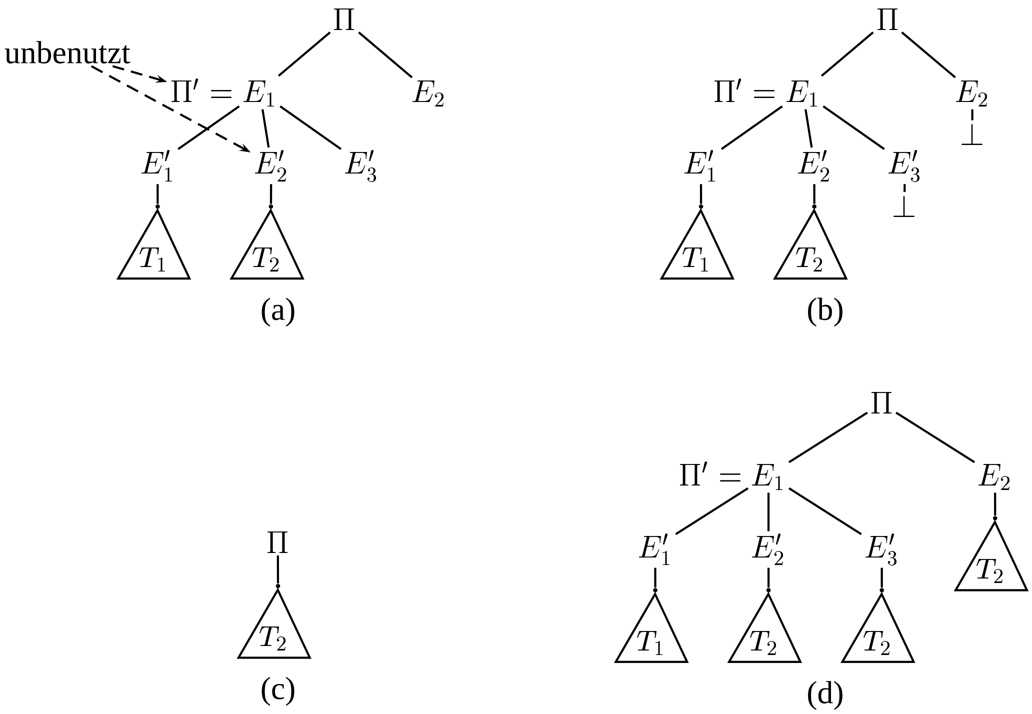


Abbildung 4.6: Anwendung der Pruning-Technik (Beispiel 4.6.2).

4.7 Zusätzliche Regelschemata

Eine einfache aber oft effektive Weise, einen Kalkül zu verstärken, ist die Einführung zusätzlicher Erweiterungsregel-Schemata. Wenn eine bestimmte Folge von Regelanwendungen häufig auftritt, ist es sinnvoll, die Tableauregel so zu erweitern, daß eine einzige Regelanwendung den gleichen Effekt hat, wie die häufig auftretenden Folge von Anwendungen.

Beispiel 4.7.1 Nehmen wir an, daß in einem bestimmten Anwendungsgebiet häufig Formeln der Form $\mathsf{T}{:}(true \to \phi)$ vorkommen, was z. B. auftreten kann, wenn die zu beweisenden Formeln automatisch generiert werden.

In diesem Fall ist es nützliche, den Kalkül $\mathcal{C}_{\mathrm{PL1}}$ für PL1 um ein zusätzliches Schema

$$\frac{\mathsf{T}{:}(true \to \phi)}{\mathsf{T}{:}\phi}$$

zu erweitern, so daß $\mathsf{T}{:}\phi$ in einem Schritt abgeleitet werden kann, ohne den Ast zuerst mit der Konklusion $\{\{\mathsf{F}{:}true\}, \{\mathsf{T}{:}\phi\}\}$ erweitern zu müssen, um dann einen der beiden neuen Teiläste durch Ableitung von $\perp$ aus der Prämisse $\{\mathsf{F}{:}true\}$ abzuschließen. $\square$

5 Konstruktion effizienter Beweisprozeduren

5.1 Einführung

5.1.1 Suchbäume

Im vorangegangenen Kapitel wurden Methoden zur Verbesserung eines Tableaukalküls diskutiert, die es ermöglichen, kürzere Beweise zu erzeugen. Das Thema dieses Kapitels ist nun, wie man auf effiziente Weise in dem verbleibenden kleineren Suchraum nach einem Beweis suchen kann.

Der Suchraum läßt sich als Suchbaum veranschaulichen, wobei jede mögliche Wahl der nächsten Erweiterungsregelanwendung auf ein Tableau T einen Knoten im Suchbaum mit sovielen Nachfolgern erzeugt, wie T Nachfolgetableaus hat. Der Suchbaum eines *beweiskonfluenten* Kalküls (der keine Sackgassen enthält) ist in Abbildung 5.1 schematisch dargestellt; dort sind alle Pfade entweder unendlich oder enden in einem Beweis (durch einen Stern $\star$ angedeutet).

Zwei Konzepte für die Beweissuche sind zu unterscheiden: *Tiefensuche* und *Breitensuche*. Damit Tiefensuche möglich ist, darf es entweder keine Pfade im Suchbaum geben, die keinen Beweis enthalten, oder es muß möglich sein, solche Pfade zu vermeiden, indem man bei der Konstruktion von Tableaus *Fairneß*strategien anwendet. Bei beweiskonfluenten Kalkülen sind es die unendlichen Pfade im Suchbaum, die vermieden werden müssen.

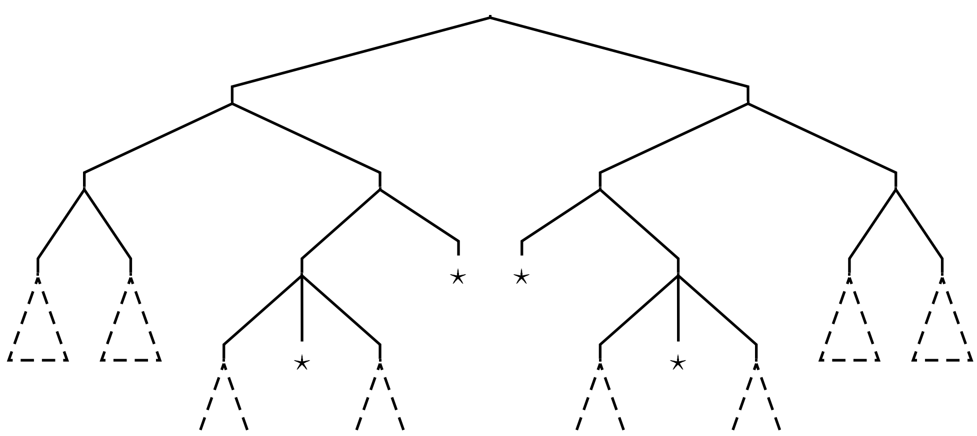


Abbildung 5.1: Suchbaum eines beweiskonfluenten Kalküls.

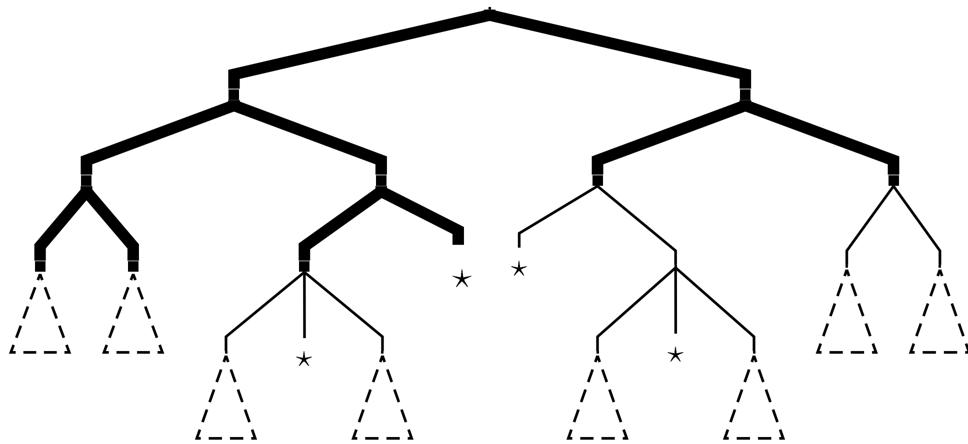


Abbildung 5.2: Breitensuche.

5.1.2 Breitensuche und Iterative deepening

Da Fairneßstrategien, die Tiefensuche ermöglichen, für Tableaukalküle mit starren Variablen schwer zu konstruieren sind, verwenden die meisten automatischen Deduktionssysteme Breitensuche. Mit Breitensuche können kürzere Beweise als mit Tiefensuche gefunden werden, weil alle Pfade des Suchbaums abgesucht werden, während mit Tiefensuche ein Pfad, der einen kurzen Beweis enthält, unter Umständen nicht betrachtet wird; Fairneßstrategien stellen nur sicher, daß *irgendein* Beweis gefunden wird, es muß aber nicht der kürzeste sein. Dabei ist zu bedenken, daß die Länge der gefundenen Beweise bei automatischer Deduktion nicht von großer Bedeutung ist (der einzige Vorteil kurzer Beweise ist, daß sie weniger Regelanwendungen erfordern, und daher leichter zu finden sind). Breitensuche ist sehr viel „teurer" als Tiefensuche, weil benachbarte Pfade des Suchbaums viele ähnliche oder sogar identische Tableaus enthalten, die bei Breitensuche alle betrachtet werden. Dieser Nachteil der Breitensuche überwiegt bei weitem jeden Vorteil, den sie haben mag. In Abbildung 5.2 ist schematisch dargestellt, welcher Teil des Suchbaums bei Breitensuche durchsucht wird.

Für alle (praktikablen) Vervollständigungsmodi, d. h., alle (monotonen) Funktionen m von $\mathbb{N}$ in die Menge aller Tableaus, so daß $\bigcup_{i \in \mathbb{N}} m(i)$ alle konstruierbaren Tableaus enthält, wächst die Größe $|m(i)|$ des Suchbaums exponentiell in i. Es ist – selbst für kleine i – in der Regel nicht möglich, alle Tableaus in $m(i)$ im Speicher zu halten. Darum verwenden die meisten Implementierungen (siehe z. B. (Beckert & Posegga, 1995)) Tiefensuche mit *Iterative deepening* (TsID) (Korf, 1985): Der partielle, endliche Suchraum, der aus allen Tableaus in $M(i) = \bigcup_{j \leq i} m(j)$ für ein $i \in \mathbb{N}$ besteht, wird mit Hilfe von Tiefensuche und Backtracking abgesucht; und wenn sich herausstellt, daß er keinen Beweis enthält, dann wird i erhöht. Die Tableaus in $M(i)$ stehen für die Konstruktion der Tableaus in $M(i+1)$ nicht zur Verfügung; sie müssen wieder neu konstruiert werden, was jedoch im Vergleich zur Breitensuche auf der „richtigen" Ebene i nur einen polynomiellen Overhead mit sich bringt, weil $M(i+1)$ exponentiell

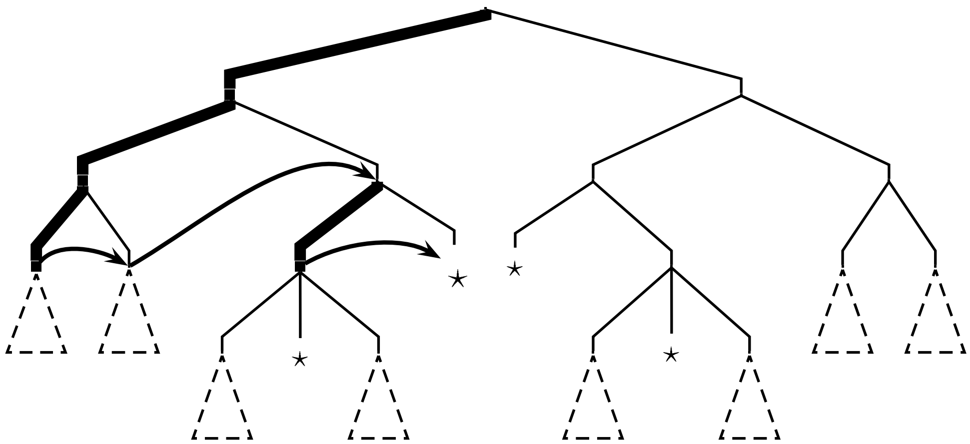


Abbildung 5.3: Tiefensuche mit Iterative deepening.

größer ist als $M(i)$. Obwohl TsID zu einer akzeptablen Performanz tableaubasierter Theorembeweiser führt, sollte man betonen, daß sie nur ein Kompromiß ist, solange keine vollständige Fairneßstrategie für Tiefensuche zur Verfügung steht.

In Abbildung 5.3 ist schematisch dargestellt, welcher Teil des Suchbaums bei TsID durchsucht wird.

5.1.3 Tiefensuche mit Fairneßstrategien

Der Vorteil von Tiefensuche mit Fairneß ist, daß die Menge der Informationen, die die Tableaus darstellen, bei jedem Beweisschritt zunimmt; es geht keine Information verloren, da es kein Backtracking gibt. Außerdem wird vermieden, daß gleiche Tableaus oder Folgen von Tableaus in verschiedenen Pfaden des Suchbaums mehrfach betrachtet werden.

In Abbildung 5.4 ist schematisch dargestellt, welcher Teil des Suchbaums bei Tiefensuche mit Fairneßstrategie durchsucht wird – nämlich nur ein Ast, der in einem Beweis endet. Durch die Fairneßstrategie werden die unendlichen Äste ohne Beweis vermieden. Allerdings kann der Beweis, der durch Tiefensuche mit Fairneß gefunden wird, (sehr viel) länger sein und tiefer im Suchbaum liegen, als der kürzeste Beweis.

Abbildung 5.5 veranschaulicht die unterschiedliche Struktur des Suchraumes bei den verschiedenen Suchstrategien. Dort ist der Teil des Suchraums, der untersucht worden ist, bevor der erste Beweis gefunden wird, eingefärbt. Die Form des Suchraums deutet dessen exponentielles Wachstum nach unten an.

Im Fall von Grundkalkülen ist es relativ leicht, Tiefensuche zu verwenden. Deren Erweiterungsregeln sind nicht-destruktiv; darum genügt es, systematisch alle möglichen Konklusionen zu verwenden, bis alle Äste des konstruierten Tableaus entweder voll expandiert oder geschlossen sind.

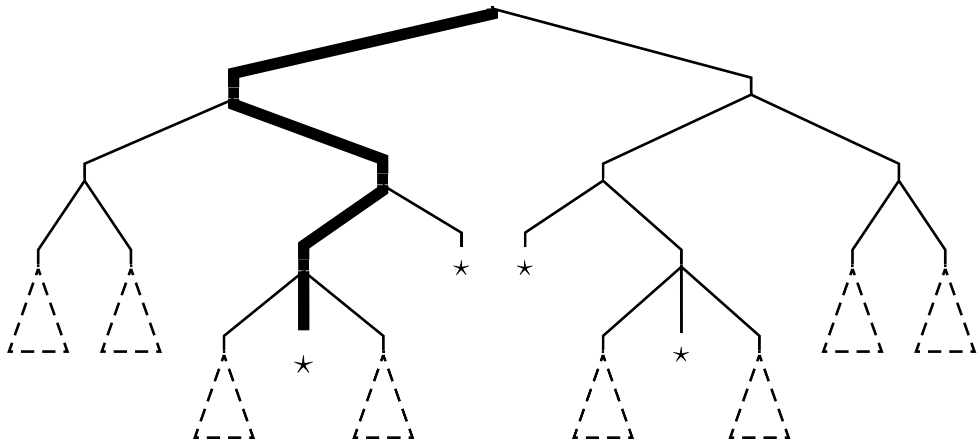


Abbildung 5.4: Tiefensuche mit Fairneßstrategie.

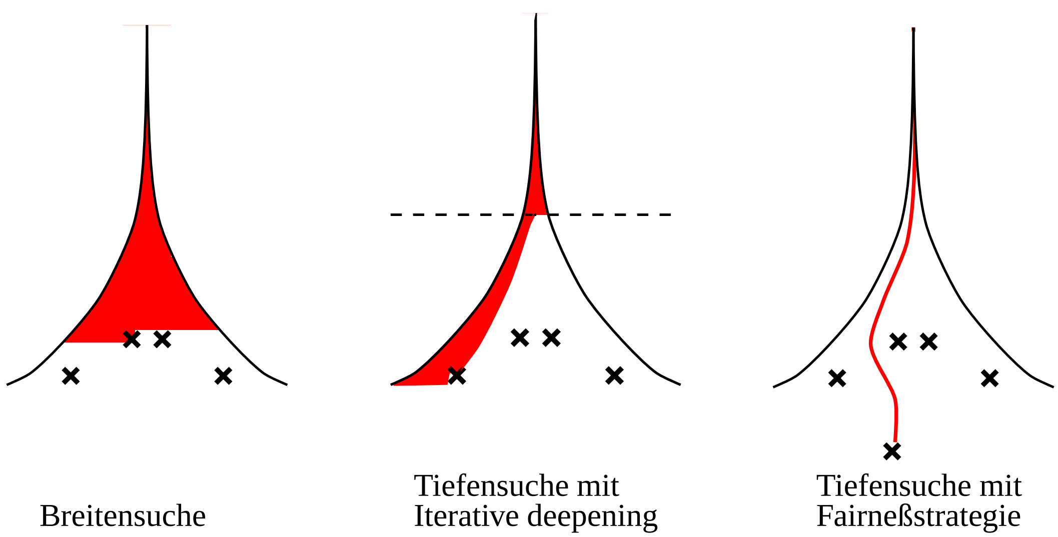


Abbildung 5.5: Vergleich der verschiedenen Suchstrategien.

Bei Kalkülen mit starren Variablen ist die Situation sehr viel komplizierter, denn sie sind destruktiv (auch wenn sie beweiskonfluent sind). Die Anwendung einer Substitution kann Formeln auf einem Tableau zerstören, die für den Beweis benötigt werden, und darum wieder hergestellt werden müssen, indem sie noch einmal aus ihren Prämissen abgeleitet werden.

Bis jetzt gab es keine praktikable Antwort auf die Frage, wie eine deterministische Beweisprozedur, die Tiefensuche verwendet und vollständig ist, für einen Kalkül mit starren Variablen zu konstruieren sei. Solche Prozeduren waren nur für den Spezialfall nicht-destruktiver Kalküle mit starren Variablen bekannt, bei denen Äste ohne die Instantiierung von Variablen erweitert werden und schließlich nur eine einzige Substitution angewendet wird, von der bekannt ist, daß mit ihre alle Äste zugleich geschlossen werden können.

In diesem Kapitel wird das Problem analysiert, eine deterministische Beweisprozedur für Kalküle mit starren Variablen zu konstruieren; und eine Lösung wird vorgestellt für den allgemeinen Fall beliebiger starre Variablen verwendender Kalküle, die gutartig sind (und also insbesondere beweiskonfluent). Keine anderen Annahmen werden gemacht; insbesondere ist die Methode nicht auf Kalküle für bestimmte Logiken oder Formeln in einer bestimmten Normalform (wie etwa Klauselnormalform) beschränkt.

Die deterministische Suchstrategie, die wir vorschlagen, basiert auf:

- *Regularität* (Abschnitt 5.2), um sicherzustellen, daß keine „Zyklen" in der Suche auftreten (es also nicht möglich ist, die gleichen Formeln oder Teiltableaus wieder und wieder abzuleiten),
- *Gewichtsordnungen* (Abschnitt 5.3), wobei jeder Tableauformel ein „Gewicht" in solcher Weise zugeordnet wird, daß es (bis auf Variablenumbenennung) nur endlich viele verschiedene Formeln mit einem bestimmten Gewicht gibt. Wenn dann Tableauformeln mit kleinerem Gewicht zuerst abgeleitet werden, wird früher oder später jede Konklusion zu allen Ästen hinzugefügt werden, die ihre Prämisse enthalten, so daß diese Strategie also fair ist.
- *Rekonstruktionsschritten* (Abschnitt 5.4), die den Kalkül in gewissem Sinne „schwach" nicht-destruktiv machen. Direkt im Anschluß an einen destruktiven Erweiterungsschritt, werden die Regelanwendungen ausgeführt, die notwendig sind, um die zerstörten Formeln zu rekonstruieren.

Die wesentliche Schwierigkeit ist, eine Regularitätsbedingung zu definieren, die auf der einen Seite restriktiv genug ist, um Zyklen in der Beweiskonstruktion zu vermeiden, und auf der anderen Seite aber auch nicht zu restriktiv ist, so daß die Vollständigkeit erhalten bleibt.[1]

[1] Baumgartner (1998) schlägt eine Regularitätsbedingung vor, die (ein wenig) zu restriktiv ist, so daß zwar Zyklen in der Beweissuche vermieden werden, andererseits aber Tiefensuche *mit Iterative deepening* verwendet werden muß, um die Vollständigkeit sicherzustellen.

Unsere Fairneßstrategie berücksichtigt das ganze Tableau (und nicht nur einen einzelnen Ast), und zwar sowohl wenn es darum geht, die Regularität zu überprüfen, als auch bei der Auswahl einer Konklusion mit minimalem Gewicht. Eine auf dieser Strategie beruhende Prozedur kann jederzeit jeden Ast des Tableaus erweitern. Das bedeutet keinen besonders großen Speicherverbrauch; zumindest ist er nicht höher als der einer Beweisprozedur, die zwar immer nur einen „aktuellen" Ast erweitert bis er geschlossen ist und erst dann andere Äste betrachtet, die aber Iterative deepening benutzt, um die Vollständigkeit sicherzustellen, und also alle schon geschlossenen Äste für den Fall, daß Backtracking eintritt, speichern muß.

5.1.4 Wann ist eine Beweisprozedur praktikabel?

Wie oben schon gesagt, gab es bisher keine *praktikablen* Beweisprozeduren für Kalküle mit starren Variablen. Unter „praktikabel" ist zu verstehen, daß die Berechnungskomplexität der Entscheidung, was die nächste Erweiterungsregelanwendung in einer bestimmten Situation sein soll, nicht unangemessen groß ist. Außerdem soll die Zahl der Erweiterungsschritte, die notwendig sind, um einen Beweis zu finden, in einem angemessenen Verhältnis zur Zahl der notwendigen Schritte stehen, wenn Breitensuche verwendet wird.

Trivialerweise existiert eine (nicht-praktikable) deterministische Beweisprozedur für alle monotonen Kalküle – nämlich die triviale Prozedur, die eine Breitensuche im Hintergrund durchführt. Bis ein Beweis (im Hintergrund) gefunden ist, wählt sie (im Vordergrund) beliebige Erweiterungsregelanwendungen. Wenn ein Beweis gefunden ist, fügt sie diesen als Teiltableau an alle offenen Äste an, was möglich ist, da der Kalkül monoton ist. Solch ein Prozedur ist natürlich völlig nutzlos. Man beachte, daß eine solch sinnlose Prozedur selbst dann definiert werden kann, wenn die Berechnungskomplexität der Erweiterungsschritte beispielsweise darauf eingeschränkt wird, höchstens polynomiell in der Größe des erweiterten Tableaus zu sein.

Wenn die in den folgenden Abschnitten vorgestellte Fairneßstrategie verwendet wird, ist die Komplexität der Entscheidung, welches der nächste Erweiterungsschritt sein soll, polynomiell in der Größe des expandierten Tableaus bzw. der Größe seiner möglichen Nachfolgetableaus. Im Mittel ist die Komplexität jedoch sehr viel geringer, weil nur diejenigen Teile eines Tableau betrachtet werden müssen, die durch die jeweils Erweiterung verändert werden.

Die Größe der Tableaubeweise, die mit dieser Fairneßstrategie gefunden werden (und die Zahl der Erweiterungsschritte) ist höchstens die der Tableaubeweise, die mit TsID im schlechtesten Fall konstruiert werden (also in dem Fall, daß zufällig alle Äste des Suchbaums, die keinen Beweis haben, zuerst betrachtet werden).

5.2 Regularität

Der Begriff der *Regularität* ist aus dem Tableaubeweisen in klassischer Logik in Klauselnormalform (Letz *et al.*, 1992) und Negationsnormalform (Hähnle & Klingenbeck, 1996; Hähnle *et al.*, 1997) bekannt; er wird in (Beckert & Hähnle, 1998) auf Nicht-Normalform-Tableaus für die Prädikatenlogik erster Stufe erweitert.

Im folgenden definieren wir ein Regularitätskonzept für beliebige gutartige Kalküle mit starren Variablen. Dieses Konzept unterscheidet sich von dem klassischen Begriff der Regularität ganz wesentlich dadurch, daß das ganze Tableau betrachtet wird und nicht nur ein einzelner Ast; es erweist sich als geeignet zur Konstruktion einer deterministischen Beweisprozedur für Kalküle mit starren Variablen.[2]

Nehmen wir an, eine Folge $T_1, \ldots, T_n$ von Tableaus sei bereits konstruiert. Wir sagen, eine Erweiterungsregelanwendung auf T_n sei *irregulär*, wenn das Nachfolgetableau T_{n+1} in einem der Vorgängertableaus T_j *enthalten* ist – insbesondere, wenn T_{n+1} in T_n enthalten ist. In diesem Fall stellt die Folge $T_j, \ldots, T_{n+1}$ einen Zyklus in der Beweissuche dar, weil T_{n+1} keine Information enthält, die nicht auch schon in T_j vorhanden sind.

Nun bleibt natürlich die Frage zu beantworten, wann ein Tableau ein anderes enthält. Dazu legen wir zunächst fest, daß ein Tableau T_j ein Tableau T_{n+1} enthält, wenn jeder Ast von T_j einen Ast von T_{n+1} enthält. Intuitiv ist in diesem Fall das Tableau T_{n+1} redundant, denn wenn geschlossene Teiltableaus unter allen seinen Ästen konstruiert werden können, dann ist es auch möglich, geschlossene Teiltableaus unter allen Ästen von T_j zu konstruieren, da ja jeder einen der Äste von T_{n+1} enthält.

Jetzt bleibt allerdings die Frage, wann ein Tableau*ast* einen anderen Ast enthält. In erster Annäherung können wir sagen, daß ein Ast B einen Ast B' enthält, wenn eine Variante jeder Tableauformel in B' in B vorkommt. Das ist allerdings eine Übervereinfachung. Drei Aspekte müssen zusätzlich beachtet werden:

Erster zusätzlicher Aspekt Es reicht unter Umständen *nicht* aus, wenn der Ast B nur *eine* Variante einer Tableauformel ϕ enthält, obwohl in B' mehrere Varianten von ϕ vorkommen. Es kann nämlich sein, daß *mehrere* Varianten einer Formel für eine bestimmte Ableitung benötigt werden. Wenn allerdings die Größe minimaler Prämissen in einem Kalkül beschränkt ist, minimale Prämissen also höchstens k Formeln enthalten (für ein festes $k \in \mathbb{N}$), dann reicht es aus, wenn der Ast B *eine* Variante jeder höchstens k Formeln enthaltenden Teilmenge von $Form(B')$ enthält – was insbesondere impliziert, daß es genügt, wenn B nicht mehr als k *variablendisjunkte* Varianten einer Formel enthält. Man beachte jedoch die unten beschriebenen weiteren zu berücksichtigenden Aspekte, die dazu führen können, daß (a) Formeln, die (auf den

[2] Kalküle mit universellen Variablen werden hier nicht betrachtet, da sie sich bezüglich Regularität wie Grundkalküle verhalten.

ersten Blick) Varianten voneinander sind, als wesentlich verschieden aufgefaßt werden müssen und daß (b) wegen der Interaktion verschiedener Äste, k Varianten unter Umständen doch nicht ausreichend sind.

Beispiel 5.2.1 Minimale Prämissen bestehen in dem Kalkül für PL1 aus Abschnitt 3.6 aus höchstens $k = 2$ Formeln. Ein Ast B enthält darum einen Ast B' unter Umständen *nicht*, selbst wenn er *eine* Variante jeder Formel auf B' enthält.

Besteht beispielsweise der Ast B' aus den beiden Formeln $\phi = \mathsf{T}{:}(\neg p(X) \wedge p(f(X)))$ und $\phi' = \mathsf{T}{:}(\neg p(X') \wedge p(f(X')))$, während B nur aus der Formel ϕ besteht, dann enthält zwar B eine Variante jeder Formel auf B', dennoch stellt aber der Übergang von B zu B' sicherlich keinen Zyklus in der Beweissuche dar, weil nämlich der Ast B' geschlossen werden kann, der Ast B aber nicht; die zweite Variante ϕ' ist notwendig. □

Wir beschränken uns also im folgenden auf solche Kalküle, bei denen minimale Prämissen nicht von beliebiger Größe sind, sondern in ihrer Kardinalität durch eine Konstante $k \in \mathbb{N}$ beschränkt sind. Wie andere Kalküle, d. h., solche mit minimalen Prämissen beliebiger Größe gehandhabt werden können, ist am Ende von Abschnitt 5.4 beschrieben.

Zweiter zusätzlicher Aspekt Bei der Beantwortung der Frage, ob ein Ast B einen Ast B' enthält, spielen nicht nur die Formeln auf B und B' eine Rolle, sondern auch die *assoziierten* Formeln auf anderen Ästen, die mit den Formeln auf B bzw. B' starre Variablen gemeinsam haben.

Definition 5.2.2 Sind ϕ und ψ Tableauformeln in einem Tableau T mit starren Variablen und gibt es eine starre Variable, die sowohl in ϕ als auch in ψ vorkommt, dann heißt ψ mit ϕ *assoziiert*. Die Menge aller Tableauformeln in T, die mit ϕ assoziiert sind, ohne ϕ selbst, sei mit $Assoc(T, \phi)$ bezeichnet.

Ist Φ eine Menge von Tableauformeln in T, so sei

$$Assoc(T, \Phi) = (\bigcup_{\phi \in \Phi} Assoc(T, \phi)) \setminus \Phi \quad .$$

□

Der Grund dafür, daß assoziierte Formeln berücksichtigt werden müssen, ist, daß ihr Vorhandensein sich bei der Anwendung von Substitutionen wegen der gemeinsamen Variablen auswirken kann. So werden wir zur Konstruktion einer deterministischen Beweisprozedur in Abschnitt 5.4 Regularität mit der Einschränkung kombinieren, daß jeweils nur eine solche von mehreren möglichen Konklusionen verwendet werden darf, die bezüglich einer *Gewichtsordnung* (Abschnitt 5.3) minimale Formeln erzeugt. Nehmen wir an, der Ast B enthalte eine Variante $\phi(X)$ einer Formel $\phi(X')$ auf B', und die Variable X komme in dem Tableau T noch in einer assoziierten Formel $\psi(X)$ vor, während es in dem Tableau T' kein weiteres Vorkommen von X' gebe. Nun kann es sein, daß eine Erweiterungsregelanwendung die Instantiierung von X bzw. X' mit einem Term t erfordert. Die entstehende Formel $\phi(t)$ wird in beiden Tableaus von gleichem Gewicht sein. In T wird aber zusätzlich die Formel $\psi(t)$ erzeugt; diese kann ein so großes Gewicht haben (insbesondere ein größeres als $\phi(t)$), daß sich dadurch die Regelanwendung auf das Tableau T verbietet – während die entsprechende Anwendung auf T' möglich ist. In solch einem Fall enthält der Ast B den Ast B' also *nicht* (es sei denn er enthält noch eine weitere Variante von $\psi(X')$, mit der keine Formeln in T assoziiert sind).

Dritter zusätzlicher Aspekt Wie schon gesagt, enthält ein Tableau T ein Tableau T', wenn es für jeden Ast B in T einen Ast B' in T' gibt, so daß B den Ast B' enthält. Dabei ist es durchaus möglich, daß zwei verschiedenen Ästen B_1 und B_2 von T derselbe Ast B' zugeordnet wird. In diesem Fall gibt es für jede Menge Φ' von k Formeln auf B' eine Formelmenge Φ_1 auf B_1 und eine Formelmenge Φ_2 auf B_2, die Varianten von Φ' sind.

Die grundlegende Idee der Enthaltenseins-Relation impliziert, daß jede Regelanwendung, die auf dem Ast B' mit der Prämisse Φ' möglich ist, auch – gleichzeitig – auf den beiden B' enthaltenden Ästen B_1 und B_2 mit den Prämissen Φ_1 bzw. Φ_2 möglich sein soll. Dafür müssen aber die beiden Variablenumbenennungen, die Φ aus Φ_1 bzw. aus Φ_2 erzeugen, kompatibel sein. Entsprechendes gilt, wenn der Ast B' mehr als zwei Ästen in T zugeordnet ist.

Definition und Eigenschaften der Enthaltenseins-Relation

Definition 5.2.3 Sei $\mathcal{C}$ ein Kalkül mit starren Variablen; sei Σ_{fv} eine Signatur; seien T und T' Tableaus über Σ^*_{fv}, die keine starren Variablen gemeinsam haben; und sei $k \in \mathbb{N}$ eine natürliche Zahl.

Das Tableau T *k-enthält* das Tableau T', in Zeichen $T' \subseteq_k T$, wenn

i. jedem Ast B von T ein Ast B' von T' zugeordnet werden kann

ii. und dann – jeweils für jedes Paar B, B' – jeder Menge $\Phi' \subseteq \mathit{Form}(B')$ von Tableauformeln auf B', die höchstens k Elemente hat, eine Menge $\Phi \subseteq \mathit{Form}(B)$ von Tableauformeln auf B und eine Variablenumbenennung $\pi \subset \mathit{Subst}(\Sigma_{\mathrm{fv}})$ zugeordnet werden kann,

so daß:

1. Für Φ, Φ' und π gilt jeweils:
   (a) $\Phi\pi = \Phi'$;
   (b) für jede Tableauformel ψ in $Assoc(T, \Phi)$ gibt es (mindestens) eine Tableauformel ψ' in $Assoc(T', \Phi')$, so daß $\psi\pi$ und ψ' identisch sind bis auf die Umbenennung starrer Variablen, die *nicht* in $\Phi\pi$ bzw. Φ' vorkommen.
2. Ist ein Ast B' von T' verschiedenen Ästen $B_1, \ldots, B_s$ von T zugeordnet ($s \geq 2$), dann gilt für alle Φ' auf B', daß die Φ' im Zusammenhang mit $B_1, \ldots, B_s$ zugeordneten Variablenumbenennungen $\pi_1, \ldots, \pi_s$ in folgendem Sinne kompatibel sind: es gibt eine Substitution π, so daß $\pi_{|Free(\Phi) \cup Assoc(T,\Phi)} = \pi_i$ ($1 \leq i \leq s$).

Seien nun T und T' Tableaus über Σ^*_{fv}, die starre Variablen gemeinsam haben; und sei ρ eine Variablenumbenennung, so daß T und $T'\rho$ *keine* Variablen gemeinsam haben. Dann gelte $T' \subseteq_k T$, falls $T'\rho \subseteq_k T$. □

Enthält eine Tableau T ein Tableau T', dann ist jedem Ast B in T ein Ast B' in T' zugeordnet. Wir sagen dann auch, daß der Ast B den Ast B' enthalte, in Zeichen $B' \subseteq_k B$.

Beispiel 5.2.4 Sei $\Phi = \{\phi(X)\}$ und $\Phi' = \{\phi(X')\}$; ferner sei

$$Assoc(T, \Phi) = \{\psi(X, Y_1), \psi(X, Y_2)\} \ .$$

Bedingung 1 (a) in Definition 5.2.3 ist beispielsweise erfüllt, wenn

$$Assoc(T', \Phi') = \{\psi(X', Y')\} \ .$$

Sie ist *nicht* erfüllt, wenn $Assoc(T', \Phi') = \emptyset$; und sie ist auch nicht erfüllt, wenn

$$Assoc(T', \Phi') = \{\psi(Y', X')\} \ ,$$

da $\psi(X', Y_1)$ und $\psi(Y', X')$ gleich zu machen die Umbenennung der Variablen X' erfordert, die in Φ' vorkommt. □

Das folgende Lemma folgt sofort aus der Definition von $\subseteq_k$.

Lemma 5.2.5 *Die Relation $\subseteq_k$ auf Tableaus (Def. 5.2.3) ist transitiv und reflexiv.*

Wenn ein Tableau T' in einem Tableau T k-enthalten ist, dann ist T' in T k'-enthalten für alle $k' < k$.

Beispiel 5.2.6 Als einfaches Beispiel betrachten wir die Tableaus in Abbildung 5.6. Die Tableaus T_1 und T_2 1-enthalten die Tableaus T'_1, T'_2, T'_3; und das Tableau T_1 2-enthält die Tableaus T'_1, T'_2, T'_3. Aber das Tableau T_2 2-enthält nur T'_1. □

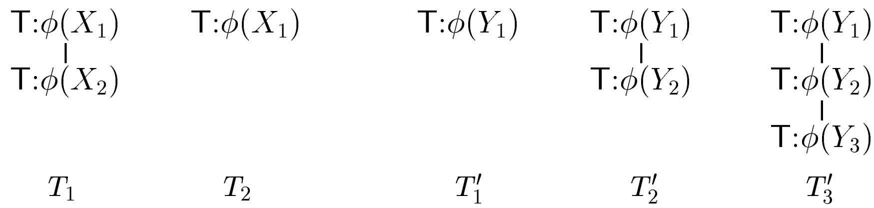


Abbildung 5.6: Die Tableaus aus Beispiel 5.2.6.

Abbildung 5.7: Tableaus, von denen keines das andere enthält (Beispiel 5.2.7).

Beispiel 5.2.7 Keines der Tableaus T_1 und T_2 in Abbildung 5.7 (a) bzw. (b) enthält das jeweils andere. Das Tableau T_1 enthält das Tableau T_2 nicht, weil der Ast von T_2 eine zusätzliche Formel enthält. Und T_2 enthält T_1 nicht, weil zwar eine Variante jeder Formelmenge des Astes von T_1 auf dem Ast von T_2 vorkommt, aber zu dem Element T:$r(X')$ von $Assoc(T_2,$ T:$q(X'))$ kein Gegenstück in $Assoc(T_1,$ T:$q(X))$ existiert, und also Bedingung 1 (a) in Definition 5.2.3 nicht erfüllt ist. □

Beispiel 5.2.8 Das Tableau T_1 in Abbildung 5.8 (a) 1-enthält das Tableau T_2 in Abbildung 5.8 (b). Aber T_2 enthält T_1 *nicht*, denn die Formeln T:$p(X')$ und T:$p(X')$ in T_2 sind miteinander assoziiert, während die entsprechenden Formeln in T_1 *nicht* miteinander assoziiert sind.

Der Übergang von T_2 zu T_1 stellt auch deutlich keinen Zyklus in der Beweissuche dar, weil das Tableau T_1 geschlossen werden kann, das Tableau T_2 aber nicht. □

Beispiel 5.2.9 Das Tableau T in Abbildung 5.9 (a) enthält das Tableau T' in Abbildung 5.9 (b) *nicht*. Das könnte nur sein, wenn die Äste B_1, B_2 von T *beide* den einzelnen Ast von T' enthielten. Zwar gibt es auf beiden Ästen mit T:$p(X,Y)$ bzw.

Abbildung 5.8: Nahezu identische Tableaus, von denen das eine (a) das andere (b) enthält, nicht aber umgekehrt (Beispiel 5.2.8).

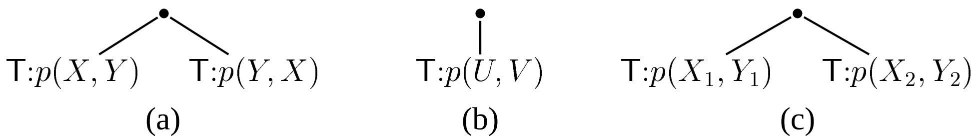


Abbildung 5.9: Beispiel für Bedingung 2 in der Definition der Enthaltenseins-Relation (siehe Beispiel 5.2.9).

$\mathsf{T}{:}p(Y,X)$ eine Variante der (einzigen) Formel $\mathsf{T}{:}p(U,V)$ im Ast von T'. Aber die erforderlichen Variablenumbenennungen $\{X \mapsto U,\, Y \mapsto V\}$ und $\{X \mapsto V,\, Y \mapsto U\}$ sind inkompatibel; sie haben keine gemeinsame Spezialisierung. Damit ist Bedingung 2 in der Definition der Enthaltenseins-Relation (Def. 5.2.3) verletzt.

Bei dem Tableau in Abbildung 5.9 (c) tritt dieses Problem nicht auf; es 1-enthält das Tableau T'. Die beiden erforderlichen Variablenumbenennungen $\{X_1 \mapsto U,\, Y_1 \mapsto V\}$ und $\{X_2 \mapsto U,\, Y_2 \mapsto V\}$ sind hier kompatibel. $\square$

Die Verwendung der Enthaltenseins-Relation macht nur Sinn, wenn sichergestellt ist, daß ein Tableau T sein Nachfolgetableau T' nur dann enthält, wenn der Übergang zu T' keinen Fortschritt in der Beweissuche bedeutet. Wie das folgende Beispiel zeigt, kann sich dabei ein Problem ergeben, wenn mehrere gleiche Regelanwendungen auf verschiedenen Äste von T möglich sind.

Beispiel 5.2.10 Bei dem in Abbildung 5.10 links gezeigten Tableau T_1 ist die gleiche Erweiterungsregelanwendung, die aus der Prämisse $\{\mathsf{F}{:}p, \mathsf{T}{:}p\}$ die Konklusion $\{\{\bot\}\}$ ableitet, auf beiden Ästen möglich; und um das Tableau abzuschließen muß diese Regelanwendung – nacheinander – auf beiden Ästen ausgeführt werden. Jedoch ist das Tableau T_2, das als Zwischenschritt nach einer Regelanwendung entsteht (Abbildung 5.10 mitte) in T_1 enthalten, denn beide Äste von T_1 enthalten den rechten, noch nicht expandierten Ast von T_2. Damit ist dieser Schritt irregulär. Das Tableau T_3, das durch die Erweiterung beider Äste entsteht (Abbildung 5.10 rechts), ist dagegen weder in T_2 noch in T_1 enthalten. Die Ableitung von T_3 aus T_1 ohne das Zwischenergebnis T_2 ist also als regulär anzusehen, und tatsächlich bedeutet sie auch einen Fortschritt in der Beweiskonstruktion. $\square$

Um dieses Problem zu umgehen, erweitern wir die Definition des Begriffs *Nachfolgetableau* und erlauben die Ausführung mehrerer gleicher Erweiterungsregelanwendungen auf verschiedenen Ästen eines Tableaus, um ein Nachfolgetableau zu konstruieren.

Definition 5.2.11 Sei $\mathcal{C}$ ein gutartiger Kalkül mit starren Variablen.

Ein Tableau T', das aus einem Tableau T durch mehrere in folgendem Sinne *gleiche* Erweiterungsregelanwendungen konstruiert werden kann, heiße ein *Nachfolgetableau* von T: Es gibt

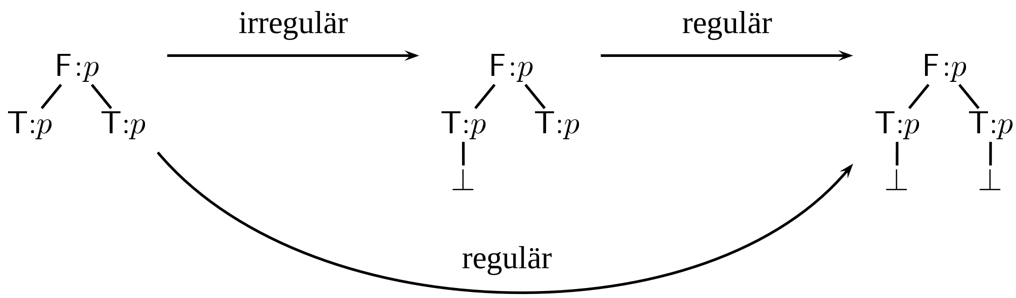


Abbildung 5.10: Beispiel, das die Notwendigkeit illustriert, mehrere gleiche Regelanwendungen als einen Schritt aufzufassen (siehe Beispiel 5.2.10).

– verschiedene Äste $B_1, \ldots, B_n$ $(n > 1)$ von T,

– Prämissen Π_i auf B_i $(1 \leq i \leq n)$, die bis auf Variablenumbenennung gleich sind,

– eine Konklusion $\langle C, \sigma \rangle$ mit $\Pi_i\sigma = \Pi_j\sigma$ $(1 \leq i, j \leq n)$,

so daß T' aus T dadurch entsteht, daß jeder der Äste B_i mit C erweitert wird und anschließend die Substitution σ angewendet wird. □

Nun können wir daran gehen, basierend auf dem Begriff des k-Enthaltenseins, unser Regularitätskonzept formal zu definieren.

Definition 5.2.12 Sei $\mathcal{C}$ ein gutartiger Kalkül mit starren Variablen; sei Σ eine Signatur; und sei $k \in \mathbb{N}$ eine natürliche Zahl.

Eine (endliche oder unendliche) Folge $(T_i)_{i \geq 0}$ von Tableaus (und insbesondere ein Tableaubeweis $T_1, \ldots, T_n$) für eine Menge $\mathfrak{F} \subset \mathit{Form}(\Sigma)$ von Formeln heiße k-*regulär*, wenn sie *keine* Tableaus T_j und T_i enthält, wobei $j < i$, so daß $T_i \subseteq_k T_j$ (Def. 5.2.3).

Eine Folge von Tableaus, die nicht k-regulär ist, heiße k-*irregulär*. □

Wenn eine Folge $T_1, \ldots, T_n$ von Tableaus k-regulär ist, T_{n+1} ein Nachfolgetableau von T_n ist und die Folge $T_1, \ldots, T_n, T_{n+1}$ k-irregular ist, dann nennen wir die Erweiterungsregelanwendung, durch die T_{n+1} aus T_n konstruiert wird k-irregulär (denn sie verursacht die Irregularität). Ob eine Erweiterungsregelanwendung regulär ist oder nicht, hängt von dem Kontext ab, in dem sie verwendet wird.

Um zu prüfen, ob eine Erweiterungsregelanwendung regulär ist, genügt es, die Teile des expandierten Tableaus zu betrachten, die von der Erweiterung betroffen sind, d. h., den erweiterten Ast und die mit seinen Formeln assoziierten Formeln des Tableaus. Für die Überprüfung der Regularität sind keine Unifikationstests erforderlich, weil nur Variablenumbenennungen betrachtet werden und keine Instantiierung von Variablen mit Termen; das heißt, die Regularität zu überprüfen, ist nicht so komplex wie ein Subsumtionstest (der NP-vollständig ist).

Man beachte auch, daß zwar assoziierte Formeln berücksichtigt werden müssen, es aber keine Rolle spielt, wo auf dem Tableau sie liegen. Die *Struktur* der Tableaus ist insofern also irrelevant und muß nicht verglichen werden.

Wenn der oben definierte Regularitäts-Begriff verwendet wird, um den Suchraum einzuschränken, indem nur reguläre Folgen von Tableaus zugelassen werden, dann bleibt die Vollständigkeit erhalten; das heißt, unser Regularitätskonzept ist nicht zu restriktiv.

Satz 5.2.13 *Sei $\mathcal{C}$ ein gutartiger Kalkül mit starren Variablen; sei Σ eine Signatur; und sei $k \in \mathbb{N}$ die maximale Größe aller minimalen Prämissen in $\mathcal{C}$.*

Wenn es einen Tableaubeweis für eine Menge $\mathfrak{F} \subset Form(\Sigma^*)$ von Formeln gibt, dann gibt es auch einen regulären Tableaubeweis für $\mathfrak{F}$.

Beweis: Dieser Satz folgt sofort aus Satz 5.4.4, der besagt, daß eine vollständige deterministische Beweisprozedur für alle gutartigen Kalküle mit starren Variablen existiert, die ausschließlich reguläre Tableaubeweise konstruiert. $\square$

Die Existenz einer vollständigen deterministischen Beweisprozedur, die ausschließlich reguläre Beweise konstruiert (Satz 5.4.4), impliziert nicht nur Satz 5.2.13; sie zeigt auch, daß unser Begriff der Regularität restriktiv genug ist, um seinen Zweck zu erfüllen.

Eine wichtige Klasse irregulärer Erweiterungsschritte ist die folgende: Nehmen wir an, ein Ast B_1 eines Tableaus T werde mit einer Konklusion $\langle C, \sigma \rangle$ erweitert; und ein Ast $B_2'\sigma$ in dem entstehenden Tableau T' sei in *allen* Ästen B von T enthalten, die von dem Erweiterungsschritt betroffen sind, d. h., in dem Ast B_1 (der erweitert wird) und in allen anderen Ästen, die von σ instantiierte Variablen enthalten. Dies ist insbesondere der Fall, wenn $B_2'\sigma$ in einem initialen Teilast S_0 von T enthalten ist, der oberhalb der ersten Vorkommen aller starrer Variablen im Definitionsbereich von σ endet.

Beispiel 5.2.14 Wir nehmen an, daß die Erweiterungsregel des Kalküls mit starren Variablen für PL1 aus Abschnitt 4.2.10 verwendet wird, um den Ast B_1 des in Abbildung 5.11 (a) dargestellten Tableaus T abzuschließen. Dabei wird die Konklusion $\langle \{\{\bot\}\}, \{X \mapsto a\} \rangle$ aus der Prämisse $\{\mathsf{F}{:}p(a), \mathsf{T}{:}p(X)\}$ abgeleitet; das Ergebnis ist das Tableau T' in Abbildung 5.11 (b).

Die Erweiterungsregelanwendung gehört zu der oben beschriebenen Klasse einfach zu erkennender irregulärer Anwendungen. Der rechte Ast $B_2'\sigma$ von T', dessen Knoten mit den Formeln $\mathsf{F}{:}p(a)$ und zweimal $\mathsf{T}{:}q(a)$ markiert sind, ist in dem initialen Teilast S_0 von T enthalten, dessen Knoten mit $\mathsf{F}{:}p(a)$ und $\mathsf{T}{:}q(a)$ markiert sind; und S_0 endet oberhalb des ersten Vorkommens von X in T, welches die einzige Variable ist, die von σ instantiiert wird.

Intuitiv ist diese Erweiterungsregelanwendung nutzlos, weil jedes geschlossene Teiltableau, daß unter $B_2'\sigma$ konstruiert werden kann, genauso sowohl unter B_1 als auch unter B_2 konstruiert werden kann. $\square$

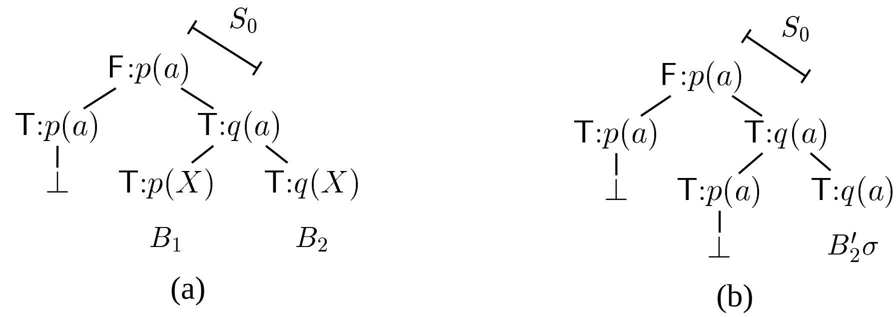


Abbildung 5.11: Eine irreguläre Erweiterungsregelanwendung (Beispiel 5.2.14).

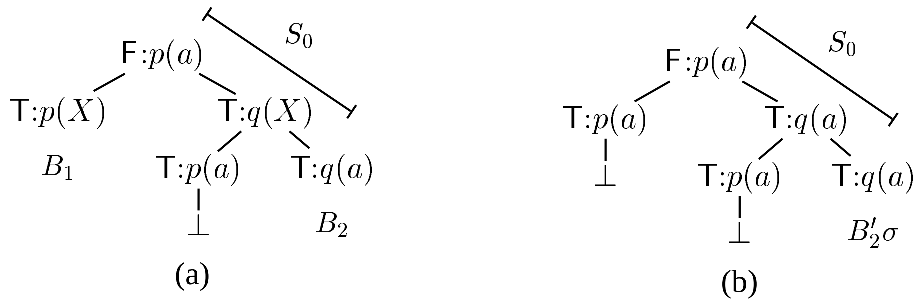


Abbildung 5.12: Beispiel für den Unterschied zwischen dem neuen und dem klassischen Begriff der Regularität (Beispiel 5.2.15).

Eine Erweiterungsregelanwendung wie oben beschrieben ist irregulär gemäß der Definition von Regularität, die gewöhnlich in der Literatur gegeben wird (z. B. in (Beckert & Hähnle, 1998)), wenn der Ast $B_2'\sigma$ die gleiche Extension mehrfach enthält; seine Beziehung zu anderen Ästen wird nicht weiter betrachtet.

Beispiel 5.2.15 Abbildung 5.12 illustriert den Unterschied zwischen unserem neuen und dem klassischen Regularitäts-Begriff. Die Situation ist der in Abbildung 5.11 dargestellten und in Beispiel 5.2.14 erläuterten sehr ähnlich. Nun aber endet der initiale Teilast S_0 von T, der den Ast $B_2'\sigma$ des Tableaus T' enthält, *nicht* oberhalb des ersten Vorkommens der Variablen X, die von σ instantiiert wird. Also gehört diese Erweiterungsregelanwendung nicht zu der Klasse einfach zu erkennender irregulärer Anwendungen; sie ist auch tatsächlich *regulär* gemäß unserer Definition von Regulärität, weil der Ast B_1 von T keinen der Äste von T' enthält.

Gemäß der klassischen Definition von Regularität ist diese Erweiterungsregelanwendung jedoch *irregulär*, weil der Ast $B_2'\sigma$ von T' die Extension $\{\mathsf{T}{:}q(a)\}$ zweimal enthält. $\square$

Wie Beispiel 5.2.15 zeigt, ist der klassische Regularitäts-Begriff in manchen Fällen restriktiver als unserer, was die Beweiskonfluenz zerstören kann.[3] Im Grundfall, wenn

[3] Ob die Verwendung des klassischen, restriktiveren Regularitäts-Begriffs tatsächlich die Beweiskon-

das Tableau keine freien Variablen enthält, besteht zwischen den beiden Regularitäts-Begriffen kein Unterschied.

Der Beweis von Satz 5.4.4, der der Hauptsatz dieses Kapitels ist, verwendet das folgende Lemma, in dem die Einschränkung, die die Regularität darstellt, beschrieben ist. Eine unendliche, reguläre Folge von Tableaus enthält unendlich viele verschiedene Tableauformeln; oder, was äquivalent ist, wenn eine reguläre Folge von Tableaus (bis auf Umbenennung starrer Variablen) nur endlich viele verschiedene Tableauformeln enthält, dann ist sie endlich.

Lemma 5.2.16 *Sei $\mathcal{C}$ ein gutartiger Kalkül mit starren Variablen; sei Σ eine Signatur; und sei $k \in \mathbb{N}$ eine natürliche Zahl. Ferner sei $\Gamma \subset \mathit{TabForm}(\Sigma^*)$ eine endliche Menge von Tableauformeln.*

Dann gibt es keine unendliche, reguläre Folge $(T_i)_{i \geq 0}$ von Tableaus, so daß für $i \geq 0$ alle Tableauformeln in T_i Varianten der Tableauformeln in Γ sind.

Beweis: Die Äquivalenzrelation $\sim_k$ auf Tableaus sei definiert durch:

$$T \sim_k T' \quad \text{genau dann, wenn} \quad T \subseteq_k T' \text{ und } T' \subseteq_k T \ .$$

Wir zeigen, daß es nur endlich viele Äquivalenzklassen (bzgl. $\sim_k$) von Tableaus geben kann, die aus Varianten der Formeln in Γ bestehen.

Da Γ endlich ist, gibt es nur endlich viele nicht nur bis auf Variablenumbenennung verschiedene Paare $\langle \Phi, \Psi \rangle$ von Tableauformelmengen, für die folgendes gilt:

- Alle Elemente von Φ und Ψ sind Varianten von Formeln in Γ;
- Φ hat höchstens k Elemente;
- jede Formel in Ψ hat mindestens eine Variable mit einer Formel in Φ gemeinsam;
- es gibt keine Formeln ψ, ψ' in Ψ, die bis auf Umbenennung nicht in Φ vorkommender Variablen identisch sind.

Sei r die maximal mögliche Zahl verschiedener, inkompatibler Umbenennungen der Variablen in Φ (diese hängt nur von der Zahl verschiedener Variablen in Φ ab); und sei $\widehat{\Gamma}_k$ eine Menge solcher wie oben beschriebenen Paare, die r verschiedene Repräsentanten jeder Klasse von bis auf Variablenumbenennung gleichen Paaren enthält; dabei sei die Menge $\widehat{\Gamma}_k$ so gewählt, daß alle ihre Elemente paarweise variablendisjunkt sind.

kluenz zerstört, hängt von den weiteren Einschränkungen ab, die verwendet werden; jedoch ist es noch niemandem gelungen, eine deterministische Beweisprozedur für starre Variablen verwendende Kalküle mit dem klassischen Begriff zu definieren. Und Grund dafür mag sehr wohl sein, daß er zu restriktiv ist, um die Beweiskonfluenz zu bewahren, wenn er mit anderen Einschränkungen kombiniert wird, die die Fairneß der Strategie sicherstellen (wie beispielsweise Gewichtsordnungen).

Da $\widehat{\Gamma}_k$ endlich ist, ist auch die Potenzmenge $\mathcal{P}(\widehat{\Gamma}_k)$ von $\widehat{\Gamma}_k$ endlich; und deren Potenzmenge $\mathfrak{P} = \mathcal{P}(\mathcal{P}(\widehat{\Gamma}_k))$ ist endlich.

Jedem Tableau T sei ein Element von $\mathfrak{P}$ wie folgt zugewiesen: Jeder Menge Φ von auf einem Ast von T auftretenden Formeln mit höchstens k Elementen sei das Paar $\langle \Phi, Assoc(T, \Phi) \rangle$ zugewiesen. Jedem Ast B von T sei ein Element von $\mathcal{P}(\widehat{\Gamma}_k)$ zugewiesen, daß Varianten der Paare enthält, die Mengen Φ von Formeln auf B mit höchstens k Elementen zugewiesen sind; dabei seien von Paaren, die bis auf Variablenumbenennung gleich sind, höchstens r ausgewählt. Dem Tableau T sei ein Element von $\mathfrak{P}$ zugewiesen, das aus Varianten der den Ästen von T zugewiesenen Mengen besteht.

Aus der Definition von $\subseteq_k$ bzw. $\sim_k$ folgt nun, daß zwei Tableaus T zur gleichen Äquivalenzklasse bzgl. $\sim_k$ gehören, wenn ihnen das gleiche Element von $\mathfrak{P}$ zugewiesen werden kann. Damit gibt es höchstens so viele Äquivalenzklassen, wie $\mathfrak{P}$ Elemente hat – also endlich viele.

Um den Beweis des Lemmas zu vervollständigen, nehmen wir an, daß die Zahl der Äquivalenzklassen von Tableaus bzgl. $\sim_k$ $n \in \mathbb{N}$ sei. Es kann keine unendliche, reguläre Folge von Tableaus geben, die ausschließlich Varianten der Formeln in Γ enthalten, weil es in einer Folge $T_1, \ldots, T_{n+1}$ der Länge $n + 1$ mindestens zwei verschiedene Tableaus T_j und T_i geben muß, die zur gleichen Äquivalenzklasse gehören, d. h., wir haben sowohl $T_j \subseteq_k T_i$ als auch $T_i \subseteq_k T_j$. Da entweder $j < i$ oder $i < j$ folgt, daß die Folge irregulär ist. $\square$

5.3 Gewichtsordnungen

Gewichtsordnungen sind das zweite wichtige Konzept (neben Regularität), auf dem unsere Fairneßstrategie basiert, mit der deterministische Beweisprozeduren für beliebige gutartige Kalkül mit starren Variablen konstruiert werden können.

Die wichtigen Eigenschaften einer Gewichtsordnung auf Tableauformeln, die benutzt wird, um Fairneß sicherzustellen, sind die folgenden:

1. Sie ist eine Wohlordnung auf der Menge der Tableauformeln (bis auf Umbenennung freier Variablen), d. h., sie ist wohlfundiert und es gibt immer nur endlich viele (von Variablenumbenennung abgesehen) verschiedene Tableauformeln, die mit einer gegebenen Tableauformel unvergleichbar sind.

2. Echte Instanzen einer Tableauformel ϕ haben ein höheres Gewicht als ϕ.

3. Tableauformeln, die bis auf Variablenumbenennung gleich sind, haben das gleiche Gewicht.

Intuitiv sind dies typische Eigenschaften einer Ordnung auf Tableauformeln, die dadurch definiert sind, daß den Symbolen der Signatur „Gewichte“ zugeordnet werden (was der Grund ist, warum wir sie *Gewichts*ordnungen nennen).

Definition 5.3.1 Sei $\mathcal{C}$ ein Kalkül mit starren Variablen; und sei Σ eine Signatur von $\mathcal{C}$.

Eine *Gewichtsfunktion* w weist jeder Tableauformel in $TabForm(\Sigma^*)$ eine natürliche Zahl (ihr *Gewicht*) zu, so daß die folgenden Bedingungen erfüllt sind:

1. Ist ϕ eine beliebige Tableauformel, dann gibt es *keine unendliche* Menge Ψ von Tableauformeln, so daß
   (a) Ψ keine Tableauformeln ψ, ψ' enthält, die bis auf die Umbenennung starrer Variablen gleich sind (d. h., alle Formeln in Ψ sind echt verschieden),
   (b) $w(\psi) \leq w(\phi)$ für alle $\psi \in \Psi$.
2. Kommt $X \in Var$ in $\phi \in TabForm(\Sigma^*)$ vor und ist $t \in Term_{\mathrm{fv}}(\Sigma^*)$, $t \notin Var$, ein Term, der keine Variable ist, dann gilt
$$w(\phi) < w(\phi\{X \mapsto t\}) \ .$$
3. Sind $\phi, \phi' \in TabForm(\Sigma^*)$ bis auf die Umbenennung starrer Variablen gleich, dann gilt $w(\phi) = w(\phi')$.

Sei w eine Gewichtsfunktion; dann ist die (von w induzierte) *Gewichtsordnung* $\leq_w$ auf Tableauformeln für alle $\phi, \psi \in TabForm(\Sigma^*)$ definiert durch:

$$\phi \leq_w \psi \text{ genau dann, wenn } w(\phi) \leq w(\psi) \ .$$

□

Eine Gewichtsordnung wird auf *Mengen* von Tableauformeln fortgesetzt, indem die *maximalen* Gewichte der Formeln, die sie enthalten, verglichen werden. Diese Fortsetzung ist ebenfalls eine Wohlordnung – vorausgesetzt die Mengen von Tableauformeln, die verglichen werden, dürfen nur eine beschränkte Zahl von Varianten jeder Tableauformel enthalten.

Beispiel 5.3.2 Sei Σ eine Signatur von PL1. Wir nehmen an, jedem Funktions- und jedem Prädikatensymbol in Σ^* sei ein (positives) Gewicht zugeordnet, so daß es nur endlich viele Funktions- und/oder Prädikatensymbole eines bestimmten Gewichtes gibt (starre Variablen haben implizit das Gewicht 0); dann sei $w(\phi)$ die Summe der Gewichte aller Funktions- und Prädikatensymbole, die in $\phi \in TabForm(\Sigma^*)$ vorkommen, wobei mehrfache Vorkommen mehrfach gezählt werden. Die Funktion w ist eine Gewichtsfunktion gemäß Definition 5.3.1.

Die Bedingung, daß nur endlich vielen Funktions- und Prädikatensymbolen das gleiche Gewicht zugeordnet werden darf, spielt in der Praxis keine Rolle, weil nur eine endliche Zahl verschiedener Symbole in den Tableaus für eine gegebene Menge von PL1-Formeln auftreten kann, wenn die verbesserte Skolemisierung aus Abschnitt 4.4 verwendet wird; allen tatsächlich vorkommenden Symbolen kann also das gleiche Gewicht zugeordnet werden. □

Der Zweck von Bedingung 1 in der Definition von Gewichtsfunktionen ist offenbar; sie stellt sicher, daß, wenn unendlich viele verschiedene Formeln zu einem Tableauast hinzugefügt werden, das maximale Gewicht der Formeln auf dem Ast früher oder später jeden beliebigen Wert erreichen wird. Bedingung 2 mag etwas mehr Erläuterung erfordern. Sie stellt sicher, daß das Gewicht zunimmt, falls keine neuen Formeln an einen Ast angehängt werden, sondern er nur immer und immer wieder dadurch verändert wird, daß eine Substitution angewendet wird. Man beachte, daß eine solche Folge von Regelanwendungen die Regularität nicht verletzt, wenn jeweils eine neue Formel entsteht.[4]

Beispiel 5.3.3 Wir nehmen an, daß eine Erweiterungsregel erlaubt, aus jeder Prämisse der Form $\{p(f(\cdots(X)\cdots))\}$ die Konklusion $\langle\{\{q\},\{r\}\},\{X \mapsto f(X')\}$ abzuleiten, wobei X' eine beliebige von X verschiedene starre Variable ist. Dann kann das in Abbildung 5.13 dargestellte (Teil-)Tableau für alle n aus dem (Teil-)Tableau abgeleitet werden, das aus einem einzigen mit $\mathsf{T}{:}p(X_1)$ markierten Knoten besteht.

Bedingung 2 in der Definition von Gewichtsfunktionen stellt sicher, daß das maximale Gewicht der Formeln in den Tableaus T_n zunimmt. □

5.4 Deterministische Beweisprozeduren für Kalküle mit starren Variablen

In diesem Abschnitt wird beschrieben, wie vollständige deterministische Beweisprozeduren für beliebige gutartige Kalküle mit starren Variablen definiert werden können; solche Beweisprozeduren können verwendet werden, um eine Tiefensuche nach Tableaubeweisen mit einer Fairneßstrategie auszuführen (siehe die Diskussion in Abschnitt 5.1). Sie werden mit Hilfe der Konzepte der Regularität (Abschnitt 5.2) und der Gewichtsordnungen (Abschnitt 5.3) konstruiert. Beispielsweise kann in dieser Weise eine deterministische Beweisprozedur für die starre bzw. gemischte Variablen verwendenden Kalküle für PL1 aus den Abschnitten 4.2.10 und 4.3.6 definiert werden.

[4] Macht man, wie im folgenden Abschnitt beschrieben, einen Kalkül durch zusätzliche Rekonstruktionsschritte schwach nicht-destruktiv, dann kann diese Situation nicht mehr eintreten und Bedingung 2 in der Definition von Gewichtsfunktionen ist nicht mehr unbedingt notwendig.

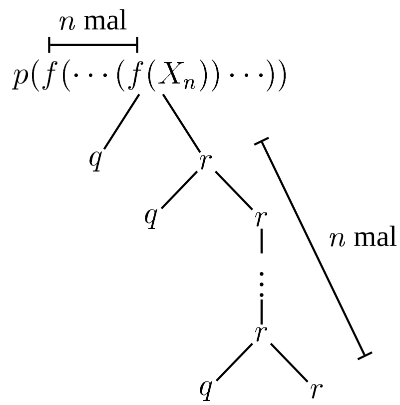


Abbildung 5.13: Ein Tableau, das die Notwendigkeit der Bedingung 2 in der Definition von Gewichtsfunktionen illustriert (siehe Beispiel 5.3.3).

Die Gutartigkeit des Kalküls ist unverzichtbar. Er muß nicht-strukturell sein, weil sonst die Reihenfolge, in der Formeln zu einem Tableau hinzugefügt werden, relevant ist und die Vollständigkeit verloren gehen kann, wenn sie in der Reihenfolge ihres Gewichtes hinzugefügt werden. Der Kalkül muß monoton sein, da deterministische Prozeduren unter Umständen zwischen nützlichen auch redundante Beweisschritte ausführen; und die redundanten Formeln, die dadurch hinzugefügt werden, dürfen die später nachzuholen nützlichen Erweiterungsregelanwendungen nicht verhindern. Es ist auch wichtig, daß der Kalkül monoton bzgl. Substitution ist, so daß die Reihenfolge in der verschiedene sinnvolle Substitutionen angewendet werden, irrelevant ist.

Wir betrachten zunächst nur solche Kalküle, in denen minimale Prämissen nicht beliebig groß sind, d. h., nicht mehr als k Formeln enthalten (für ein festes $k \in \mathbb{N}$). Wie Kalküle, die diese Eigenschaft nicht haben, dennoch behandelt werden können, wird am Ende dieses Abschnittes diskutiert.

Um sicherzustellen, daß eine Beweisprozedur in jedem Fall einen Tableaubeweis findet (vorausgesetzt ein Beweis existiert), müssen die Folgen von Tableaus, die konstruiert werden, die folgenden beiden Bedingungen erfüllen:

1. Die Folge von Tableaus, die konstruiert wird, muß k-regulär sein, d. h., Erweiterungsregelanwendungen, die ein in einem seiner Vorgänger k-enthaltenes Tableau erzeugen, sind verboten.

2. Bei jedem Ableitungsschritt wird aus allen möglichen Erweiterungsregelanwendungen, die die Regularitätsbedingung nicht verletzen (würden), eine solche ausgewählt, die ein Nachfolgetableau erzeugt, in dem das Maximalgewicht der Formeln so klein wie möglich ist (d. h., Nachfolgetableaus werden anhand des maximalen Gewichtes aller ihrer Formeln verglichen).

Wenn mehrere mögliche Erweiterungsregelanwendungen die obigen Bedingungen erfüllen, kann eine Beweisprozedur beliebige Heuristiken verwenden, um eine von ihnen auszuwählen. Eine typische Heuristik ist beispielsweise, Konklusionen zu bevorzugen, die wenige Extensionen enthalten, so daß wenige neue Teiläste erzeugt werden.

Man beachte, daß Formeln nicht notwendig in der Reihenfolge ihres Gewichtes zu einem Tableau hinzugefügt werden, weil eine Formel ϕ erst dann hinzugefügt werden kann, wenn ihre Prämisse Π auf dem Ast vorhanden ist, – und das Gewicht der Formeln in Π kann höher sein als das von ϕ. Auch wird der Zeitpunkt, zu dem eine Konklusion verwendet wird, von derjenigen ihrer Formeln bestimmt, die das höchste Gewicht hat, so daß Formeln mit geringem Gewicht, die nur zusammen mit Formeln hohen Gewichtes erzeugt werden könne, erst später zum Tableau hinzugefügt werden.

Um der Bedingung genüge zu tun, daß *alle* Erweiterungsregelanwendungen, die Formeln geringeren Gewichts hinzufügen, ausgeführt werden müssen, bevor Formeln größeren Gewichtes abgeleitet werden, kann es notwendig sein, schon geschlossene Äste zu erweitern. Solche Erweiterungen sind nicht immer redundant, denn auch geschlossene Äste enthalten noch immer nützliche Information und können andere Äste durch Substitutionen beeinflussen, die bei ihrer Erweiterung auf das Tableau angewendet werden (die erste Substitution, die angewendet wird, um einen Ast abzuschließen, ist nicht notwendigerweise die „richtige“). Wenn ein geschlossener Ast allerdings keine starren Variablen mit anderen Ästen gemeinsam hat, dann braucht er nicht erweitert zu werden.

Leider ist die Regularitäts-Bedingung, wie sie oben definiert ist, immer noch sehr schwierig zu implementieren; sie erfordert, ein Tableau T_{n+1} mit *allen* seinen Vorgängern $T_1, \ldots, T_n$ zu vergleichen, und nicht nur mit dem Tableau T_n, aus dem es abgeleitet wird. Solch ein Regularitätstest ist bezüglich des erforderlichen Speicherplatzes wie auch des Zeitaufwands sehr teuer. Noch schwerer wiegt, daß, wenn eine Irregularität entdeckt wird, wenn also T_{n+1} in einem seiner Vorgängertableaus T_j enthalten ist, andere Nachfolgetableaus von T_j (neben T_{j+1}) betrachtet werden müssen; das entspricht jedoch in gewissem Sinne Backtracking. Der Grund für die Notwendigkeit, andere Nachfolgetableaus von T_j zu betrachten, ist der folgende: Ein Tableau T_{n+1}, das in einem Tableau T_j enthalten ist, muß bei der Beweissuche nicht berücksichtigt werden, weil alle Tableaubeweise, die man aus T_{n+1} konstruieren kann, genauso aus T_j konstruiert werden können. Gilt $j = n$, dann genügt es, das Nachfolgetableau T_{n+1} auszuschließen; und man kann sicher sein, daß ein Beweis, der aus T_{n+1} ableitbar ist, genauso aus T_n ableitbar ist, ohne T_{n+1} zu betrachten. Gilt jedoch $j \neq n$, dann ist möglicherweise das Tableau T_n *nicht* Teil des aus T_{n+1} und also T_j ableitbaren Tableaubeweises, sondern erfordert, mit einem alternativen Nachfolgetableau T'_{j+1} fortzufahren, das von T_{j+1} verschieden ist. Die Situation ist in Abbildung 5.14 schematisch dargestellt.

Alle diese Probleme resultieren aus der Tatsache, daß ein Tableau T_j nicht notwendig in seinem Nachfolgetableau T_{j+1} enthalten ist, weil Kalküle mit starren Variablen destruktiv sind und Formeln, die in T_j vorkommen, in T_{j+1} möglicherweise nicht mehr enthalten sind. Ist der Kalkül jedoch in dem Sinne *schwach nicht-destruktiv*,

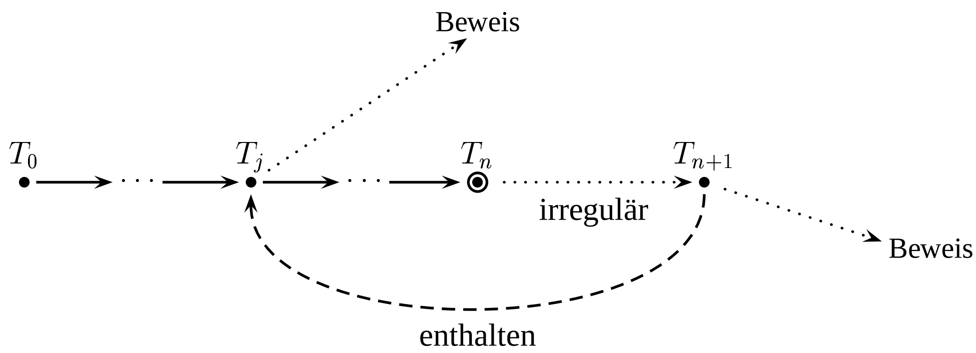


Abbildung 5.14: Beweissuche mit einem destruktiven Kalkül.

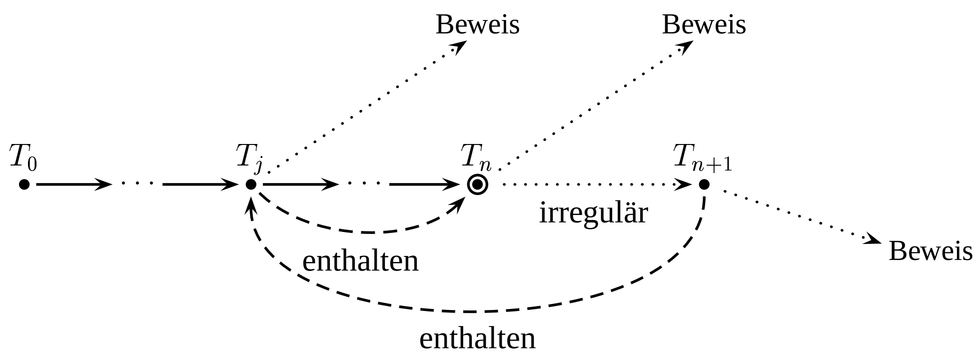


Abbildung 5.15: Beweissuche mit einem schwach nicht-destruktiven Kalkül.

daß Tableaus stets in ihren Nachfolgetableaus enthalten sind, ergibt sich die in Abbildung 5.15 dargestellte Situation. Nun ist das Tableau T_j in dem Tableau T_n enthalten, und man kann sicher sein, daß der aus T_{n+1} ableitbare Beweis auch direkt aus T_n ableitbar ist, also ohne den Weg über T_{n+1} zu nehmen. In gewisser Weise sind also (schwach) nicht-destruktive Kalküle beweiskonfluent bzgl. *regulärer* Beweise.

Hat ein Kalkül die Eigenschaft, daß Tableaus stets in ihren Nachfolgetableaus enthalten sind, nicht, dann können „unangenehme“ Zyklen auftreten. Es kann in einer unendlichen irreguläre Folge von Tableaus einen Zyklus geben, der aus unendlich vielen Tableaus T_{n_i} besteht, die sich alle gegenseitig enthalten; und es kann sein, daß keine der irregulären Erweiterungsregelanwendungen einfach zu erkennen ist, also kein Tableau in der Folge in seinem *unmittelbaren* Vorgänger enthalten ist.

Beispiel 5.4.1 Wir nehmen an, die in Tabelle 5.1 gezeigten symmetrische Regelschemata charakterisieren die Erweiterungsregel eines Kalküls mit starren Variablen.

Beginnend mit einem initialen Tableau, das aus einem Ast mit den Tableauformeln $\mathsf{T}{:}p(a)$, $\mathsf{T}{:}q(b)$ und $\mathsf{T}{:}p(X_1)$ besteht, kann eine Folge $T_1, T_2, \ldots$ von Tableaus abgeleitet werden, so daß jedes Tableau T_i ($i \geq 0$) der Folge das Tableau T_{i+2} 1-enthält; aber keines der Tableaus T_i ($i \geq 0$) enthält das Tableau T_{i+1} ($i \geq 0$). Die ersten vier Tableaus der Folge sind in Abbildung 5.16 dargestellt. $\square$

$\dfrac{\mathsf{T}{:}p(t)}{\mathsf{T}{:}q(Y)}$	$\dfrac{\mathsf{T}{:}q(t)}{\mathsf{T}{:}p(X)}$
falls t mit der Konstanten a unifizierbar ist; Y ist eine beliebig starre Variable; ein Unifikator von t und a ist anzuwenden	falls t mit der Konstanten b unifizierbar ist; X ist eine beliebig starre Variable; ein Unifikator von t und a ist anzuwenden

Tabelle 5.1: Symmetrische Regelschemata, die eine Zyklus erlauben.

$$
\begin{array}{cccc}
\mathsf{T}{:}p(a) & \mathsf{T}{:}p(a) & \mathsf{T}{:}p(a) & \mathsf{T}{:}p(a) \\
| & | & | & | \\
\mathsf{T}{:}q(b) & \mathsf{T}{:}q(b) & \mathsf{T}{:}q(b) & \mathsf{T}{:}q(b) \\
| & | & | & | \\
\mathsf{T}{:}p(X_1) & \mathsf{T}{:}p(a) & \mathsf{T}{:}p(a) & \mathsf{T}{:}p(a) \\
 & | & | & | \\
 & \mathsf{T}{:}q(Y_1) & \mathsf{T}{:}q(b) & \mathsf{T}{:}q(b) \\
 & & | & | \\
 & & \mathsf{T}{:}p(X_2) & \mathsf{T}{:}p(a) \\
 & & & | \\
 & & & \mathsf{T}{:}q(Y_2) \\
 & & & \\
T_1 & T_2 & T_3 & T_4
\end{array}
$$

Abbildung 5.16: Eine irreguläre Folge von Tableaus, die einen Zyklus enthält (Beispiel 5.4.1).

Das obige Beispiel illustriert den Grund dafür, daß schwer zu erkennende Zyklen auftreten können; es ist jedoch ein sehr künstliches Beispiel, und der Zyklus könnte vermieden werden, wenn immer eine solche Prämisse zur Ableitung einer bestimmten Konklusion verwendet würde, die die wenigsten Instantiierungen starrer Variablen notwendig macht (was ohnehin eine gute Heuristik ist). Das heißt, anstatt die Konklusion $\langle\{\{\mathsf{T}{:}p(Y)\}\},\{X \mapsto a\}\rangle$ aus der Prämisse $\{\mathsf{T}{:}p(X)\}$ abzuleiten, wird die Konklusion $\langle\{\{\mathsf{T}{:}p(Y)\}\}, id\rangle$ aus der Prämisse $\{\mathsf{T}{:}p(a)\}$ abgeleitet. Es gibt jedoch, wie das folgende Beispiel zeigt, kompliziertere Situationen, in denen auch die Verwendung anderer Prämissen nicht hilft.

Beispiel 5.4.2 Abbildungen 5.17 und 5.18 zeigen einen Zyklus in einer unendlichen irregulären Folge von Tableaus, die mit Hilfe des gemischte Variablen verwendenden Kalküls für PL1 aus Abschnitt 4.3.6 konstruiert ist.

Zur Konstruktion dieser Tableaufolge werden die folgenden Erweiterungsregelanwendungen wiederholt ausgeführt:

– Eine Konklusion der Form $\{\{\mathsf{T}{:}r(Y)\},\{\mathsf{T}{:}s(Y)\}\}$ wird aus $\{\mathsf{T}{:}(r(\boldsymbol{y}) \vee s(\boldsymbol{y}))\}$ abgeleitet und benutzt, um den ganz rechten Ast des Tableaus zu erweitern (z. B. um das Tableau T_6 aus dem Tableau T_5 abzuleiten).
– Der dritte Ast von rechts wird abgeschlossen; dabei wird eine Variable X mit a instantiiert (z. B. um T_7 aus T_6 abzuleiten).

– Eine Konklusion der Form $\{\{\mathsf{T}{:}p(X)\}, \{\mathsf{T}{:}q(X)\}\}$ wird aus $\{\mathsf{T}{:}(p(\boldsymbol{x}) \vee q(\boldsymbol{x}))\}$ abgeleitet und benutzt, um den ganz rechten Ast des Tableaus zu erweitern (z. B. um T_8 aus T_7 abzuleiten).
– Der dritte Ast von rechts wird abgeschlossen; dabei wird eine Variable Y mit b instantiiert (z. B. um T_9 aus T_8 abzuleiten).

Nach Ausführung dieser vier Schritte ist der Zyklus vollendet, d. h., jedes Tableau T_i der Folge enthält das Tableau T_{i+4} (für $i \geq 5$). Kein Tableau in der Folge enthält sein unmittelbares Nachfolgetableau, so daß der Zyklus schwierig zu erkennen ist. Man beachte, daß für jede Erweiterungsregelanwendung eine Prämisse verwendet wird, die die Instantiierung möglichst weniger Variablen erfordert. $\square$

Die oben beschriebenen Probleme, die alle dadurch entstehen, daß ein Tableau T_{i+1} nicht notwendigerweise sein Vorgängertableau T_i enthält, können gelöst werden, indem man die Beweisprozedur auf folgende Weise schwach nicht-destruktiv macht: Wir fordern, daß sofort nach einem destruktiven Erweiterungsschritt die Regelanwendungen ausgeführt werden, die notwendig sind, um die zerstörten Formeln zu rekonstruieren. Dabei wird im schlimmsten Fall eine Kopie des von den Instantiierungen betroffenen Teiltableaus an jeden betroffenen Teilast angehängt. Solch eine Rekonstruktion kann immer ausgeführt werden, weil der Kalkül gutartig ist. Das Ergebnis ist ein Tableau T^+_{i+1}, das das Tableau T_i und das Tableau T_{i+1} enhält, wie auch alle Tableaus, die als Zwischenergebnisse während der Rekonstruktion auftreten.

Beispiel 5.4.3 Der linke Ast des Tableaus T_i in Abbildung 5.19 (a) wird mit Hilfe der Konklusion $\{\{\bot\}\}, \{X \mapsto a\}$ abgeschlossen. Das Ergebnis ist das Tableau T_{i+1} in Abbildung 5.19 (b), in dem alle die starre Variable X enthaltenden Formeln zerstört worden sind. Sie werden rekonstruiert, indem Kopien des Teiltableaus $S(X)$ an alle Äste von T_{i+1} angehängt werden, auf denen Formeln fehlen; $S(X)$ besteht aus allen Formeln in T_i, in denen X vorkommt. Das Ergebnis ist das Tableau T^+_{i+1} (Abbildung 5.19 (c)), das sowohl T_i als auch T_{i+1} enthält. $\square$

Wenn eine deterministische Beweisprozedur nach jeder Erweiterungsregelanwendung zunächst eine Rekonstruktion ausführt, dann entsteht eine Folge $T^+_1, T^+_2, \ldots$ von Tableaus, wobei T^+_{i+1} aus T^+_i durch eine reguläre Erweiterungsregelanwendung und anschließende Rekonstruktion aller zerstörten Formeln entsteht. Um zu überprüfen, ob solch eine Folge regulär ist, genügt es, zu testen, ob das unmittelbare Nachfolgetableau T_{i+1} von T^+_i in T^+_i enthalten ist. Weitere Vorgängertableaus müssen nicht betrachtet werden, weil sie alle in T^+_i enthalten sind.

Satz 5.4.4 *Sei $\mathcal{C}$ ein gutartiger Kalkül mit starren Variablen für eine Logik $\mathbf{L}$; sei Σ eine Signatur von $\mathbf{L}$; sei $k \in \mathbb{N}$ die maximale Größe minimaler Prämissen in $\mathcal{C}$; und sei w eine Gewichtsfunktion.*

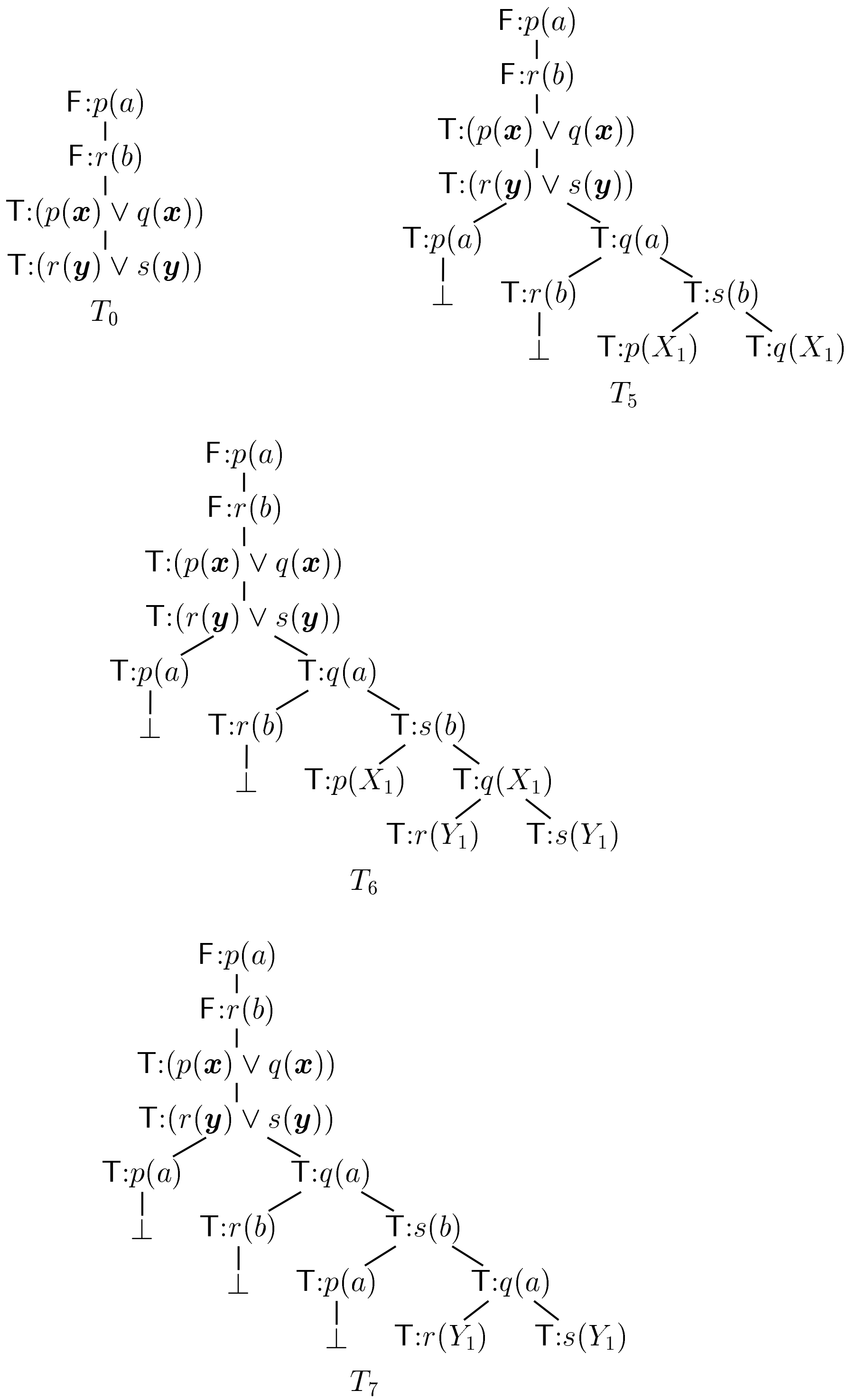
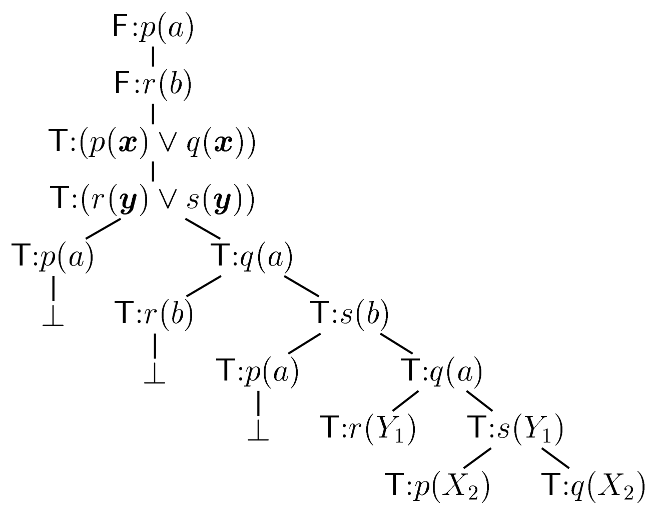
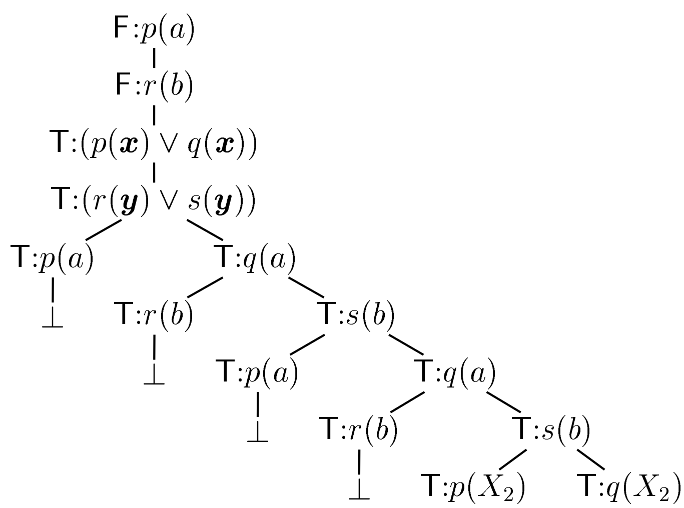


Abbildung 5.17: Eine irreguläre Folge von Tableaus, die einen Zyklus enthält (Teil 1); siehe Beispiel 5.4.2.



T_8



T_9

Abbildung 5.18: Eine irreguläre Folge von Tableaus, die einen Zyklus enthält (Teil 2); siehe Beispiel 5.4.2.

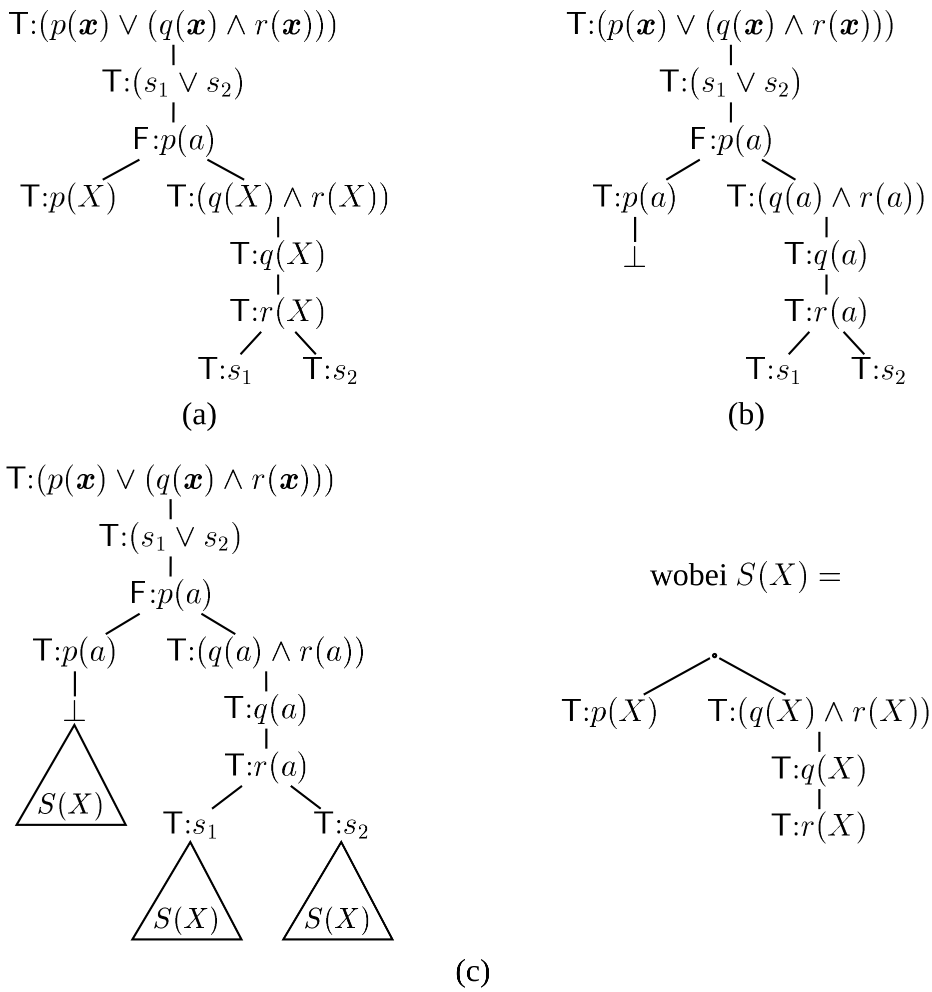


Abbildung 5.19: Eine Tableau-Rekonstruktion (Beispiel 5.4.3).

Falls es einen Tableaubeweis für eine Menge $\mathfrak{F} \subset Form(\Sigma^*)$ *gibt, dann enthält jede Folge* $(T_i^+)_{i\geq 1}$ *von Tableaus für* $\mathfrak{F}$*, die wie unten beschrieben konstruiert ist, ein geschlossenes Tableau* T_n^+ *(*$n \in \mathbb{N}$*):*

- T_1^+ *ist ein initiales Tableau für* $\mathfrak{F}$*.*
- *Für alle* $i > 1$ *gilt:*
  1. T_{i+1} *ist ein Nachfolgetableau von* T_i^+ *(Def. 5.2.11), so daß (a)* T_{i+1} *nicht in* T_i^+ k*-enthalten ist und (b) es kein Nachfolgetableau* T'_{i+1} *von* T_i^+ *gibt, das Bedingung (a) erfüllt und zudem ein kleineres maximales Formelgewicht als* T_{i+1} *hat (bzgl. der Gewichtsfunktion* w*).*
  2. *Sei* $\langle C_i, \tau_i \rangle$ *die aus einer Prämisse auf* T_i^+ *ableitbare Konklusion, die verwendet wird, um* T_{i+1} *zu konstruieren; und sei* S_i *das minimale Teiltableau von* T_{i+1}*, das alle Vorkommen der von* τ_i *instantiierten freien Variablen enthält. Das Tableau* T_{i+1}^+ *entsteht aus* T_{i+1} *durch (wiederholte) Ausführung aller Erweiterungsregelanwendungen, die notwendig sind, um das Teiltableau* S_i *zu erzeugen; dabei wird* S_i *an das Ende aller Äste angefügt, die durch das* S_i *entsprechende Teiltableau in* T_{i+1} *führen (welches durch Anwendung von* τ_i *auf* S_i *entsteht).*

Beweis: *Teil 1:* Da $\mathcal{C}$ gutartig ist, gilt folgendes: Wenn $\langle C, \sigma \rangle$ eine Konklusion einer Prämisse Π ist, dann ist $\langle C, id \rangle$ eine Konklusion von $\Pi\sigma$. Darum erfordern die Erweiterungsregelanwendungen, die notwendig sind, um T_i^+ aus T_i zu konstruieren, keine Instantiierung starrer Variablen, d. h., sie sind nicht-destruktiv. Daraus folgt, daß

$$T_i \subseteq_k T_i^+$$

für alle $i \geq 1$ gilt.

Per Konstruktion von T_{i+1}^+ werden alle Formeln, die durch die Anwendung der Substitution τ_i auf T_i^+ zerstört werden, auf allen Ästen auf denen sie in T_{i+1} fehlen, wieder eingeführt. Also gilt

$$T_i^+ \subseteq_k T_{i+1}^+ \quad .$$

Teil 2: Wir zeigen, daß die Folge $(T_i^+)_{i\geq 1}$ regulär ist (Def. 5.2.12). Nehmen wir das Gegenteil an; dann enthält die Folge Tableaus T_j^+, T_i^+, $j < i$, so daß $T_i^+ \subseteq_k T_j^+$. Mit den Resultaten aus Teil 1 impliziert dies

$$T_i \subseteq_k T_i^+ \subseteq_k T_j^+ \subseteq_k T_{i-1}^+ \quad ,$$

was der Tatsache widerspricht, daß (per Definition) das Tableau T_i *nicht* in T_{i-1}^+ k-enthalten ist.

Teil 3: Sei $w_{\max} \in \mathbb{N}$ ein beliebiges Gewicht. Wir zeigen, daß die initiale Teilfolge von $(T_i^+)_{i \geq 1}$ endlich sein muß, die nur Formeln ϕ mit Gewicht $w(\phi) \leq w_{\max}$ enthält.

Sei Γ eine Menge von (Varianten von) Repräsentanten jeder Klasse von Tableauformeln in $(T_i^+)_{i \geq 1}$, die bis auf Variablenumbenennung gleich sind und deren Gewicht nicht größer als $w_{\max}$ ist; ist also ϕ eine Tableauformel mit $w(\phi) \leq w_{\max}$ in $(T_i^+)_{i \geq 1}$, dann gibt es eine Variante von ϕ in Γ; wir wählen Γ so, daß die Formeln in Γ paarweise variablendisjunkt sind.

Die Definition von Gewichtsordnungen impliziert sofort, daß die Menge Γ endlich sein muß. Darum ist Lemma 5.2.16 anwendbar, und die initiale Teilfolge von $(T_i^+)_{i \geq 1}$, in der keine Formel mit einem Gewicht von mehr als $w_{\max}$ auftritt, ist endlich.

Teil 4: Per Voraussetzung gibt es einen (möglicherweise irregulären) Tableaubeweis $T_1', \ldots, T_m'$ für $\mathfrak{F}$. Sei $w_{\max}$ das maximale Gewicht aller in diesem Tableaubeweis auftretenden Tableauformeln; und sei T_n^+ das letzte Tableau in der (wie in Teil 3 gezeigt) endlichen initialen Teilfolge von $(T_i^+)_{i \geq 1}$, in der keine Tableauformeln mit einem Gewicht von mehr als $w_{\max}$ auftreten (das Tableau T_{n+1}^+ enthält also eine Formel mit einem Gewicht von mehr als $w_{\max}$).

Wir beweisen durch Induktion über j, daß es Substitutionen $\sigma_1, \ldots, \sigma_m \in \mathit{Subst}(\Sigma^*)$ gibt, so daß

$$T_j' \subseteq_k T_n^+ \sigma_j \quad .$$

$j = 1$: Sei $\sigma_1 = id$; die Tableaus T_1' und T_1^+ sind beide initiale Tableaus für $\mathfrak{F}$ und enthalten keine starren Variablen; also gilt trivialerweise $T_1' = T_1^+ \subseteq_k T_n^+ = T_n^+ \sigma_1$.

$j \to j+1$: Seien Π' die Prämisse und $\langle C, \rho \rangle$ die Konklusion, die für die Konstruktion von T_{j+1}' aus T_j' verwendet worden sind; und sei B_j' der Ast in T_j', der erweitert worden ist. Die Substitution ρ sei so gewählt, daß sie nur in Π' vorkommende Variablen instantiiert; das ist immer möglich, da der Kalkül gutartig ist.

Betrachten wir zunächst den Fall, daß kein Ast in $T_n^+ \sigma_j$ den Ast B_j' k-enthält. Dann enthält jeder Ast von $T_n^+ \sigma_j$ einen nicht-expandierten Ast von T_j'; und es gilt also $T_{j+1}' \subseteq_k T_n^+ \sigma_j \rho$ und daher $T_{j+1}' \subseteq_k T_n^+ \sigma_{j+1}$ mit $\sigma_{j+1} = (\rho \circ \sigma_j)$.

Andernfalls gibt es Äste $B^1, \ldots, B^s$ in $T_n^+ \sigma_j$, die B_j' k-enthalten. Da $B_j' \subseteq_k B^l$ für $1 \leq l \leq s$, gibt es eine Menge Π^l von Tableauformeln auf B^l und eine Substitution π (die wegen Bedingung 2 in der Definition der Enhaltenseins-Relation [Def. 5.2.3] nicht von l abhängt), so daß $\Pi^l \pi = \Pi'$. Also kann B mit der Konklusion $\langle C, \rho \circ \pi \rangle$ erweitert werden. Sei nun $\widehat{T}$ das Tableau, das dadurch aus $T_n^+ \sigma_j$ entsteht, daß jeder der Äste $B^1, \ldots, B^s$ mit der Konklusion $\langle C, \rho \circ \pi \rangle$ erweitert wird. Dieses Nachfolgetableau $\widehat{T}$ von $T_n^+ \sigma_j$ k-enthält T_{j+1}'.

Wir fahren fort, indem wir zeigen, daß jede Erweiterungsregelanwendung, die zur Konstruktion von $\widehat{T}$ aus $T_n^+ \sigma_j$ ausgeführt wird, *irregulär* ist. Alle durch eine solche Regelanwendung neu entstehenden Tableauformeln, nämlich die in C und die in

$(Assoc(T_n^+, \Pi))\pi\rho$, kommen nach der Definition von $\subseteq_k$ bis auf Variablenumbenennung in T'_{j+1} vor und haben also ein Gewicht kleiner oder gleich $w_{\max}$. Da aber per Konstruktion T_n^+ das letzte Tableau in der (regulären) Tableaufolge ist, das keine Formel mit einem Gewicht von mehr als $w_{\max}$ enthält, muß jede Erweiterungsregelanwendung, die keine Formel mit einem Gewicht von mehr als $w_{\max}$ erzeugt, irregulär sein. Damit gilt $T'_{j+1} \subseteq_k \widehat{T} \subseteq_k T_n^+\sigma_j = T_n^+\sigma_{j+1}$ mit $\sigma_{j+1} = \sigma_j$.

Teil 5: Wir können nun die Aussage des Satzes beweisen, denn insbesondere k-enthält das Tableau $T_n^+\sigma_m$ das Tableau T'_m. Da T'_m geschlossen ist, enthält jeder Ast von T'_m die Tableauformel $\perp$. Also ist $\perp$ auch in jedem Ast von $T_n^+\sigma_m$ und somit in jedem Ast von T_n^+ enthalten. Daraus folgt, daß T_n^+ geschlossen ist. $\square$

Eine Beweisprozedur, die die Bedingungen des Satzes 5.4.4 erfüllt und eine reguläre Folge $T_1^+, T_2^+, \ldots$ von Tableaus konstruiert, so daß $T_i^+ \subseteq_k T_{i+1}^+$ für alle $i \geq 1$, *simuliert* – in gewissem Sinne – eine Tiefensuche mit Iterative deepening (siehe Abschnitt 5.1.2). Das Gewicht der Formeln, die in einem Tableau vorkommen dürfen, wird schrittweise erhöht. Wenn ein (möglicherweise irregulärer) Tableaubeweis existiert, der keine Formeln mit einem Gewicht von mehr als $w_{\max}$ enthält, dann gibt es ein geschlossenes Tableau T_n^+, das das letzte in der konstruierten Folge ist, das keine Formel mit einem Gewicht von mehr als $w_{\max}$ enthält. Es enthält alle ableitbaren Tableaus $T_{<w_{\max}}$, die keine Formeln mit einem Gewicht von mehr als $w_{\max}$ enthalten. Der große Vorteil dieser simulierten TsID gegenüber klassischer TsID mit Backtracking ist, daß das Tableau T_n^+ eine sehr kompakte Darstellung des Suchraums ist. Alle Information, die in einem der Tableaus $T_{<w_{\max}}$ enthalten ist, ist in der einheitlichen Struktur T_n^+vorhanden; alle Tableaus im Suchraum, die zueinander symmetrisch sind und also die gleiche Information enthalten, sind durch ein einziges Teiltableau von T_n^+ dargestellt. Da kein Backtracking verwendet wird, geht schon abgeleitete Information niemals wieder verloren. Es kann Teile des Tableaus T_n^+ geben, die redundante Information darstellen und nutzlos sind (d. h., redundante Teiltableaus, die nicht hätten erzeugt werden sollen); diese sind aber nicht schädlich, da sie mit Hilfe der in Abschnitt 4.6 beschriebenen Pruning-Technik entfernt werden können.

Die in diesem Abschnitt beschriebene Methode zum Entwurf deterministischer Beweisprozeduren kann nur angewendet werden, wenn die Größe minimaler Prämissen durch eine Konstante $k \in \mathbb{N}$ beschränkt ist. Die Methode kann jedoch leicht so angepaßt werden, daß sie auch auf beliebige andere Kalküle anwendbar ist (wie beispielsweise den Kalkül für modale Logiken aus Abschnitt 3.7.4). Dazu wird die Schranke k während der Konstruktion einer regulären Folge $(T_i^+)_{i \geq 1}$ von Tableaus schrittweise erhöht. Man kann beispielsweise festlegen, daß $k = k_w(w_{\max})$ gelten soll, wobei k_w eine monoton wachsende Funktion ist und $w_{\max}$ das maximale Gewicht von Formeln in dem jeweils aktuellen Tableau.

5.5 Gutartigkeitserhaltende Suchraumeinschränkungen

Die im vorangegangenen Abschnitt beschriebene Methode zur Konstruktion einer deterministischen Beweisprozedur für Kalküle mit starren Variablen ist mit allen Suchraumeinschränkungen kompatibel, die die Gutartigkeit des Kalküls erhalten. Solche Einschränkungen basieren gewöhnlich auf semantischen Eigenschaften der jeweiligen Logik, so daß sie nur schwer in uniformer Weise formuliert werden können. Es folgenden wichtige Beispiele für solche Suchraumeinschränkungen.

Konnektivität Für viele Kalküle (und Logiken) kann eine *Konnektions*-Relation zwischen Tableauformeln definiert werden, so daß eine Formel ϕ, die keine Konnektion zu einer anderen Formel (oder Menge von Formeln) auf einem Ast B hat, mit Sicherheit redundant ist; in diesem Fall kann weder ϕ noch eine aus ϕ ableitbare Formel zum Abschluß von B beitragen. Das heißt, (1) es gibt keine minimale, ϕ enthaltenden Prämisse, aus der $\perp$ abgeleitet werden kann, und – iterativ – (2) keine aus einer minimalen, ϕ enthaltende Prämisse ableitbare Konklusion hat eine Konnektion zu einer Formel oder einer Menge von Formeln auf B. Um für die Einschränkung des Suchraums nützliche zu sein, muß die Konnektionsbedingung einfach zu überprüfen sein.

Für den PL1-Kalkül aus Abschnitt 3.6 kann beispielsweise eine Konnektions-Relation verwendet werden, bei der zwei Formeln genau dann eine Konnektion haben, wenn sie das gleiche Atom mit verschiedener Polarität enthalten (siehe z. B. (Beckert & Hähnle, 1998)).

Ein Konnektions-Begriff, der darauf beruht, das Vorkommen gleicher Atome mit verschiedener Polarität zu überprüfen, kann in den meisten Logiken in dem Sinne verwendet werden, daß das Vorkommen solcher Atome für eine Konnektion *ausreicht*. Oft müssen aber auch noch anderer Arten von Konnektionen berücksichtigt werden (beispielsweise haben Formeln, die Gleichungen repräsentieren, eine Konnektion zu allen Formeln, auf die diese Gleichung „angewendet“ werden kann).

Wird ein gutartiger Kalkül in einer Weise eingeschränkt, so daß eine minimale Prämissen nur zur Erweiterung eines Astes verwendet werden darf, wenn alle ihre Formeln eine Konnektion zu anderen Formeln auf dem Ast haben, dann bleibt die Gutartigkeit des Kalküls erhalten. Gutartigkeit und Beweiskonfluenz gehen jedoch verloren, wenn eine *starke* Konnektions-Bedingung verwendet wird, bei der eine minimale Prämisse nur verwendet werden darf, wenn eine ihrer Formeln eine Konnektion zum *Blatt* des Astes hat, der erweitert wird.

Selektionsfunktionen Ein weitere wichtige Methode zur Einschränkung des Suchraums, die die Gutartigkeit eines Kalküls bewahrt, ist die Verwendung einer Selektionsfunktion zur Auswahl der nächsten zu verwendenden Prämisse, die auf einer *Li-*

teralordnung beruht; diese Technik ist aus dem Beweisen in klassischer Klausellogik bekannt (Hähnle & Klingenbeck, 1996; Hähnle & Pape, 1997).

Man beachte, daß das Konzept der Literalordnungen von dem der Gewichtsordnungen (Abschnitt 5.3) sehr verschieden ist; so sind Literalordnungen (im allgemeinen) *keine* Wohlordnungen, da es unendlich viele unvergleichbare Formeln geben kann.

6 Fasern

6.1 Einführung

Die Methode des Faserns (*fibring*) ist ein sehr nützliches Konzept für die Kombination logischer Systeme basierend auf der Kombination ihrer Semantiken (Gabbay, 1996c; Gabbay, 1996a; Gabbay, 1999), siehe Abschnitt 6.2. Die grundlegende Idee ist, die Modelle, die die Semantiken zweier Logiken $\mathbf{L}_1$ und $\mathbf{L}_2$ definieren, so zu kombinieren, daß die resultierenden Strukturen benutzt werden können, um die Semantik von Ausdrücken (Formeln) aus der Kombination der Sprachen von $\mathbf{L}_1$ und $\mathbf{L}_2$ zu definieren. Die allgemeine Voraussetzung für Fasern ist, daß die Modelle Komponenten wie die Welten in Kripke-Modellen haben, was völlig mit den Annahmen, die wir über die Form von Modellen gemacht haben, übereinstimmt.

Um gefaserte Modelle zu bilden, werden Faser-Funktionen $\mathcal{F}_{(1,2)}$ definiert, die jeder Welt w eines $\mathbf{L}_1$-Modells $\mathbf{m}_1$ ein $\mathbf{L}_2$-Modell $\mathbf{m}_2$ zuweisen. Wenn eine $\mathbf{L}_2$-Formel in der Welt w ausgewertet werden soll, wo ihr Wahrheitswert zunächst undefiniert ist, dann wird sie statt dessen in $\mathbf{m}_2 = \mathcal{F}_{(1,2)}(w)$ ausgewertet. Die volle Stärke des Faserns kommt zum Vorschein, wenn der Prozeß iteriert wird, um eine Semantik für die Logik $\mathbf{L}_{[1,2]}$ zu definieren, in deren Formeln Operatoren der Komponentenlogiken beliebig verschachtelt auftreten können. Fasern ist in vielen Bereichen der Logik erfolgreich eingesetzt worden, um Systeme und ihre Semantiken zu kombinieren; für einen Überblick siehe (Gabbay, 1996c; Gabbay, 1999).

In diesem Kapitel erweitern wir das Konzept des Faserns auf *Tableaukalküle*. Wir beschreiben, wie in uniformer Weise ein korrekter und vollständiger Tableaukalkül für die kombinierte Logik aus Tableaukalkülen der Komponentenlogiken konstruiert werden kann. Da Tableaukalküle für die meisten „Basislogiken" schon zur Verfügung stehen (einschließlich klassischer Logiken, modaler Logiken, intuitionistischer Logiken, temporaler Logiken und vieler anderer mehr), können Kalküle für alle „komplexen" Logiken gewonnen werden, die sich durch Fasern aus Basislogiken ergeben, wie beispielsweise modale Prädikatenlogik, intuitionistische temporale Logik, usw.

Der wesentliche Vorteil eines uniformen Konzeptes für das Fasern von Tableaukalkülen ist, daß für die Konstruktion eines Kalküls für die Kombination $\mathbf{L}_{[1,2]}$ zweier Logiken kein Wissen über deren Interaktion benötigt wird. Also können Kalküle für $\mathbf{L}_{[1,2]}$ schnell und einfach gewonnen werden. Korrektheit und Vollständigkeit des gefaserten Kalküls müssen nicht neu bewiesen werden; sie folgen aus Korrektheitsund Vollständigkeitseigenschaften der Komponentenkalküle (Satz 6.4.3).

Es ist auch möglich, einen Kalkül $\mathcal{C}_1$ für eine Logik $\mathbf{L}_1$ mit einem Kalkül $\mathcal{C}_2$ für eine „Teillogik“ $\mathbf{L}_2$ von $\mathbf{L}_1$ zu fasern (beispielsweise ist Aussagenlogik eine Teillogik der Prädikatenlogik); wenn auch $\mathcal{C}_1$ die ganze Logik $\mathbf{L}_1$ behandeln kann, so mag doch der Kalkül $\mathcal{C}_2$ für Formeln aus $\mathbf{L}_2$ effizienter sein, so daß der gefaserte Kalkül $\mathcal{C}_{(1,2)}$ effizienter als $\mathcal{C}_1$ ist.

Es ist nicht möglich, völlig beliebige Tableaukalküle in uniformer Weise zu fasern. Wenn nichts über die Kalküle bekannt ist, dann ist nicht klar, wo und wie der Kalkül für $\mathbf{L}_2$ an den Kalkül für $\mathbf{L}_1$ „angeschraubt“ werden muß. Darum betrachten wir in diesem Kapitel nur Tableaukalküle, die – syntaktisch –

– gutartig sind

und – semantisch –

– die starken Korrektheitseigenschaften aus Definition 3.5.8 haben (Angemessenheit der Menge der Tableauinterpretationen und Korrektheit der Erweiterung),
– die starke Vollständigkeitseigenschaft 1 aus Definition 3.5.10 haben (Angemessenheit der Menge der Tableauinterpretationen) und
– stark semantisch analytisch sind (Def. 3.5.16).

Kalküle mit diesen Eigenschaften sind hinreichend „normal“, um gefasert werden zu können. Gutartigkeit der Komponentenkalküle ist ausreichend, um den gefaserten Kalkül in uniformer Weise zu definieren (d. h., seine Syntax und seine Semantik). Wenn die Komponentenkalküle darüber hinaus die oben aufgelisteten semantischen Eigenschaften haben, dann ist der entstehende gefaserte Kalkül automatisch korrekt und vollständig für die Logik, die durch Fasern der Komponentenlogiken entsteht. Außerdem hat dann der gefaserte Kalkül selbst diese semantischen Eigenschaften, was es ermöglicht, das Fasern von Tableaukalkülen zu iterieren und so einen Kalkül für die voll gefaserte Logik $\mathbf{L}_{[1,2]}$ zu konstruieren.

Als ein Beispiel werden in Abschnitt 6.5 ein Kalkül für PL1 und ein Kalkül für die Modallogik $\widehat{\mathrm{K}}$ gefasert, wodurch ein Kalkül für eine modale Prädikatenlogik entsteht.

Es kann sein, daß mit einem gefaserter Kalkül nur ein Semi-Entscheidungsverfahren für die Erfüllbarkeit von Formeln der gefaserten Logik gebildet werden kann, d. h., eine auf dem gefaserten Kalkül beruhende Beweisprozedur terminiert möglicherweise nur für unerfüllbare Eingabeformeln – selbst wenn die Komponentenkalküle benutzt werden können, um die Erfüllbarkeit in den Komponentenlogiken zu entscheiden. Das ist jedoch nicht überraschend, weil eine gefaserte Logik auch dann unentscheidbar sein kann, wenn ihre Komponentenlogiken entscheidbar sind.

Zu verwandten Arbeiten ist (D'Agostino & Gabbay, 1996) zu zählen, wo eine Methode zum Fasern von Tableaukalkülen für Logiken mit sub-struktureller Implikation vorgestellt wird. In (Gabbay & Governatori, 1998) ist eine Methode zum Fasern von Tableaukalkülen für modale Logiken beschrieben, um Kalküle für multi-modale Logiken zu erzeugen; sie kann als Instanz des allgemeinen Konzepts angesehen werden, daß im folgenden vorgestellt wird. In (Giunchiglia & Sebastiani, 1996) ist ein Deduktionssystem für modale Logiken beschrieben, das separate Prozeduren für die Handhabung des klassischen bzw. des modalen Anteils von Formeln verwendet.

In (Pfalzgraf, 1991; Pfalzgraf & Stokkermans, 1994) wird das sehr ähnliche Konzept der *logischen Fasern* eingeführt und motiviert; als konkrete Anwendungen wird die Modellierung der logischen Kontrolle kooperierender Agenten genannt. Baader und Schulz (1995a; 1995b) benutzen eine weitere Variante des Faserns, um die Definitionsbereiche symbolischer Constraints und Lösungsverfahren zu kombinieren.

Die Vorteile der Kombination von Logiken werden in (de Rijke, 1997) diskutiert.

6.2 Fasern logischer Systeme

In diesem Abschnitt ist die allgemeine Methode des Faserns logischer Systeme beschrieben, und Syntax und Semantik gefaserter Logiken werden – basierend auf Syntax und Semantik der Komponentenlogiken – definiert.

Intuitiv werden, wenn zwei Logiken $\mathbf{L}_1$ und $\mathbf{L}_2$ gefasert werden, Formeln betrachtet, die aus Symbolen und Operatoren beider Logiken aufgebaut sind (Gabbay, 1996c; Gabbay, 1999). In einem ersten Schritt betrachten wir eine Logik $\mathbf{L}_{(1,2)}$, bei der $\mathbf{L}_2$-Formeln innerhalb von $\mathbf{L}_1$-Formeln auftreten können aber nicht umgekehrt.

Die Logik $\mathbf{L}_{[1,2]} \equiv \mathbf{L}_{[2,1]}$, die die volle Kombination von $\mathbf{L}_1$ und $\mathbf{L}_2$ ist, wobei dann Ausdrücke aus beiden Logiken beliebig verschachtelt auftreten können, kann behandelt werden, indem die im folgenden beschriebene Konstruktion induktiv wiederholt wird. Genauso ist es möglich, drei oder mehr Logiken zu kombinieren.

Da wir logische Systeme in einer sehr allgemeinen Weise definiert haben, steht keine Information darüber zur Verfügung, wie Formeln aus Signaturen aufgebaut sind. Wir modellieren die Annahme, daß $\mathbf{L}_{(1,2)}$-Formeln konstruiert werden, indem $\mathbf{L}_2$-Formeln (auf irgendeine Art und Weise) als Teilformeln von $\mathbf{L}_1$-Formeln verwendet werden, wie folgt: Wir betrachten die Logik $\mathbf{L}_{(1,2)}$ als eine besondere Version von $\mathbf{L}_1$, die die Formeln von $\mathbf{L}_2$ als (zusätzliche) Atome enthält. Und in jeder Welt w der $\mathbf{L}_{(1,2)}$-Modelle (die angereicherte $\mathbf{L}_1$-Modelle sind) ist der Wahrheitswert der zusätzlichen Atome (die $\mathbf{L}_2$-Formeln sind) derselbe wie in der initialen Welt eines $\mathbf{L}_2$-Modells, das der Welt w zugewiesen ist. Also bestehen $\mathbf{L}_{(1,2)}$-Modelle aus einem $\mathbf{L}_1$-Modell $\mathbf{m}_1$ und einer *Faser-Funktion* $\mathcal{F}$, die jeder Welt w in $\mathbf{m}_1$ ein $\mathbf{L}_2$-Modell zuweist (dies ist in Abbildung 6.1 illustriert). Intuitiv wird eine $\mathbf{L}_2$-Formel, wenn sie in w ausgewertet werden soll, wo ihr Wahrheitswert undefiniert ist, statt dessen in $\mathbf{m}_2 = \mathcal{F}(w)$ ausgewertet.

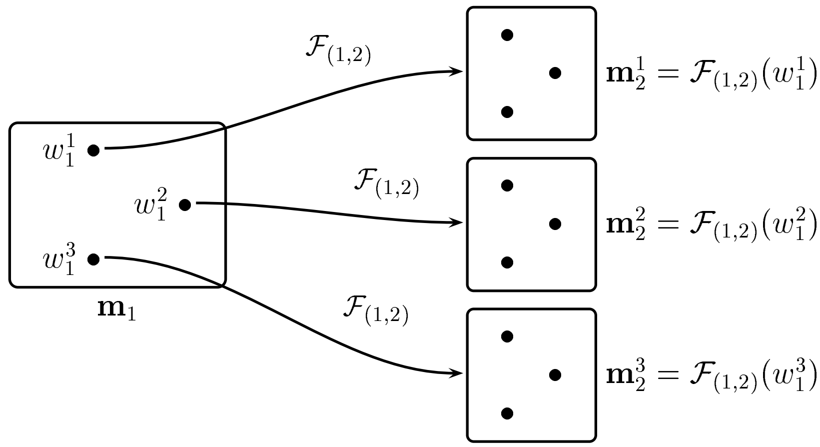


Abbildung 6.1: Ein gefasertes Modell.

Definition 6.2.1 Seien $\mathbf{L}_1$ und $\mathbf{L}_2$ Logiken; sei jedem Paar $\langle \Sigma_1, \Sigma_2 \rangle \in Sig_1 \times Sig_2$ von Signaturen eine Signatur $\Sigma_{(1,2)} \in Sig_1$ zugewiesen, so daß

$$Form_2(\Sigma_2) \subset Atom_1(\Sigma_{(1,2)}) \ .$$

Das *gefaserte logische System* $\mathbf{L}_{(1,2)}$, das durch Fasern der Logiken $\mathbf{L}_1$ und $\mathbf{L}_2$ entsteht, ist wie folgt definiert:

- Die Menge $Sig_{(1,2)}$ der Signaturen von $\mathbf{L}_{(1,2)}$ besteht aus allen Signaturen $\Sigma_{(1,2)}$ in Sig_1, die einem Paar $\langle \Sigma_1, \Sigma_2 \rangle \in Sig_1 \times Sig_2$ zugewiesen sind.

- Für alle Signaturen $\Sigma_{(1,2)} \in Sig_{(1,2)}$ ist die Menge $Form_{(1,2)}(\Sigma_{(1,2)})$ von $\mathbf{L}_{(1,2)}$-Formeln über $\Sigma_{(1,2)}$ identisch mit der Menge $Form_1(\Sigma_{(1,2)})$ von $\mathbf{L}_1$-Formeln über $\Sigma_{(1,2)}$, und die Menge $Atom_{(1,2)}(\Sigma_{(1,2)})$ von $\mathbf{L}_{(1,2)}$-Atomen über $\Sigma_{(1,2)}$ ist identisch mit der Menge $Atom_1(\Sigma_{(1,2)})$ von $\mathbf{L}_1$-Atomen über $\Sigma_{(1,2)}$.

- Für alle Signaturen $\Sigma_{(1,2)} \in Sig_{(1,2)}$ besteht die Menge $\mathcal{M}_{(1,2)}(\Sigma_{(1,2)})$ der $\mathbf{L}_{(1,2)}$-Modelle aus allen Paaren $\langle \mathbf{m}_1, \mathcal{F} \rangle$, wobei $\mathbf{m}_1 \in \mathcal{M}_1$ ein $\mathbf{L}_1$-Modell ist und $\mathcal{F}$ eine Faser-Funktion, die jeder Welt w in $\mathbf{m}_1$ ein $\mathbf{L}_2$-Modell $\mathbf{m}_2 \in \mathcal{M}_2(\Sigma_2)$ zuweist, so daß eine Formel $G \in Form_2(\Sigma_2)$ genau dann in w wahr ist, wenn sie in $\mathcal{F}(w)$ wahr ist, d. h.,

  $$w \models_1 G \ \text{ gdw. } \ \mathcal{F}(w) \models_2 G \ ;$$

  dabei bezeichnen $\models_1$ und $\models_2$ die Wahrheit ausdrückenden Relationen zwischen Welten und Formeln in $\mathbf{L}_1$ bzw. $\mathbf{L}_2$.

  Die Menge $W_{(1,2)}$ der Welten von $\langle \mathbf{m}_1, \mathcal{F} \rangle$ besteht aus allen Welten in $\mathbf{m}_1$ und allen Welten in den $\mathbf{L}_2$-Modellen, die Welten in $\mathbf{m}_1$ von $\mathcal{F}$ zugewiesen sind. Die initiale Welt von $\langle \mathbf{m}_1, \mathcal{F} \rangle$ ist die initiale Welt von $\mathbf{m}_1$.

Die die Wahrheit von Formeln in $\mathbf{L}_{(1,2)}$-Welten ausdrückende Relation $\models_{(1,2)}$ ist für alle Welten w und alle Formeln G in $\mathit{Form}_{(1,2)}(\Sigma_{(1,2)}) = \mathit{Form}_1(\Sigma_{(1,2)})$ definiert durch: $w \models_{(1,2)} G$ genau dann, wenn $w \models_1 G$. □

Beispiel 6.2.2 Um Prädikatenlogik erster Stufe PL1 und eine modale Logik fasern zu können, nehmen wir an, daß es für jede PL1-Signatur Σ_{PL1} eine Signatur Σ_{mod} in $\mathit{Sig}_{\mathrm{mod}}$ gibt, so daß die Atome über Σ_{PL1} die aussagenlogischen Variablen in Σ_{mod} sind. Dann ist $\Sigma_{(\mathrm{PL1},\mathrm{mod})}$ eine PL1-Signatur, in der die Prädikatensymbole von der Form $\circ_1 \cdots \circ_n\ p$ $(n \geq 0)$ sind, wobei $\circ_i \in \{\Box, \Diamond, -\}$ und p ein Prädikatensymbol in Σ_{PL1} ist.

Sind nun p und q Prädikatensymbole in Σ_{PL1} und ist a eine Konstante, dann ist $p(a)$ eine aussagenlogische Variable in Σ_{mod}, $\Box p$ ist ein Prädikatensymbol in $\Sigma_{(\mathrm{PL1},\mathrm{mod})}$ und $\Box p(a)$ ist sowohl ein Element von $\mathit{Atom}_{\mathrm{PL1}}(\Sigma_{(\mathrm{PL1},\mathrm{mod})})$ als auch von $\mathit{Form}_{\mathrm{mod}}(\Sigma_{\mathrm{mod}})$.

Beispiele für Formeln in $\mathit{Form}_{\mathrm{PL1}}(\Sigma_{(\mathrm{PL1},\mathrm{mod})})$ sind

$$\begin{array}{l} \Box p(a), \\ (\forall x)(p(x)), \\ (\forall x)(\Box p(x)), \\ (\forall x)(\Box p(x)) \to (\exists x)(\Diamond q(x)); \end{array}$$

jedoch ist $\Box(\forall x)(p(x))$ *keine* Formel in $\mathit{Form}_{\mathrm{PL1}}(\Sigma_{(\mathrm{PL1},\mathrm{mod})})$, da Modalitäten nur auf atomarer Ebene vorkommen dürfen; und $\Box p(x)$ ist weder eine Formel in $\mathbf{L}_{\mathrm{mod}}$ noch ein Atom in PL1, weil freie Objektvariablen gemäß Definition in PL1-Formeln nicht vorkommen dürfen.

Die gefaserte Logik $\mathbf{L}_{(\mathrm{PL1},\mathrm{mod})}$ ist eine modale Prädikatenlogik, wobei die modalen Operatoren nur auf atomarer Ebene vorkommen dürfen. Wenn jedoch der Faser-Prozeß iteriert wird, dann entsteht eine volle modale Prädikatenlogik, weil dann die logischen Operatoren $\vee$ und $\wedge$ der klassischen Prädikatenlogik innerhalb modallogischer Formeln verwendet werden können. □

6.3 Das Fasern von Tableaukalkülen

In diesem Abschnitt ist beschrieben, wie in uniformer Weise ein Tableaukalkül für eine gefaserte Logik $\mathbf{L}_{(1,2)}$ aus zwei Tableaukalkülen $\mathcal{C}_1$ und $\mathcal{C}_2$ für $\mathbf{L}_1$ bzw. $\mathbf{L}_2$ konstruiert werden kann.[1]

Wenn wir den Standpunkt einnehmen, daß die Suche nach einem Tableaubeweis ein Versuch ist, ein Modell für die Formeln des initialen Tableaus zu konstruieren, dann

[1] Wir betrachten hier nur Grundkalküle. Um einen Kalkül mit starren oder universellen Variablen für die gefaserte Logik $\mathbf{L}_{(1,2)}$ zu konstruieren, muß zunächst ein Grundkalkül $\mathcal{C}^{\mathrm{gd}}_{(1,2)}$ (durch Fasern) konstruiert werden, was dann erlaubt, einen Kalkül mit freien Variablen für $\mathbf{L}_{(1,2)}$ mit der Hebe-Technik zu konstruieren, wie sie in Kapitel 4 beschrieben ist.

entspricht die Suche nach einem Tableaubeweis in einem gefaserten Kalkül der Konstruktion eines gefaserten Modells. Die Tableauformeln des gefaserten Kalküls müssen daher Wissen über gefaserte Modelle repräsentieren können. Zu diesem Zweck benutzen wir Markierungen, die entweder von der Form $\sigma_1 \in Lab_1$ sind und Welten des $\mathbf{L}_1$-Modells bezeichnen oder von der Form $(\sigma_1; \sigma_2)$ sind (wobei $\sigma_1 \in Lab_1$ und $\sigma_2 \in Lab_2$) und Welten desjenigen $\mathbf{L}_2$-Modells bezeichnen, das durch die Faser-Funktion der von σ_1 bezeichneten Welt des $\mathbf{L}_1$-Modells zugewiesen wird. Eine Tableauformel $\mathsf{T}{:}\sigma_1{:}G$ drückt aus, daß G in $I_1(\sigma_1)$ wahr ist; die Bedeutung einer Tableauformel $\mathsf{T}{:}(\sigma_1; \sigma_2){:}G$ ist, daß G wahr ist in der Welt $I_2(\sigma_2)$ des Modells, das von der Faser-Funktion der Welt $I_1(\sigma_1)$ zugewiesen ist.

Der kombinierte Kalkül konstruiert keine getrennten Tableaus für $\mathbf{L}_1$- und $\mathbf{L}_2$-Formeln, sondern bildet eine einheitliche Struktur mit einer einheitlichen Tableauerweiterungsregel.

Die Tableauerweiterungsregel des gefaserten Kalküls $\mathcal{C}_{(1,2)}$, der aus $\mathcal{C}_1$ und $\mathcal{C}_2$ gebildet wird, hat drei Komponenten:

1. die Erweiterungsregel von $\mathcal{C}_1$; sie kann auf die $\mathbf{L}_1$-Tableauformeln eines Astes angewendet werden;

2. die Erweiterungsregel von $\mathcal{C}_2$; sie kann auf die $\mathbf{L}_2$-Tableauformeln mit einer Markierung der Form $(\sigma_1; \sigma_2)$ angewendet werden;

3. Ein Faser-Regelschema, das erlaubt, $\mathsf{S}{:}(\sigma_1; \sigma_2^0){:}G_2$ aus $\mathsf{S}{:}\sigma_1{:}G_2$ abzuleiten, wenn G_2 eine $\mathbf{L}_2$-Formel ist und also von der $\mathcal{C}_2$-Regel behandelt werden muß. Dieses Regelschema drückt aus, daß eine $\mathbf{L}_2$-Formel G_2, wenn sie in einer $\mathbf{L}_1$-Welt $w = I_1(\sigma_1)$ wahr ist, auch in der initialen Welt des w zugewiesenen $\mathbf{L}_2$-Modells wahr ist.

Die einzige zusätzliche Annahme – neben Gutartigkeit der Kalküle $\mathcal{C}_1$ und $\mathcal{C}_2$ für die Komponentenlogiken –, die wir machen müssen, um Syntax und Semantik des gefaserten Kalküls $\mathcal{C}_{(1,2)}$ definieren zu können, ist, daß die Erweiterungen der Signaturen, die von $\mathcal{C}_1$ und $\mathcal{C}_2$ benutzt werden, kompatibel sind.

Definition 6.3.1 Die Logik $\mathbf{L}_{(1,2)}$ entstehe durch Fasern der Logiken $\mathbf{L}_1$ und $\mathbf{L}_2$; und seien $\mathcal{C}_1$ und $\mathcal{C}_2$ Kalküle für $\mathbf{L}_1$ bzw. $\mathbf{L}_2$, so daß für alle Signaturen $\Sigma_1 \in Sig_1$ und $\Sigma_2 \in Sig_2$ eine Erweiterung $\Sigma_2^{(*)}$ von Σ_2 mit

– $Form_2(\Sigma_2^{(*)}) \subset Atom(\Sigma_{(1,2)}^*)$, und

– $\Sigma_2^* = (\Sigma_2^{(*)})^*$

existiert.

Dann sei der gefaserte Kalkül $\mathcal{C}_{(1,2)}$ für $\mathbf{L}_{(1,2)}$ für alle $\Sigma_1 \in Sig_1, \Sigma_2 \in Sig_2$ wie folgt definiert:

Markierungen: Die Menge der Markierungen von $\mathcal{C}_{(1,2)}$ ist

$$Lab_{(1,2)} = Lab_1 \cup \{(\sigma_1;\sigma_2) \mid \sigma_1 \in Lab_1, \sigma_2 \in Lab_2\} \ ;$$

die initiale Markierung $\sigma^0_{(1,2)}$ von $\mathcal{C}_{(1,2)}$ ist identisch mit der initialen Markierung σ^0_1 von $\mathcal{C}_1$.

Erweiterungsregel: Die Erweiterungsregel $\mathcal{E}_{(1,2)}$ von $\mathcal{C}_{(1,2)}$ weist Prämissen

$$\Pi \subset \mathit{TabForm}_{(1,2)}(\Sigma^*_{(1,2)})$$

die folgenden möglichen Konklusionen zu (wobei $\mathcal{E}_1$ und $\mathcal{E}_2$ die Erweiterungsregeln von $\mathcal{C}_1$ bzw. $\mathcal{C}_2$ sind):

1. die Konklusionen in $\mathcal{E}_1(\Pi_1)$, wobei Π_1 aus allen Tableauformeln ϕ in Π besteht, die von der Form $\phi = \mathsf{S}{:}\sigma_1{:}G$ sind *(Erweiterungsregel von $\mathcal{C}_1$),*
2. die Konklusionen für alle $\sigma_1 \in Lab_1$, die aus Konklusionen in $\mathcal{E}_2(\Pi_{2,\sigma_1})$ dadurch entstehen, daß man σ_2 durch $(\sigma_1;\sigma_2)$ ersetzt, wobei

$$\Pi_{2,\sigma_1} = \{\mathsf{S}{:}\sigma_2{:}G \mid \mathsf{S}{:}(\sigma_1;\sigma_2){:}G \in \Pi\}$$

   (Erweiterungsregel von $\mathcal{C}_2$),
3. die Konklusion $\{\{\mathsf{S}{:}(\sigma_1;\sigma^0_2){:}G\}\}$ für alle Tableauformeln ϕ in Π von der Form $\phi = \mathsf{S}{:}\sigma_1{:}G$ mit $G \in \mathit{Form}_2(\Sigma^*_2)$ *(Faser-Regelschema).*

Tableauinterpretationen: Die Menge $\mathit{TabInterp}_{(1,2)}$ der Tableauinterpretationen des Kalküls $\mathcal{C}_{(1,2)}$ besteht aus allen Paaren $\langle \mathbf{m}_{(1,2)}, I_{(1,2)}\rangle$, wobei $\mathbf{m}_{(1,2)} = \langle \mathbf{m}_1, \mathcal{F}\rangle$ ein $\mathbf{L}_{(1,2)}$-Modell ist, so daß gilt:

1. es gibt eine Tableauinterpretation $\langle \mathbf{m}_1, I_1\rangle$ in $\mathit{TabInterp}_1(\Sigma_{(1,2)})$;
2. für alle $\mathbf{L}_2$-Modelle $\mathbf{m}_{2,w} = \mathcal{F}(w)$, die einer Welt w in $\mathbf{m}_1$ zugewiesen sind, gibt es eine Tableauinterpretation $\langle \mathbf{m}_{2,w}, I_{2,w}\rangle$ in $\mathit{TabInterp}_2(\Sigma_2)$;
3. $I_{(1,2)}(\sigma_1) = I_1(\sigma_1)$ und $I_{(1,2)}((\sigma_1;\sigma_2)) = I_{2,I_1(\sigma_1)}(\sigma_2)$ für alle $\sigma_1 \in Lab_1$ und $\sigma_2 \in Lab_2$. □

6.4 Semantische Eigenschaften gefaserter Kalküle

Wie schon in Abschnitt 6.1 erwähnt, fordern wir, daß ein Kalkül, der zum Fasern verwendet wird, außer gutartig zu sein auch die Korrektheits- und Vollständigkeitseigenschaften aus Definitionen 3.5.8 und 3.5.10 hat und stark semantisch analytisch ist.

Definition 6.4.1 Ein Kalkül $\mathcal{C}$ ist *zum Fasern geeignet*, wenn er

– gutartig ist,
– die starken Korrektheitseigenschaften aus Definition 3.5.8 hat (Angemessenheit der Menge der Tableauinterpretationen und Korrektheit der Erweiterung),
– die starke Vollständigkeitseigenschaft 1 aus Definition 3.5.10 hat (Angemessenheit der Menge der Tableauinterpretationen) und
– stark semantisch analytisch ist (Def. 3.5.16). □

Die Gutartigkeit der Komponentenkalküle $\mathcal{C}_1$ und $\mathcal{C}_2$ reicht aus, um den gefaserten Kalkül $\mathcal{C}_{(1,2)}$ syntaktisch definieren zu können. Aber $\mathcal{C}_{(1,2)}$ ist nur dann *per Konstruktion* korrekt und vollständig (und selbst zum Fasern geeignet), wenn $\mathcal{C}_1$ und $\mathcal{C}_2$ die semantischen Eigenschaften haben, die sie (wie oben definiert) zum Fasern geeignet machen.

Es reicht zum Fasern nicht aus, wenn die Komponentenkalküle nur die schwachen Korrektheits- und Vollständigkeitseigenschaften haben (Def. 3.5.3 und 3.5.6). Bleibt beispielsweise nur die Erfüllbarkeit durch *irgendeine* Tableauinterpretation erhalten, wenn die Erweiterungsregel eines Komponentenkalküls angewendet wird, dann ist der gefaserte Kalkül möglicherweise nicht korrekt. Intuitiv ist der Grund dafür folgender: Nehmen wir an, $T_{(1,2)}$ sei ein Tableau für Formeln der gefaserten Logik $\mathbf{L}_{(1,2)}$, die $\mathbf{L}_1$-Tableauinterpretation $\langle \mathbf{m}_1, I_1 \rangle$ erfülle den $\mathbf{L}_1$-Anteil von $T_{(1,2)}$ und einer Welt w in $\mathbf{m}_1$ sei das $\mathbf{L}_2$-Modell $\mathbf{m}_2$ zugewiesen. Wenn nun die Erweiterungsregel des Komponentenkalküls für $\mathbf{L}_1$ nur die Erfüllbarkeit durch *irgendeine* Tableauinterpretation erhält, d. h., der $\mathbf{L}_1$-Anteil eines Nachfolgetableaus $T'_{(1,2)}$ von $T_{(1,2)}$ nur von irgendeiner von $\langle \mathbf{m}_1, I_1 \rangle$ verschiedenen Tableauinterpretation $\langle \mathbf{m}'_1, I'_1 \rangle$ erfüllt wird, dann würde ein Problem entstehen, falls $\mathbf{m}_2$ und die w entsprechende Welt w' in $\langle \mathbf{m}'_1, I'_1 \rangle$ nicht kompatibel sind, falls also eine $\mathbf{L}_2$-Formel, die in der initialen Welt von $\langle \mathbf{m}_1, I_1 \rangle$ (und in w) wahr ist, in w' *nicht* wahr ist.

Das folgende Beispiel zeigt, daß ein Kalkül tatsächlich semantisch analytisch sein muß, um zum Fasern geeignet zu sein, da sonst der gefaserte Kalkül unvollständig sein könnte.

Beispiel 6.4.2 Die Vollständigkeit eine Kalküls $\mathcal{C}_1$ für PL1 bleibt erhalten, wenn eine Formel der Form $\mathsf{T}{:}(p \vee q)$ dann nicht für die Erweiterung eines Astes verwendet wird, wenn das Atom q nirgends sonst in B auftritt (d. h., falls das Atom q *pur* ist). Ein Kalkül, der diese Suchraumbeschränkung benutzt, ist jedoch *nicht* semantisch analytisch.

Wenn $\mathcal{C}_1$ mit einem Kalkül $\mathcal{C}_2$ für eine modale Logik gefasert wird, dann kann ein PL1-Atom eine unerfüllbare modallogische Formel sein. Also kann es sein, daß eine pure Atome verwendende Erweiterung *nicht* redundant ist; und ein Kalkül für PL1, der zum Fasern geeignet sein soll, muß die Erweiterung eines Astes zulassen, der z. B. $\mathsf{T}{:}\sigma{:}[\Diamond(r \wedge \neg r) \vee q]$ enthält, so daß neue Teiläste entstehen, die $\mathsf{T}{:}\sigma{:}\Diamond(r \wedge \neg r)$ bzw.

$\mathsf{T}{:}\sigma{:}q$ enthalten, was es ermöglicht, die Formel $\mathsf{T}{:}\sigma{:}\Diamond(r \wedge \neg r)$ an die modale Komponente des gefaserten Kalküls weiterzureichen und so ihre Unerfüllbarkeit festzustellen. $\square$

Es folgt der Hauptsatz dieses Kapitels.

Satz 6.4.3 *Die Logik $\mathbf{L}_{(1,2)}$ entstehe durch Fasern der Logiken $\mathbf{L}_1$ und $\mathbf{L}_2$; $\mathcal{C}_1$ und $\mathcal{C}_2$ seien Kalküle für $\mathbf{L}_1$ bzw. $\mathbf{L}_2$, die zum Fasern geeignet sind (Def. 6.4.1), und der Kalkül $\mathcal{C}_{(1,2)}$ für $\mathbf{L}_{(1,2)}$ entstehe durch Fasern von $\mathcal{C}_1$ und $\mathcal{C}_2$ gemäß Definition 6.3.1.*

Dann ist der Kalkül $\mathcal{C}_{(1,2)}$ zum Fasern geeignet.

Beweis: *Gutartigkeit:* Per Konstruktion ist $\mathcal{C}_{(1,2)}$ ein Kalkül mit Erweiterungsregel, und es ist leicht zu überprüfen, daß seine Erweiterungsregel monoton und nicht-strukturell ist. Also ist $\mathcal{C}_{(1,2)}$ gutartig.

Starke Korrektheitseigenschaft 1: Die Menge $\mathfrak{F} \subset \mathit{Form}_{(1,2)}(\Sigma_{(1,2)})$ von Formeln werde durch ein Modell $\langle \mathbf{m}, \mathcal{F} \rangle \in \mathcal{M}_{(1,2)}(\Sigma_{(1,2)})$ erfüllt.

Da der Kalkül $\mathcal{C}_1$ die starke Korrektheitseigenschaft 1 hat, gibt es eine Tableauinterpretation $\langle \mathbf{m}_1^*, I_1 \rangle \in \mathit{TabForm}_1(\Sigma_{(1,2)}^*)$, die das initiale Tableau für $\mathfrak{F}$ erfüllt, so daß $\mathbf{m}_1^*$ eine Erweiterung von $\mathbf{m}_1$ ist. Nun sei für alle Welten w von $\mathbf{m}_1^*$ die Menge $\mathfrak{F}_w$ definiert als die Menge aller Formeln in $\mathit{Form}_2(\Sigma_2^*)$, die in w wahr sind; $\mathfrak{F}_w$ wird von dem Modell $\mathbf{m}_{2,w} = \mathcal{F}(w)$ erfüllt. Weil der Kalkül $\mathcal{C}_2$ die starke Korrektheitseigenschaft 1 hat, gibt es eine Tableauinterpretation $\langle \mathbf{m}_{2,w}^*, I_{2,w} \rangle \in \mathit{TabForm}_2(\Sigma_{(1,2)}^*)$, die das intiale Tableau für $\mathfrak{F}_w$ erfüllt, so daß $\mathbf{m}_{2,w}^*$ eine Erweiterung von $\mathbf{m}_{2,w}$ ist.

Die Tableauinterpretation $\langle\langle \mathbf{m}_1^*, \mathcal{F}^* \rangle, I_{(1,2)} \rangle \in \mathit{TabInterp}_{(1,2)}(\Sigma_{(1,2)}^*)$ mit

- $\mathcal{F}^*(w) = \mathbf{m}_{2,w}^*$,
- (a) $I_{(1,2)}(\sigma_1) = I_1(\sigma_1)$ und (b) $I_{(1,2)}((\sigma_1; \sigma_2)) = I_{2,I_1(\sigma_1)}(\sigma_2)$ für alle Paare
$$\langle \sigma_1, \sigma_2 \rangle \in \mathit{Lab}_1 \times \mathit{Lab}_2$$
  von Markierungen

erfüllt nun das initiale Tableau für $\mathfrak{F}$, und sie ist eine Erweiterung von $\langle \mathbf{m}, \mathcal{F} \rangle$.

Starke Korrektheitseigenschaft 2: Da der Kalkül $\mathcal{C}_{(1,2)}$ gutartig ist, können wir Lemma 3.5.9 benutzen. Sei $\Pi \subset \mathit{TabForm}_{(1,2)}(\Sigma_{(1,2)}^*)$ eine minimale Prämisse einer Konklusion C; und nehmen wir an, daß Π von einer Tableauinterpretation
$$\langle\langle \mathbf{m}_1, \mathcal{F} \rangle, I_{(1,2)} \rangle \in \mathit{TabInterp}_{(1,2)}(\Sigma_{(1,2)}^*)$$
erfüllt werde. Wir zeigen, daß $\langle\langle \mathbf{m}_1, \mathcal{F} \rangle, I_{(1,2)} \rangle$ eine der Extensionen in C erfüllt; wir betrachten die folgenden Fälle in Abhängigkeit von der Form von Π:

Wenn Π nur Formeln der Form $\mathsf{S}{:}\sigma_1{:}G$ enthält, C also aus Π mit der Erweiterungsregel von $\mathcal{C}_1$ abgeleitet wird, dann wird eine der Extensionen in C von der Tableauinterpretation $\langle\langle \mathbf{m}_1, \mathcal{F}\rangle, I_{(1,2)}\rangle$ erfüllt, weil $\mathcal{C}_1$ die starke Korrektheitseigenschaft 2 hat.

Wenn Π nur Formeln der Form $\mathsf{S}{:}(\sigma_1;\sigma_2){:}G$ enthält, C also aus Π mit der Erweiterungsregel von $\mathcal{C}_2$ abgeleitet wird, dann wird eine der Extensionen in C von der Tableauinterpretation $\langle\langle \mathbf{m}_1, \mathcal{F}\rangle, I_{(1,2)}\rangle$ erfüllt, weil $\mathcal{C}_2$ die starke Korrektheitseigenschaft 2 hat.

Wenn $\Pi = \{\mathsf{S}{:}\sigma_1{:}G\}$ mit $G \in \mathit{Form}_2(\Sigma_2^*)$ und $C = \{\{\mathsf{S}{:}(\sigma_1;\sigma_2^0){:}G\}\}$, dann ist der Wahrheitswert von G in $w_1 = I_{(1,2)}(\sigma_1)$ der gleiche wie in der Welt $I_{(1,2)}(\sigma_1;\sigma_2^0)$, die die initiale Welt von $\mathcal{F}(w_1)$ ist; daher erfüllt die Tableauinterpretation $\langle\langle \mathbf{m}_1, \mathcal{F}\rangle, I_{(1,2)}\rangle$ die Formel $\mathsf{S}{:}(\sigma_1;\sigma_2^0){:}G$.

Starke Vollständigkeitseigenschaft 1: Sei $\langle\langle \mathbf{m}_1^*, \mathcal{F}^*\rangle, I_{(1,2)}\rangle$ eine Tableauinterpretation, die das initiale Tableau T_1 für eine Menge $\mathfrak{F} \subset \mathit{Form}_{(1,2)}(\Sigma_{(1,2)})$ von Formeln erfüllt.

Die $\mathcal{C}_1$-Tableauinterpretation $\langle \mathbf{m}_1^*, I_1\rangle$, wobei I_1 die Einschränkung von $I_{(1,2)}$ auf Markierungen in Lab_1 ist, erfüllt T_1 ebenfalls. Da der Kalkül $\mathcal{C}_1$ die starke Vollständigkeitseigenschaft 1 hat, gibt es eine Modell $\mathbf{m}_1 \in \mathcal{M}_1(\Sigma_{(1,2)})$, das eine Einschränkung von $\mathbf{m}_1^*$ ist und $\mathfrak{F}$ erfüllt.

Nun sei für alle Welten w von $\mathbf{m}_1^*$ die Menge $\mathfrak{F}_w$ definiert als die Menge aller Tableauformeln $\sigma_2^0{:}\mathsf{T}{:}G$, so daß $G \in \mathit{Form}_2(\Sigma_{(1,2)}^*)$ in w wahr ist; und sei σ_1 eine Markierung in Lab_1, für die $I_1(\sigma_1) = w$ gilt. Die $\mathcal{C}_2$-Tableauinterpretation $\langle \mathcal{F}^*(w), I_2\rangle$ mit $I_2(\sigma_2) = I_{(1,2)}((\sigma_1;\sigma_2))$ erfüllt $\mathfrak{F}_w$. Da der Kalkül $\mathcal{C}_2$ die starke Vollständigkeitseigenschaft 1 hat, gibt es ein Modell $\mathbf{m}_{2,w}$ in $\mathcal{M}_2(\Sigma_2)$, das eine Einschränkung von $\mathcal{F}^*(w)$ ist und $\mathfrak{F}_w$ erfüllt. Per Konstruktion erfüllt das Modell $\mathbf{m}_{2,w}$ alle Formeln G, die in w wahr sind.

Damit erfüllt das $\mathbf{L}_{(1,2)}$-Modell $\langle \mathbf{m}_1, \mathcal{F}\rangle$ von $\mathbf{L}_{(1,2)}$ die Formelmenge $\mathfrak{F}$ und ist eine Einschränkung von $\langle\langle \mathbf{m}_1^*, \mathcal{F}^*\rangle$, wobei $\mathcal{F}(w) = \mathbf{m}_{2,f(w)}$ für alle Welten w in $\mathbf{m}_1$ (dabei ist $f(w)$ die w entsprechende Welt in $\mathbf{m}_1^*$).

Der Kalkül ist stark semantisch analytisch: Sei B ein voll expandierter Ast, der nicht geschlossen ist; und sei $\Phi_{(1,2)} \subset \mathit{TabForm}_{(1,2)}(\Sigma^*)$ eine Menge atomarer Tableauformeln, so daß für *kein* ϕ in $\Phi_{(1,2)}$ sowohl ϕ als auch $\overline{\phi}$ Element von $\mathit{Form}(B) \cup \Phi_{(1,2)}$ sind. Wir zeigen, daß es eine Tableauinterpretation in $\mathit{TabInterp}_{(1,2)}(\Sigma_{(1,2)})$ gibt, die B und $\Phi_{(1,2)}$ erfüllt.

Für alle $\sigma_1 \in \mathit{Lab}_1$ sei B_{2,σ_1} der Teilast von B, der aus Formeln der Form $\mathsf{S}{:}(\sigma_1;\sigma_2){:}G$ besteht. Da B_{2,σ_1} ein voll expandierter, nicht-geschlossener $\mathcal{C}_2$-Ast ist und zudem der Kalkül $\mathcal{C}_2$ stark semantisch analytisch ist, gibt es eine Tableauinterpretation

$$\langle \mathbf{m}_{2,\sigma_1}, I_{2,\sigma_1}\rangle \in \mathit{TabInterp}_{(1,2)}(\Sigma_{(1,2)}) \ ,$$

die sowohl B_{2,σ_1} als auch die Menge

$$\{\mathsf{S}{:}\sigma_2{:}G \mid \mathsf{S}{:}(\sigma_1;\sigma_2){:}G \in \Phi_{(1,2)}\}$$

atomarer Formeln erfüllt.

Sei B_1 der Teilast von B, der aus Formeln der Form $S{:}\sigma_1{:}G$ besteht, wobei $\sigma_1 \in \mathit{Lab}_1$. Da B_1 ein voll expandierter, nicht-geschlossener $\mathcal{C}_1$-Ast ist und $\mathcal{C}_1$ stark semantisch analytisch ist, gibt es eine Tableauinterpretation $\langle \mathbf{m}_1, I_1 \rangle \in \mathit{TabInterp}_1(\Sigma_{(1,2)})$, die sowohl B_1 als auch die atomaren Formeln der Form $\mathsf{S}{:}\sigma_1{:}G$ in $\Phi_{(1,2)}$ erfüllt.

Sei die Menge $\Phi_1 \subset \mathit{TabForm}_1(\Sigma_{(1,2)})$ atomarer Tableauformeln wie folgt definiert:

$$\Phi_1 = \{\mathsf{S}{:}\sigma_1{:}G \mid G \in \mathit{Form}_2(\Sigma_2^*), \text{ und } \langle \mathbf{m}_{2,\sigma_1}, I_{2,\sigma_1} \rangle \text{ erfüllt } \mathsf{S}{:}\sigma_2{:}G\} \ .$$

Per Konstruktion kann die Menge Φ_1 kein Atom ϕ und sein Komplement $\overline{\phi}$ enthalten; und das Komplement von $\phi = \mathsf{S}{:}\sigma_1{:}G \in \Phi$ kann nicht in $\mathit{Form}(B_1)$ vorkommen, weil sonst B_{2,σ_1} auch das Komplement von $\mathsf{S}{:}\sigma_2{:}G$ enthalten würde (da B voll expandiert ist), das ist jedoch nicht möglich, weil $\langle \mathbf{m}_{2,\sigma_1}, I_{2,\sigma_1} \rangle$ sowohl $\mathsf{S}{:}\sigma_2{:}G$ als auch B_{2,σ_1} erfüllt.

Damit gibt es, da der Kalkül $\mathcal{C}_1$ stark semantisch analytisch ist, eine Tableauinterpretation $\langle \mathbf{m}_1', I_1' \rangle \in \mathit{TabInterp}_1(\Sigma^*_{(1,2)})$, die sowohl B_1 als auch Φ_1 erfüllt.

Schließlich folgt nun, daß die Tableauinterpretation

$$\langle \langle \mathbf{m}_1', \mathcal{F} \rangle, I_{(1,2)} \rangle \in \mathit{TabInterp}_{(1,2)}(\Sigma^*_{(1,2)})$$

den Ast B erfüllt, die definiert ist durch:

– Für alle Welten w' in $\mathbf{m}_1'$ ist das $\mathbf{L}_2$-Modell, das w' zugewiesen wird, $\mathcal{F}(w') = \mathbf{m}_{2,\sigma_1}$ (dabei ist σ_1 eine beliebige Markierung in Lab_1, so daß $I_1'(\sigma_1) = w'$),
– (a) $I_{(1,2)}(\sigma_1) = I_1'(\sigma_1)$ und (b) $I_{(1,2)}((\sigma_1; \sigma_2)) = I_{2,\sigma_1}(\sigma_2)$ für alle Paare

$$\langle \sigma_1, \sigma_2 \rangle \in \mathit{Lab}_1 \times \mathit{Lab}_2$$

von Markierungen.

□

Ein Kalkül, der zum Fasern geeignet ist, hat per Definition Eigenschaften, die seine Korrektheit und Vollständigkeit zusichern (Sätze 3.5.4 und 3.5.7). Darum impliziert Satz 6.4.3 das folgende Korollar.

Korollar 6.4.4 *Der Kalkül $\mathcal{C}_{(1,2)}$ ist korrekt und vollständig.*

6.5 Beispiel: Fasern von Kalkülen für PL1 und einer modalen Logik

Als ein Beispiel fasern wir den in Abschnitt 3.6 definierten Kalkül $\mathcal{C}_{\mathrm{PL1}}$ für Prädikatenlogik erster Stufe PL1 und die Einschränkung $\mathcal{C}_{\widehat{\mathrm{K}}}$ des in Abschnitt 3.7 definierten

Kalküls für die modale Logik $\widehat{\mathrm{K}}$ ohne binäre logische Operatoren aus Abschnitt 2.5.[2] Das Resultat ist ein Kalkül $\mathcal{C}_{(1,2)}$ für eine modale Prädikatenlogik, wobei die modallogischen Operatoren nur auf atomarer Ebene vorkommen dürfen (Beispiel 6.2.2).

Die Tableauerweiterungsregel des gefaserten Kalküls kann leicht durch Instantiierung der Kalküle $\mathcal{C}_1$ und $\mathcal{C}_2$ in Definition 6.3.1 mit $\mathcal{C}_{\mathrm{PL1}}$ bzw. $\mathcal{C}_{\widehat{\mathrm{K}}}$ gewonnen werden. Als Beispiel präsentieren wir einen Tableaubeweis für die Gültigkeit der Formel

$$G \;=\; (\forall x)(\Box p(x)) \to [\neg(\exists y)(\Diamond{-}p(y)) \land \neg(\exists z)(\Diamond{-}p(z))]$$

in der Logik $\mathbf{L}_{(1,2)} = \mathbf{L}_{(\mathrm{PL1},\widehat{\mathrm{K}})}$; wir benutzen den Kalkül $\mathcal{C}_{(1,2)} = \mathcal{C}_{(\mathrm{PL1},\widehat{\mathrm{K}})}$, um ein geschlossenes Tableau für $\neg G$ zu konstruieren. Das Tableau ist in Abbildung 6.2 gezeigt; es wird wie folgt erzeugt: Die Tableauformel 1 ist im initialen Tableau enthalten; dann werden die Formeln 2–7 durch Anwendung der Regelschemata von $\mathcal{C}_{\mathrm{PL1}}$ für α- und β-Formeln hinzugefügt. Das Schema von $\mathcal{C}_{\mathrm{PL1}}$ für δ-Formeln wird angewendet, um 8 aus 7 abzuleiten, wobei die Skolem-Konstante $c_1 = sko((\exists y)(\Diamond{-}p(y)))$ eingeführt wird. Nun wird, da 8 eine $\widehat{\mathrm{K}}$-Formel enthält, das Faser-Regelschema angewendet, um 9 zum Ast hinzuzufügen, was es dann erlaubt, die $\mathcal{C}_{\widehat{\mathrm{K}}}$-Erweiterungsregel anzuwenden, um 10 aus 9 abzuleiten (wir nehmen an, daß $\lceil \Diamond{-}p(c_1) \rceil = 1$) und um 11 aus 10 abzuleiten. An dieser Stelle wird das Regelschema von $\mathcal{C}_{\mathrm{PL1}}$ für γ-Formeln auf 3 angewendet, um 12 abzuleiten, wobei die universell quantifizierte Objektvariable x durch den Grundterm c_1 ersetzt wird (was zeigt, daß $\mathcal{C}_1$- und $\mathcal{C}_2$-Regelschemata in beliebiger Reihenfolge angewendet werden können). Schließlich wird das Faser-Regelschema auf 12 angewendet, um 13 abzuleiten, und das Regelschema von $\mathcal{C}_{\widehat{\mathrm{K}}}$ für ν-Formeln wird angewendet, um 14 abzuleiten. Nun kann der linke Ast des Tableaus abgeschlossen werden, und zwar mit Hilfe des Erweiterungsregel-Schemas für Astabschluß des Kalküls $\mathcal{C}_{\widehat{\mathrm{K}}}$; denn er enthält die komplementären Atome 11 und 14. Der rechte Ast ist in genau der gleichen Weise erweitert und geschlossen.

Seine volle Kraft entfaltet das Fasern, wenn der Faser-Prozeß iteriert wird, um aus $\mathcal{C}_{\mathrm{PL1}}$ und $\mathcal{C}_{\widehat{\mathrm{K}}}$ einen Kalkül $\mathcal{C}_{[\mathrm{PL1},\widehat{\mathrm{K}}]}$ für die volle modale Prädikatenlogik $\mathbf{L}_{[\mathrm{PL1},\widehat{\mathrm{K}}]}$ zu konstruieren; das ist möglich, weil die Kalküle $\mathcal{C}_{(1,2)}, \mathcal{C}_{(1,(2,1))}, \ldots$ alle zum Fasern geeignet sind. Als Beispiel benutzen wir $\mathcal{C}_{[\mathrm{PL1},\widehat{\mathrm{K}}]}$, um die Gültigkeit der Formel $G' =$

$$(\forall x)(\Box(r(x) \land s(x))) \to [\neg(\exists y)(\Diamond({-}r(y) \lor {-}s(y))) \land \neg(\exists z)(\Diamond({-}r(z) \lor {-}s(z)))]$$

in $\mathbf{L}_{[\mathrm{PL1},\widehat{\mathrm{K}}]}$ zu beweisen; die Formel G' entsteht aus G wie folgt: das Atom $p(x)$ wird durch $r(x) \land s(x)$ ersetzt, und ${-}p(y)$ und ${-}p(z)$ werden durch ${-}r(y) \lor {-}s(y)$ bzw. ${-}r(z) \lor {-}s(z)$ ersetzt. Die Konstruktion des Tableaus beginnt wie oben für G (Abbildung 6.2). Wir betrachten nur den linken Ast (der rechte Ast kann in der gleichen Weise geschlossen werden). Anstelle der Atome 10 und 14 enthält der Ast nun $10' = \mathsf{T}{:}(*;1.1){:}({-}r(c_1) \lor {-}s(c_1))$ und $14' = \mathsf{T}{:}(*;1.1){:}(r(c_1) \land s(c_1))$. Die Erweiterung des Astes erfolgt wie in Abbildung 6.3 (um die Notation zu vereinfachen,

[2] Um Negationen in PL1 und in $\widehat{\mathrm{K}}$ zu unterscheiden, wird in modallogischen Formeln das Symbol $-$ anstelle von $\neg$ verwendet.

1 $\mathsf{T}{:}*{:}\neg((\forall x)(\Box p(x)) \to [\neg(\exists y)(\Diamond{-}p(y)) \wedge \neg(\exists z)(\Diamond{-}p(z))])$

2 $\mathsf{F}{:}*{:}(\forall x)(\Box p(x)) \to [\neg(\exists y)(\Diamond{-}p(y)) \wedge \neg(\exists z)(\Diamond{-}p(z))]$

3 $\mathsf{T}{:}*{:}(\forall x)(\Box p(x))$

4 $\mathsf{F}{:}*{:}\neg(\exists y)(\Diamond{-}p(y)) \wedge \neg(\exists z)(\Diamond{-}p(z))$

5 $\mathsf{F}{:}*{:}\neg(\exists y)(\Diamond{-}p(y))$	**6** $\mathsf{F}{:}*{:}\neg(\exists y)(\Diamond{-}p(y))$
7 $\mathsf{T}{:}*{:}(\exists y)(\Diamond{-}p(y))$	**15** $\mathsf{T}{:}*{:}(\exists y)(\Diamond{-}p(y))$
8 $\mathsf{T}{:}*{:}\Diamond{-}p(c_1)$	**16** $\mathsf{T}{:}*{:}\Diamond{-}p(c_1)$
9 $\mathsf{T}{:}(*;1){:}\Diamond{-}p(c_1)$	**17** $\mathsf{T}{:}(*;1){:}\Diamond{-}p(c_1)$
10 $\mathsf{T}{:}(*;1.1){:}{-}p(c_1)$	**18** $\mathsf{T}{:}(*;1.1){:}{-}p(c_1)$
11 $\mathsf{F}{:}(*;1.1){:}p(c_1)$	**19** $\mathsf{F}{:}(*;1.1){:}p(c_1)$
12 $\mathsf{T}{:}*{:}\Box p(c_1)$	**20** $\mathsf{T}{:}*{:}\Box p(c_1)$
13 $\mathsf{T}{:}(*;1){:}\Box p(c_1)$	**21** $\mathsf{T}{:}(*;1){:}\Box p(c_1)$
14 $\mathsf{T}{:}(*;1.1){:}p(c_1)$	**22** $\mathsf{T}{:}(*;1.1){:}p(c_1)$
$\perp$	$\perp$

Abbildung 6.2: Ein Tableaubeweis für die Gültigkeit der Formel G.

$$
\begin{array}{c}
\mathbf{10'}\ \ \mathsf{T}{:}(*;1.1){:}-r(c_1) \lor -s(c_1) \\
\mathbf{14'}\ \ \mathsf{T}{:}(*;1.1){:}r(c_1) \land s(c_1) \\
\mathbf{23}\ \ \mathsf{T}{:}(*;1.1;*){:}r(c_1) \land s(c_1) \\
\mathbf{24}\ \ \mathsf{T}{:}(*;1.1;*){:}r(c_1) \\
\mathbf{25}\ \ \mathsf{T}{:}(*;1.1;*){:}s(c_1) \\
\mathbf{26}\ \ \mathsf{T}{:}(*;1.1;*){:}-r(c_1) \lor -s(c_1)
\end{array}
$$

$$
\begin{array}{cc}
\mathbf{27}\ \ \mathsf{T}{:}(*;1.1;*){:}-r(c_1) & \mathbf{31}\ \ \mathsf{T}{:}(*;1.1;*){:}-s(c_1) \\
\mathbf{28}\ \ \mathsf{T}{:}(*;1.1;*;1){:}-r(c_1) & \mathbf{32}\ \ \mathsf{T}{:}(*;1.1;*;1){:}-s(c_1) \\
\mathbf{29}\ \ \mathsf{F}{:}(*;1.1;*;1){:}r(c_1) & \mathbf{33}\ \ \mathsf{F}{:}(*;1.1;*;1){:}s(c_1) \\
\mathbf{30}\ \ \mathsf{T}{:}(*;1.1;*;1){:}r(c_1) & \mathbf{34}\ \ \mathsf{T}{:}(*;1.1;*;1){:}s(c_1) \\
\bot & \bot
\end{array}
$$

Abbildung 6.3: Die Fortsetzung des Tableaus für G'.

schreiben wir $(*;1;*)$ anstatt $(*;(1;*))$, usw.). Die Tableauformel $14'$ enthält eine PL1-Formel. Darum wird das Faser-Regelschema angewendet, und 23 wird aus $14'$ abgeleitet; dies ist das Faser-Regelschema des Kalküls $\mathcal{C}_{(\widehat{\mathrm{K}},\mathrm{PL1})}$, der während der Iteration mit $\mathcal{C}_{\mathrm{PL1}}$ gefasert wurde, um $\mathcal{C}_{(\mathrm{PL1},(\widehat{\mathrm{K}},\mathrm{PL1}))}$ zu konstruieren. Das Regelschema von $\mathcal{C}_{\mathrm{PL1}}$ für α-Formeln wird benutzt, um 24 und 25 aus 23 abzuleiten; dann wird 26 aus $10'$ abgeleitet, indem wieder das Faser-Regelschema angewendet wird, und das Regelschema für β-Formeln wird angewendet, um 27 und 31 aus 26 abzuleiten. Das Atom $-p(c_1)$ in 27 enthält das modallogische und nicht das prädikatenlogische Negationszeichen. Also muß das Faser-Regelschema noch einmal angewendet werden, um 28 abzuleiten, was dann erlaubt, die Formel 29 durch Anwendung des Schemas für modallogische Negation abzuleiten. Die atomaren Tableauformeln 24 und 29 können nicht benutzt werden, um den Ast abzuschließen, denn ihre Markierungen sind verschieden. Also wird das Faser-Regelschema ein letztes mal angewendet, um 30 aus 24 abzuleiten. Dann wird der Ast abgeschlossen, und zwar durch Anwendung des Schemas für den Astabschluß auf 29 und 30.

7 Theorieschließen

7.1 Einführung

Theorieschließen ist eine wichtige Technik für die Verbesserung der Performanz von automatischen Beweissystemen; sie ist aus dem Bereich des Theorembeweisens in Prädikatenlogik erster Stufe wohlbekannt. Das spezifische Wissen aus einem gegebenen Bereich (einer Theorie) wird ausgenutzt, indem effiziente Methoden zur Deduktion in diesem Bereich eingesetzt werden.

Für das Automatische Beweisen in aus realen Anwendungen stammenden Bereichen ist das Theorieschließen besonders wichtig. Die Gleichheitstheorie wird zum Beispiel sehr häufig verwendet, aber die meisten Spezifikationen von Problemen realer Anwendungen verwenden auch noch andere Theorien; algebraische Theorien in mathematischen Problemen und Spezifikationen abstrakter Datentypen in der Software-Verifikation, um nur einige zu nennen.

Der wegbereitenden Arbeit Stickels (1985) folgend, wurden Methoden des Theorieschließens für die verschiedensten Kalkültypen beschrieben; dazu gehören Resolution (Stickel, 1985; Policriti & Schwartz, 1995), Pfad-Resolution (Murray & Rosenthal, 1987b), die Konnektionsmethode (Petermann, 1992; Petermann, 1993), Modellelimination (Baumgartner, 1992), Tableaukalküle mit starker Konnektionsbedingung (Baumgartner *et al.*, 1992; Furbach, 1994) und die Matrixmethode (Murray & Rosenthal, 1987a).

Folgendes abstraktes Modell wird gewöhnlich verwendet, um ein automatisches Deduktionssystem zu charakterisieren, daß eine Komponente zum Theorieschließen hat: Ein allgemeiner Vordergrund-Beweiser (*foreground reasoner*) ruft einen speziellen Hintergrund-Beweiser (*background reasoner*) auf, um Probleme aus einer bestimmten Theorie zu behandeln. Im Falle des tableaubasierten Theorembeweisens ist der Vordergrund-Beweiser ein Tableaukalkül (bzw. eine Implementierung eines Tableaukalküls), während der Hintergrund-Beweiser ein beliebiger Algorithmus ist, der für eine gegebene Prämisse eine Konklusion oder eine Menge von Konklusionen berechnet.

Dieses Modell paßt hervorragend in unseren allgemeinen Rahmen. Es ist nicht notwendig, eine neue Notation für das Theorieschließen einzuführen oder Bedingungen zu definieren, die die von einem Hintergrund-Beweiser berechneten Konklusionen erfüllen müssen, damit der resultierende Kalkül für eine Theorie $\mathcal{T}$ und eine Logik $\mathbf{L}$

korrekt und vollständig ist (wie es gewöhnlich in der Literatur über tableaubasiertes Theorieschließen geschieht, siehe (Beckert, 1998a)). Statt dessen nehmen wir den Standpunkt ein, daß eine Theorie $\mathcal{T}$ ein neues logisches System $\mathbf{L}_\mathcal{T}$ definiert. Diese Logik $\mathbf{L}_\mathcal{T}$ hat die gleiche Syntax wie die ursprüngliche Logik $\mathbf{L}$; aber Modelle von $\mathbf{L}_\mathcal{T}$ sind nur solche Modelle von $\mathbf{L}$, die $\mathcal{T}$ erfüllen. Dann können alle Begriffe und Methoden aus den vorangegangenen Kapiteln verwendet werden, ohne die Notwendigkeit spezielle Versionen für das Theorieschließen zu definieren. Ein Hintergrund-Beweiser ist dann eine Prozedur oder ein Algorithmus zur Berechnung von Konklusionen, so daß der entstehende Kalkül korrekt und vollständig für die Logik $\mathbf{L}_\mathcal{T}$ ist. Besondere Korrektheits- oder Vollständigkeitskriterien müssen für das Theorieschließen nicht definiert werden, sondern die Techniken zum Nachweise von Korrektheit und Vollständigkeit, die in Kapitel 3 beschrieben sind, können benutzt werden. Außerdem lassen sich alle Methoden zur Verbesserung der Effizienz einer tableaubasierten Beweisprozedur aus Kapiteln 4 und 5 anwenden, einschließlich Verwendung starrer und universeller Variablen.

Hintergrund-Beweiser sind für viele verschiedene Theorien entworfen worden, besonders für das Gleichheitsbeweisen; ein Überblick findet sich in (Baumgartner *et al.*, 1992; Furbach, 1994), für die Mengentheorie in (Cantone *et al.*, 1989). Die Deduktion in einzelnen Modellen, beispielsweise den natürlichen Zahlen, wird in (Bürckert, 1990) diskutiert.

7.2 Theorien

Wir definieren, daß jede erfüllbare Menge von Formeln eine Theorie sei.

Definition 7.2.1 Sei $\mathbf{L}$ eine Logik; und sei $\Sigma \in Sig$ eine Signatur. Dann ist eine erfüllbare Menge $\mathcal{T}(\Sigma) \subset Form(\Sigma)$ von Formeln eine *Theorie*.

Das logische System $\mathbf{L}_\mathcal{T}$ ist identisch mit $\mathbf{L}$ mit der einzigen Abweichung, daß für alle Signaturen $\Sigma \in Sig$ seine Menge $\mathcal{M}_\mathcal{T}(\Sigma)$ von Modellen nur solche Modelle von $\mathbf{L}(\Sigma)$ enthält, die $\mathcal{T}(\Sigma)$ erfüllen. □

In der Literatur wird häufig zusätzlich gefordert (neben der Erfüllbarkeit), daß eine Theorie unter der logischen Konsequenzrelation abgeschlossen sein muß. Ohne diese Einschränkung entfällt die Notwendigkeit, zwischen einer Theorie und der Menge der sie definierenden Axiome zu unterscheiden.

Definition 7.2.2 Eine Theorie $\mathcal{T}(\Sigma)$ ist *(endlich) axiomatisierbar*, wenn es eine (endliche) entscheidbare Menge $\Psi \subset Form(\Sigma)$ von Formeln (den Axiomen) gibt, so daß folgendes gilt: $\phi \in Form(\Sigma)$ ist genau dann in allen Modellen von $\mathbf{L}_\mathcal{T}(\Sigma)$ erfüllt, wenn ϕ in allen Modellen von $\mathbf{L}(\Sigma)$ erfüllt ist, die Ψ erfüllen. □

Die meisten Theorien von praktischem Interesse sind axiomatisierbar. Ein Beispiel für eine nicht axiomatisierbare Theorie ist die Menge $\mathcal{T}$ aller erfüllbaren PL1-Formeln. Wenn eine Theorie $\mathcal{T}$ axiomatisierbar ist, dann ist die Menge der in $\mathbf{L}_{\mathcal{T}}$ unerfüllbaren Formeln mit Hilfe eines Tableaukalküls für $\mathbf{L}$ aufzählbar.

Beispiel 7.2.3 Die in der Praxis wichtigste PL1-Theorie ist die Gleichheitstheorie $\mathcal{T}_{\approx}$. Sie besteht aus den folgenden Axiomen:

(1) $(\forall x)(x \approx x)$ (Reflexivität),

(2)

$$(\forall x_1) \cdots (\forall x_n)(\forall y_1) \cdots (\forall y_n)((x_1 \approx y_1 \wedge \ldots \wedge x_n \approx y_n) \to f(x_1, \ldots, x_n) \approx f(y_1, \ldots, y_n))$$

für alle Funktionssymbole $f \in F(\Sigma)$, wobei $n = \alpha_\Sigma(f)$ (Monotonie für Funktionssymbole),

(3)

$$(\forall x_1) \cdots (\forall x_n)(\forall y_1) \cdots (\forall y_n)((x_1 \approx y_1 \wedge \ldots \wedge x_n \approx y_n) \to (p(x_1, \ldots, x_n) \to p(y_1, \ldots, y_n)))$$

für alle Prädikatensymbole $p \in P(\Sigma)$, wobei $n = \alpha_\Sigma(p)$ (Monotonie für Prädikatensymbole),

Symmetrie und Transitivität von $\approx$ werden von Reflexivität (1) und Monotonie für Prädikatensymbole (3) impliziert (man beachte, daß $\approx \in P(\Sigma)$). □

Beispiel 7.2.4 Die PL1-Theorie $\mathcal{T}_<$ partieller Ordnungen besteht aus den folgenden Axiomen:

(1) $(\forall x)\neg(x < x)$ (Anti-Reflexivität),

(2) $(\forall x)(\forall y)(\forall z)((x < y) \wedge (y < z) \to (x < z))$ (Transitivität).

Die Theorie $\mathcal{T}_<$ ist endlich; im Gegensatz zur Gleichheitstheorie enthält sie keine Monotonie-Axiome. □

7.3 Beispiele für Hintergrund-Beweiser

Tabelle 7.1 enthält Beispiele für Konklusionen, die ein korrekter und vollständiger Hintergrund-Beweiser für die Gleichheitstheorie $\mathcal{T}_{\approx}$ berechnen könnte; die Prämissen und Konklusionen enthalten starre wie auch universelle Variablen.

Wie das folgende Beispiel illustriert, ist es im Zusammenhang mit Theorieschließen besonders wichtig, die Technik der universellen Variablen einzusetzen.

Prämisse	Konklusionen
$\{\mathsf{F}{:}(a \approx a)\}$	$\langle\{\{\bot\}\}\quad, id\rangle$
$\{\mathsf{F}{:}(X \approx a)\}$	$\langle\{\{\bot\}\}\quad, \{X \mapsto a\}\rangle$
$\{\mathsf{F}{:}(\boldsymbol{x} \approx a)\}$	$\langle\{\{\bot\}\}\quad, id\rangle$
$\{\mathsf{T}{:}p(a), \mathsf{F}{:}p(b)\}$	$\langle\{\{\mathsf{F}{:}(a \approx b)\}\}\quad, id\rangle$
$\{\mathsf{T}{:}p(f(a), f(b)), \mathsf{T}{:}(f(X) \approx X)\}$	$\langle\{\{\mathsf{T}{:}p(a, f(b))\}\}, \{X \mapsto a\}\rangle$ $\langle\{\{\mathsf{T}{:}p(f(a), b)\}\}, \{X \mapsto b\}\rangle$
$\{p(f(a), f(b)), f(\boldsymbol{x}) \approx \boldsymbol{x}\}$	$\langle\{\{\mathsf{T}{:}p(a, b)\}\}\quad, id\rangle$

Tabelle 7.1: Beispiele für Prämissen und Konklusionen unter Verwendung der Gleichheitstheorie.

Beispiel 7.3.1 Wir nehmen an, ein Tableauast enthalte die Gleichung $\mathsf{T}{:}(f(\boldsymbol{x}) \approx \boldsymbol{x})$ und die Atome $\mathsf{T}{:}p(f(a), f(b))$ und $\mathsf{F}{:}p(a, b)$. In diesem Fall kann die Konklusion $\langle\{\{\bot\}\}, id\rangle$ abgeleitet werden, und der Ast kann sofort geschlossen werden.

In einem Tableau mit starren Variablen dagegen, in dem ein (ähnlicher) Ast die Gleichung $\mathsf{T}{:}(f(X) \approx X)$ anstelle von $\mathsf{T}{:}(f(\boldsymbol{x}) \approx \boldsymbol{x})$ enthält, die nur die Ableitung der möglichen Konklusionen

$$\langle\{\{\mathsf{T}{:}(f(b) \approx b)\}\}, \{X \mapsto a\}\rangle \text{ und } \langle\{\{\mathsf{T}{:}(f(a) \approx a)\}\}, \{X \mapsto b\}\rangle$$

gestattet, kann der Ast nicht (sofort) geschlossen werden. □

Beispiel 7.3.2 Ein korrekter und vollständiger Grund-Tableaukalkül für das logische System $\mathrm{PL1}_{\mathcal{T}_>}$, das entsteht, wenn die Theorie partieller Ordnungen (Beispiel 7.2.4) zur Prädikatenlogik erster Stufe hinzugefügt wird, kann konstruiert werden, indem die Erweiterungsregel des Grundkalküls $\mathcal{C}_{\mathrm{PL1}}$ aus Abschnitt 3.6 mit den beiden Erweiterungsregel-Schemata verstärkt wird, die in Tabelle 7.2 dargestellt sind. Das heißt, für alle Prämissen $\Pi \in \mathit{TabForm}_{\mathrm{PL1}}(\Sigma^*)$ ist $\mathcal{E}_{\mathcal{T}_>}(\Sigma)(\Pi)$ die kleinste Menge, die die folgenden möglichen Konklusionen enthält:

- alle Konklusionen in $\mathcal{E}_{\mathrm{PL1}}(\Sigma)(\Pi)$,
- die Konklusion $\{\{\mathsf{T}{:}(t < t'')\}\}$ für alle Terme $t, t'' \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$, so daß es Formeln in Π $\mathsf{T}{:}(t < t')$ und $\mathsf{T}{:}(t' < t'')$ gibt, wobei $t' \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$,
- die Konklusion $\{\{\bot\}\}$, falls $\mathsf{T}{:}(t < t) \in \Pi$ für einen Term $t \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$. □

$$\frac{\begin{array}{c} t < t' \\ t' < t'' \end{array}}{t < t''}$$

für alle $t, t', t'' \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$

$$\frac{t < t}{\perp}$$

für alle $t \in \mathit{Term}^0_{\mathrm{PL1}}(\Sigma^*)$

Tabelle 7.2: Zusätzliche Erweiterungsregel-Schemata für die Theorie partieller Ordnungen (Beispiel 7.3.2).

Literaturverzeichnis

ANDREWS, PETER B. 1981. Theorem Proving through General Matings. *Journal of the ACM*, **28**, 193–214.

BAADER, F., & SCHULZ, K. 1995a. Combination of Constraints Solving Techniques, and Algebraic Point of View. *In: Proceedings, International Conference on Rewriting Techniques and Applications (RTA)*. LNCS 914. Springer.

BAADER, F., & SCHULZ, K. 1995b. On the Combination of Symbolic Constraints, Solution Domains and Constraint Solvers. *In: Proceedings, CP-95*. LNCS 976.

BAUMGARTNER, PETER. 1992. A Model Elimination Calculus with Built-in Theories. *Pages 30–42 of:* OHLBACH, H.-J. (ed), *Proceedings, German Workshop on Artificial Intelligence (GWAI)*. LNCS 671. Springer.

BAUMGARTNER, PETER. 1998. *Fairness Strategies in Hyper Tableaux*. Talk given as part of the Tele Seminar *Automated Theorem Proving and Applications* jointly held at the University of Karlsruhe and the University of Koblenz.

BAUMGARTNER, PETER, FURBACH, ULRICH, & PETERMANN, UWE. 1992. *A Unified Approach to Theory Reasoning*. Forschungsbericht 15/92. University of Koblenz.

BECKERT, BERNHARD. 1998a. Equality and Other Theories. *In:* D'AGOSTINO, M., GABBAY, D., HÄHNLE, R., & POSEGGA, J. (eds), *Handbook of Tableau Methods*. Kluwer. To appear.

BECKERT, BERNHARD. 1998b. Rigid E-Unification. *In:* BIBEL, WOLFGANG, & SCHMITT, PETER H. (eds), *Automated Deduction – A Basis for Applications*, vol. I. Kluwer, Dordrecht. To appear.

BECKERT, BERNHARD, & GABBAY, DOV. 1998. Fibring Semantic Tableaux. *Pages 77–92 of: Proceedings, International Conference on Theorem Proving with Analytic Tableaux and Related Methods, Oisterwijk, The Netherlands*. LNCS 1397. Springer.

BECKERT, BERNHARD, & GORÉ, RAJEEV. 1997. Free Variable Tableaux for Propositional Modal Logics. *Pages 91–106 of: Proceedings, International Conference on Theorem Proving with Analytic Tableaux and Related Methods, Pont-á-Mousson, France*. LNCS 1227. Springer.

BECKERT, BERNHARD, & HÄHNLE, REINER. 1992. An Improved Method for Adding Equality to Free Variable Semantic Tableaux. *Pages 507–521 of:* KAPUR, DEPAK (ed), *Proceedings, 11th International Conference on Automated Deduction (CADE), Saratoga Springs, NY, USA*. LNCS 607. Springer.

BECKERT, BERNHARD, & HÄHNLE, REINER. 1998. Analytic Tableaux. *In:* BIBEL, WOLFGANG, & SCHMITT, PETER H. (eds), *Automated Deduction — A Basis for Applications*, vol. I: Foundations. Kluwer, Dordrecht. To appear.

BECKERT, BERNHARD, & HARTMER, ULRIKE. 1998. A Tableau Calculus for Quantifier-free Set Theoretic Formulae. *Pages 93–107 of: Proceedings, International Conference on Theorem Proving with Analytic Tableaux and Related Methods, Oisterwijk, The Netherlands*. LNCS 1397. Springer.

BECKERT, BERNHARD, & POSEGGA, JOACHIM. 1995. lean$T^{A}P$: Lean Tableau-based Deduction. *Journal of Automated Reasoning*, **15**(3), 339–358.

BECKERT, BERNHARD, HÄHNLE, REINER, & SCHMITT, PETER H. 1993. The Even More Liberalized δ-Rule in Free Variable Semantic Tableaux. *Pages 108–119 of:* GOTTLOB, G., LEITSCH, A., & MUNDICI, D. (eds), *Proceedings, 3rd Kurt Gödel Colloquium (KGC), Brno, Czech Republic*. LNCS 713. Springer.

BETH, EVERT W. 1955. Semantic entailment and formal derivability. *Mededelingen van de Koninklijke Nederlandse Akademie van Wetenschappen, Afdeling Letterkunde, N.R.*, **18**(13), 309–342. Reprinted as pages 262–266 of: Karel Berka and Lothar Kreiser, editors. *Logik-Texte. Kommentierte Auswahl zur Geschichte der modernen Logik*. Akademie-Verlag, Berlin, 1986.

BETH, EVERT W. 1959. *The Foundations of Mathematics*. Amsterdam: North-Holland.

BIBEL, WOLFGANG. 1982. *Automated Theorem Proving*. Vieweg, Braunschweig.

BOWEN, K. A. 1982. Programming with Full First-order Logic. *Machine Intelligence*, **10**, 421–440. *First published as:* Technical Report 6-80, Syracuse University, Syracuse, NY, USA, 1980.

BRODA, KRYSIA. 1980. The Relationship between Semantic Tableaux and Resolution Theorem Proving. *In: Proceedings, Workshop on Logic, Debrecen, Hungary*. Also as technical report, Imperial College, Department of Computing, London, UK.

BROWN, FRANK MALLOY. 1978. Towards the Automation of Set Theory and its Logic. *Artificial Intelligence*, **10**, 281–316.

BÜRCKERT, H. 1990. A Resolution Principle for Clauses with Constraints. *Pages 178–192 of: Proceedings, 10th International Conference on Automated Deduction (CADE)*. LNCS 449. Springer.

CANTONE, DOMENICO. 1991. Decision Procedures for Elementary Sublanguages of Set Theory. X. *Journal of Automated Reasoning*, **7**, 193–230.

CANTONE, DOMENICO. 1997. A Fast Saturation Strategy for Set-theoretic Tableaux. *Pages 122–137 of: Proceedings, TABLEAUX, Pont-á-Mousson, France*. LNCS 1227. Springer.

CANTONE, DOMENICO, & FERRO, ALFREDO. 1995. Techniques of Computable Set Theory with Applications to Proof Verification. *Communications on Pure and Applied Mathematics*, **48**, 901–946.

CANTONE, DOMENICO, & SCHWARTZ, T. J. 1991. Decision Procedures for Elementary Sublanguages of Set Theory. XI. *Journal of Automated Reasoning*, **7**, 231–256.

CANTONE, DOMENICO, FERRO, ALFREDO, & SCHWARTZ, T. J. 1985. Decision Procedures for Elementary sublanguages of Set Theory. VI. *Communications on Pure and Applied Mathematics*, **38**, 549–571.

CANTONE, DOMENICO, FERRO, ALFREDO, & SCHWARTZ, T. J. 1987. Decision Procedures for Elementary Sublanguages of Set Theory. V. *Journal of Computer and Systems Sciences*, **34**, 1–18.

CANTONE, DOMENICO, FERRO, ALFREDO, & OMODEO, EUGENIO. 1989. *Computable Set Theory*. International Series of Monographs on Computer Science, vol. 6. Oxford University Press.

CARNIELLI, WALTER. 1987. Systematization of Finite Many-valued Logics through the Method of Tableaux. *Journal of Symbolic Logic*, **52**(2), 473–493.

COHEN, J., TRILLING, L., & WEGNER, P. 1974. A Nucleus of a Theorem Prover Described in ALGOL-68. *International Journal of Computer and Information Sciences*, **3**(1), 1–31.

D'AGOSTINO, MARCELLO, & GABBAY, DOV. 1996. Fibred Tableaux for Multi-implication Logics. *Pages 16–35 of:* MIGLIOLI, P., MOSCATO, U., MUNDICI, D., & ORNAGHI, M. (eds), *Proceedings, 5th International Workshop on Theorem Proving with Analytic Tableaux and Related Methods (TABLEAUX), Terrasini, Palermo, Italy*. LNCS 1071. Springer.

D'AGOSTINO, MARCELLO, GABBAY, DOV, HÄHNLE, REINER, & POSEGGA, JOACHIM (eds). 1998. *Handbook of Tableau Methods*. Kluwer, Dordrecht. To appear.

DE NIVELLE, HANS. 1997 (Feb.). *Implementation of Sequent Calculus and Set Theory*. Draft.

DE RIJKE, MAARTEN. 1997. Why Combine Logics? *Studia Logica*, **59**(1), 5–27.

DROSDOWSKI, GÜNTHER, MÜLLER, WOLFGANG, SCHOLZE-STUBENRECHT, WERNER, & WERMKE, MATTHIAS (eds). 1991. *Duden. Rechtschreibung der Deutschen Sprache*. 20th edn. Mannheim: Dudenverlag.

EGLY, UWE. 1998. Cuts in Tableaux. *In:* BIBEL, WOLFGANG, & SCHMITT, PETER H. (eds), *Automated Deduction – A Basis for Applications*, vol. I. Kluwer, Dordrecht. To appear.

FERRO, ALFREDO, & OMODEO, EUGENIO. 1987. Decision procedures for Elementary Sublanguages of Set Theory. VII. *Communications on Pure and Applied Mathematics*, **40**, 265–280.

FITTING, MELVIN. 1998. Introduction. *Chap. 1 of:* M., GABBAY, D., HÄHNLE, R., & POSEGGA, J. (eds), *Handbook of Tableau Methods*. Kluwer, Dordrecht. To appear.

FITTING, MELVIN C. 1969. *Intuitionistic Logic Model Theory and Forcing*. Amsterdam: North-Holland.

FITTING, MELVIN C. 1983. *Proof Methods for Modal and Intuitionistic Logics*. Synthese Library, vol. 169. Dordrecht: Reidel.

FITTING, MELVIN C. 1990. *First-Order Logic and Automated Theorem Proving*. New York: Springer. Second edition published in 1996.

FURBACH, ULRICH. 1994. Theory Reasoning in First Order Calculi. *Pages 139–156 of:* V. LUCK, K., & MARBURGER, H. (eds), *Proceedings, Third Workshop on Information Systems and Artificial Intelligence, Hamburg, Germany*. LNCS 777. Springer.

GABBAY, DOV. 1996a. Fibred Semantics and the Weaving of Logics. Part 1: Modal and Intuitionistic Logics. *Journal of Symbolic Logic*, **61**, 1057–1120.

GABBAY, DOV. 1996b. *Labelled Deductive Systems*. Oxford Logic Guides, no. 33. Oxford University Press.

GABBAY, DOV. 1996c. An Overview of Fibred Semantics and the Combination of Logics. *Pages 1–55 of:* BAADER, F., & SCHULZ, K. (eds), *Proceedings, Frontiers of Combining Systems*. Kluwer, Dordrecht.

GABBAY, DOV. 1999. *Fibring Logics*. Oxford University Press, New York.

GABBAY, DOV, & GOVERNATORI, GUIDO. 1998. Fibred Modal Tableaux. *In: Proccedings, First Interntational Workshop on Labelled Deduction, Freiburg, Germany*. To appear.

GENTZEN, GERHARD. 1935. Untersuchungen über das logische Schließen. *Mathematische Zeitschrift*, **39**, 176–210, 405–431. English translation in: Szabo, M., (ed.), *The Collected Papers of Gerhard Gentzen*, pages 68–131. North-Holland, Amsterdam, 1969.

GIUNCHIGLIA, FAUSTO, & SEBASTIANI, ROBERTO. 1996. A SAT-based Decision Procedure for $\mathcal{ALC}$. *Pages 304–314 of:* AIELLO, L., DOYLE, J., & SHAPIRO, S. (eds), *Proceedings, International Conference on Principals of Knowledge Representation and Reasoning (KR), Boston, USA*. Morgan Kaufmann.

GORÉ, RAJEEV. 1998. Tableau Methods for Modal and Temporal Logics. *Chap. 7 of:* M., GABBAY, D., HÄHNLE, R., & POSEGGA, J. (eds), *Handbook of Tableau Methods*. Kluwer, Dordrecht. To appear.

HÄHNLE, REINER. 1990. Towards an Efficient Tableau Proof Procedure for Multiple-valued Logics. *Pages 248–260 of: Proceedings, Workshop on Computer Science Logics (CLS), Heidelberg, Germany*. LNCS 533. Springer.

HÄHNLE, REINER. 1993. *Automated Deduction in Multiple-Valued Logics*. International Series of Monographs on Computer Science, vol. 10. Oxford University Press.

HÄHNLE, REINER, & KLINGENBECK, STEFAN. 1996. A-Ordered Tableaux. *Journal of Logic and Computation*, **6**(6), 819–834.

HÄHNLE, REINER, & PAPE, CHRISTIAN. 1997. Ordered Tableaux: Extensions and Applications. *Pages 173–187 of: Proceedings, International Conference on Theorem Proving with Analytic Tableaux and Related Methods, Pont-á-Mousson, France*. LNCS 1227. Springer.

HÄHNLE, REINER, & SCHMITT, PETER H. 1994. The Liberalized δ-rule in Free Variable Semantic Tableaux. *Journal of Automated Reasoning*, **13**(2), 211–222.

HÄHNLE, REINER, MURRAY, NEIL, & ROSENTHAL, ERIK. 1997. Completeness for Linear Regular Negation Normal Form Inference Systems. *In:* RAS, Z. (ed), *Proceedings, International Symposium on Methodologies for Intelligent Systems (ISMIS), Charlotte, NC, USA*. LNCS. Springer.

HARTMER, ULRIKE. 1997. *Erweiterung des Tableaukalküls mit freien Variablen um die Behandlung von Mengentheorie*. Diplomarbeit, Universität Karlsruhe.

HAYWARD, ARTHUR L., & SPARKES, JOHN J. 1968. *The Concise English Dictionary*. Fourth edn. Munich: Orbis Verlag.

HILBERT, DAVID, & BERNAYS, PAUL. 1939. *Grundlagen der Mathematik II*. Die Grundlehren der mathematischen Wissenschaften in Einzeldarstellungen mit besonderer Berücksichtigung der Anwendungsgebiete, vol. 50. Springer.

HINTIKKA, JAAKKO. 1955. Form and Content in Quantification Theory. *Acta Philosohica Fennica*, **8**, 7–55.

JECH, THOMAS. 1978. *Set Theory*. New York: Academic Press.

KANGER, STIG. 1957. *Provability in Logic*. Acta Universitatis Stockholmiensis, vol. 1. Almqvist & Wiksell, Stockholm.

KANGER, STIG. 1963. A Simplified Proof Method for Elementary Logic. *Pages 87–94 of:* BRAFFORT, P., & HIRSCHBERG, D. (eds), *Computer Programming and Formal Systems*. North Holland. Reprinted in (Siekmann & Wrightson, 1983, vol. 1, pages 364–371).

KORF, RICHARD E. 1985. Depth-First Iterative Deepening: An Optimal Admissible Tree Search. *Artificial Intelligence*, **27**, 97–109.

KRIPKE, SAUL. 1959. A Completeness Theorem in Modal Logic. *Journal of Symbolic Logic*, **24**(1), 1–14.

LETZ, REINHOLD, SCHUMANN, JOHANN, BAYERL, STEPHAN, & BIBEL, WOLFGANG. 1992. SETHEO: A High-Performance Theorem Prover. *Journal of Automated Reasoning*, **8**(2), 183–212.

LIS, Z. 1960. Wynikanie semantyczne a wynikanie formalne (logical consequence, semantic and formal). *Studia Logica*, **10**, 39–60. Polish, with Russian and English abstracts.

MATSUMOTO, K., & OHNISHI, M. 1957. Gentzen Method in Modal Calculi I. *Osaka Mathematical Journal*, **9**, 113–130.

MATSUMOTO, K., & OHNISHI, M. 1959. Gentzen Method in Modal Calculi II. *Osaka Mathematical Journal*, **11**, 115–120.

MEYER VIOL, WILFRIED. 1995. *Instantiational Logic: An Investigation into Reasoning with Instances. ILLC Dissertation Series 1995-11*. Ph.D. thesis, University of Utrecht.

MURRAY, NEIL V., & ROSENTHAL, ERIC. 1987a. Inference with path resolution and semantic graphs. *Journal of the ACM*, **34**(2), 225–254.

MURRAY, NEIL V., & ROSENTHAL, ERIC. 1987b. Theory Links: Applications to Automated Theorem Proving. *Journal of Symbolic Computation*, **4**, 173–190.

OPPACHER, F., & SUEN, E. 1988. HARP: A Tableau-Based Theorem Prover. *Journal of Automated Reasoning*, **4**, 69–100.

PASTRE, D. 1978. Automatic Theorem Proving in Set Theory. *Journal of Artificial Intelligence*, **10**, 1–27.

PETERMANN, UWE. 1992. How to Build-in an Open Theory into Connection Calculi. *Journal on Computer and Artificial Intelligence*, **11**(2), 105–142.

PETERMANN, UWE. 1993. *Building-in a Theory into a Connection Calculus with Positive Refinement*. Preprint Nr. 25. Naturwissenschaftlich-Theoretisches Zentrum, Universität Leipzig.

PFALZGRAF, J. 1991. Logical Fiberings and Polycontextural Systems. *In:* JORRAND, PH., & KELEMEN, J. (eds), *Fundamentals of Artificial Intelligence Resarch*. LNCS 535. Springer.

PFALZGRAF, J., & STOKKERMANS, K. 1994. On Robotics Scenarios and Modeling with Fibered Structures. *In:* PFALZGRAF, J., & WANG, D. (eds), *Automated Practical Reasoning: Algebraic Approaches*. Texts and Monographs in Symbolic Computation. Springer.

POLICRITI, ALBERTO, & SCHWARTZ, JACOB T. 1995. T-Theorem Proving I. *Journal of Symbolic Computation*, **20**, 315–342.

POPPLESTONE, R. J. 1967. Beth-Tree Methods in Automatic Theorem Proving. *Pages 31–46 of:* COLLINS, N., & MICHIE, D. (eds), *Machine Intelligence*, vol. 1. Oliver and Boyd.

PRAWITZ, DAG. 1960. An Improved Proof Procedure. *Theoria*, **26**, 102–139. Reprinted in (Siekmann & Wrightson, 1983, vol. 1, pages 162–199).

PRAWITZ, DAG, PRAWITZ, HÅKAN, & VOGHERA, NERI. 1960. A Mechanical Proof Procedure and its Realization in an Electronic Computer. *Journal of the ACM*, **7**(1–2), 102–128.

REEVES, STEVE V. 1987. Semantic Tableaux as a Framework for Automated Theorem-Proving. *Pages 125–139 of:* MELLISH, C., & HALLAM, J. (eds), *Advances in Artificial Intelligence (Proceedings of AISB-87)*. Wiley.

RESCHER, N., & URQUHART, A. 1971. *Temporal Logic*. Heidelberg: Springer.

ROUSSEAU, G. 1967. Sequents in Many-valued Logic I. *Fundamenta Methematica*, **60**, 23–33.

SCHMITT, PETER H. 1987. *The THOT Theorem Prover*. Tech. rept. 87.9.7. IBM Germany, Scientific Center, Heidelberg, Germany.

SCHÜTTE, KURT. 1956. Ein System des verknüpfenden Schließens. *Archiv für mathematische Logik und Grundlagenforschung*, **2**(2–4), 55–67.

SHULTS, BENJAMIN. 1997 (May). *Comprehension and Description in Tableaux*. Draft.

SIEKMANN, JÖRG, & WRIGHTSON, GRAHAM (eds). 1983. *Automation of Reasoning: Classical Papers in Computational Logic*. Springer.

SMULLYAN, RAYMOND M. 1968. *First-Order Logic*. Springer, Heidelberg. Second corrected edition published in 1995 by Dover Publications, New York.

STENZ, GERNOT. 1997. *Beweistransformation in Gentzenkalkülen*. Diplomarbeit, Universität Karlsruhe.

STENZ, GERNOT, AHRENDT, WOLFGANG, & BECKERT, BERNHARD. 1999. Proof Transformations from Search-oriented into Interaction-oriented Tableau Calculi. *Journal of Universal Computer Science*, **5**(3), 113–134.

STICKEL, MARK E. 1985. Automated Deduction by Theory Resolution. *Journal of Automated Reasoning*, **1**, 333–355.

SUCHON, W. 1974. La méthode de Smullyan de construire le calcul n-valent des propositions de Łukasiewicz avec implication et négation. *Reports on Mathematical Logic*, **2**, 37–42.

SURMA, S. J. 1984. An Algortihm for Axiomatizing every Finite Logic. *Pages 143–149 of:* RINE, D. C. (ed), *Computer Science and Multiple-valued Logic*, revised edn. Amsterdam: North Holland.

WANG, HAO. 1960. Toward Mechanical Mathematics. *IBM Journal of Research and Development*, **4**(1). Reprinted in (Siekmann & Wrightson, 1983, vol. 1, pages 244–264).

WEIDENBACH, CHRISTOPH. 1995. First-Order Tableaux with Sorts. *Journal of the IGPL*, **3**(6), 887–906.

WRIGHTSON, GRAHAM. 1984. *Semantic Tableaux, Unification and Links*. Technical Report CSD-ANZARP-84-001. Victoria University, Wellington, New Zealand.

Symbolverzeichnis

. . . but we need notions,
not notation.
— A. TARSKI, 1943

$\lceil\cdot\rceil$	Bijektion der Menge der modalen Formeln in die Menge der natürlichen Zahlen, S. 58
α	α-Formel, S. 46
β	β-Formel, S. 46
δ	δ-Formel, S. 46
γ	γ-Formel, S. 46
Γ	Menge von Markierungen, S. 54
ν	ν-Formel, S. 56
$\overline{\phi}$	Komplement der Tableauformel ϕ, Def. 3.2.1, S. 27
ϕ, ψ	Tableauformel, S. 27
π	π-Formel, S. 56
Π	Prämisse (endliche Menge von Tableauformeln), S. 33
$\sigma_{\|V}$	Einschränkung einer Substitution σ auf eine Menge V von Variablen, Def. 2.2.4, S. 13
$\sigma \circ \tau$	Komposition von Substitutionen σ und τ, Def. 2.2.4, S. 13
Σ^*_{fv}	Erweiterung einer Signatur Σ, die freie Variablen einführt, Def. 4.2.3, S. 93
Σ^*_{gd}	Erweiterung einer Grund-Signatur Σ, d. h., einer Signatur, die keine freien Variablen einführt, Def. 4.2.3, S. 93
$[\sigma]$	Äquivalenzklasse der Markierungen, die mit σ bis auf das Vorhandensein von Klammerungen übereinstimmen, Def. 3.7.1, S. 53
σ, τ	Substitution, S. 13
Σ^*	Erweiterung der Signatur Σ, S. 28
Σ	Signatur, S. 9
$\Box$	„immer“-Operator in Modallogiken, S. 18
$\Diamond$	„manchmal“-Operator in Modallogiken, S. 18
$\wedge$	Konjunktion, S. 16
$\rightarrow$	Implikation, S. 16
$\neg$	Negation, S. 16

$\vee$	Disjunktion, S. 16
$\exists$	Existentielle Quantifizierung, S. 16
$\forall$	Universelle Quantifizierung, S. 16
$\|\sigma\|$	Länge einer Markierung σ, Def. 3.7.1, S. 53
$\leq_w$	Gewichtsordnung, Def. 5.3.1, S. 166
$\rhd$	Zugänglichkeitsrelation auf der Menge der Markierungen, Def. 3.7.3, S. 54
$\leq^W$	Subsumtionsrelation auf Konklusionen mit starren Variablen, Def. 4.2.11, S. 97
$\leq^W$	Subsumtionsrelation auf Substitutionen, Def. 2.2.6, S. 14
$\models$	Beziehung zwischen Welten und Formeln, Def. 2.1.1, S. 9
$\subseteq_k$	k-Enthaltenseins-Relation, Def. 5.2.3, S. 157
$\cap$	Durchschnitt auf der Metaebene, S. 22
$\sqcap$	Durchschnitt auf der Objektebene, S. 22
$\cup$	Vereinigung auf der Metaebene, S. 22
$\sqcup$	Vereinigung auf der Objektebene, S. 22
$\setminus$	Mengen-Differenz, S. 22
$=$	Gleichheit auf der Metaebene, S. 21
$\approx$	Gleichheit auf der Objektebene, S. 21
$\in$	Elementbeziehung auf der Metaebene, S. 21
$\sqsubset\!\!\!-$	Elementbeziehung auf der Objektebene, S. 21
$\{\cdot\}_n$	Mengenkonstruktor auf der Objektebene, S. 22
$\sqsubseteq$	Teilmengenbeziehung auf der Objektebene, S. 21
$\subset$	Teilmengenbeziehung auf der Metaebene, S. 21
$\emptyset$	leere Menge auf der Metaebene, S. 22
$\emptyset$	leere Menge auf der Objektebene, S. 22
$Assoc(T, \phi)$	Menge der mit ϕ assoziierten Tableauformeln in T, Def. 5.2.2, S. 156
$Atom$	Menge aller atomaren Formeln (Atome) einer Logik, Def. 2.1.1, S. 9
B	Tableauast, S. 27
$CondLab(\mathbb{N})$	die Menge der bedingten Markierungen, die aus natürlichen Zahlen bestehen, Def. 3.7.1, S. 53
C	Konklusion, S. 30
$\mathcal{C}$	Tableaukalkül, S. 28
$dom(\sigma)$	Definitionsbereich der Substitution σ, Def. 2.2.4, S. 13
E	Extension (endliche Menge von Tableauformeln), S. 30
$\mathcal{E}$	Erweiterungsregel, S. 30
F, G	Formel, S. 9
$\mathfrak{F}, \mathfrak{G}$	Menge von Formeln, S. 9
$Form(B)$	Menge der Formeln auf einem Tableauast B, Def. 3.2.1, S. 27

$Form$	Menge einer Formeln einer Logik, Def. 2.1.1, S. 9
$Form^0_{\mathrm{PL1}}$	Menge aller geschlossenen PL1-Grundformeln (Sätze), Def. 2.3.1, S. 16
$Form^{nc}$	Menge aller PL1-Grundformeln, die (möglicherweise) Objektvariablen enthalten, die nicht durch einen Quantor gebunden sind, Def. 2.3.1, S. 16
fv	deutet an, daß es sich um eine Version mit freien Variablen eines Kalküls, einer Signatur usw. handelt, S. 93
$\mathcal{F}$	Faser-Funktion, Def. 6.2.1, S. 183
F	Vorzeichen (falsch), Def. 3.2.1, S. 27
$\mathcal{G}$	die Menge aller Mengenterme über Σ in einer Menge Π von MLSS-Tableauformeln über Σ^*, Def. 3.8.8, S. 75
id	die leere Substitution, Def. 2.2.4, S. 13
ipr	Menge der initialen Präfixe einer Markierung, Def. 3.7.1, S. 53
$Lab(\mathbb{N})$	die Menge der Markierungen, die aus natürlichen Zahlen bestehen, Def. 3.7.1, S. 53
L	formale Sprache, S. 11
$\mathbf{L}$	logisches System (Logik), S. 9
$\widehat{\mathbf{L}}$	Modallogik $\mathbf{L}$ ohne binäre Konnektive, S. 20
MLSSF	das logische System MLSSF, S. 22
MLSS	das logische System MLSS, S. 22
mv	deutet an, daß es sich um eine Version mit gemischt starren und universellen Variablen eines Kalküls, einer Signatur, usw. handelt, S. 93
$\mathcal{M}$	Menge aller Modelle einer Logik, Def. 2.1.1, S. 9
$\mathbf{m}$	Modell, S. 9
n, m	natürlichen Zahlen, S. 53
$[n]$	Position in einer Markierung, die bedingt oder unbedingt sein kann, S. 54
(n)	bedingte Position in einer Markierung, S. 53
$\mathbb{N}$	Menge der natürlichen Zahlen, S. 53
Ord	die Klasse aller Ordinalzahlen, Def. 2.6.4, S. 23
$\mathrm{PL1}$	Prädikatenlogik erster Stufe, S. 15
$ran(\sigma)$	Wertebereich einer Substitution σ, Def. 2.2.4, S. 13
rv	deutet an, daß es sich um eine Version mit starren Variablen eines Kalküls, einer Signatur usw. handelt, S. 93
R	Erreichbarkeitsrelation in Kripke-Rahmen, Def. 2.4.1, S. 18
$\mathcal{R}$	Realisation einer Menge von MLSS-Tableauformeln, S. 77
$\mathcal{R}$	Tableauregel, S. 28
Sig	Menge aller Signaturen einer Logik, Def. 2.1.1, S. 9
sko_{fv}	Funktion, die Formeln mit freien Variablen Skolem-Terme zuweist, Def. 4.2.28, S. 107
sko	Funktion, die Formeln Skolem-Terme zuweist, Def. 3.6.2, S. 46
$sort$	Funktion, die den Termen ihre Sorte zuweist, Def. 2.2.1, S. 11

$Subst$	Menge aller idempotenten Substitutionen einer Logik, Def. 2.2.4, S. 13
S	Menge von Sorten, S. 11
$TabForm$	Menge der Tableauformeln einer Logik, Def. 3.2.1, S. 27
$TabInterp$	Menge der Tableauinterpretationen eines Kalküls, Def. 3.4.2, S. 37
$TabTerm$	Menge der Terme in Tableauformeln, Def. 4.2.3, S. 93
$Term^{\mathrm{fv}}$	Menge aller (Nicht-Grund-)Terme einer Sprache, Def. 2.2.2, S. 12
$Term^{0}_{\mathrm{PL1}}$	Menge aller PL1-Grundterme, die keine Objektvariablen enthalten, Def. 2.3.1, S. 16
$Term^{nc}_{\mathrm{PL1}}$	Menge aller PL1-Grundterme, die (möglicherweise) Objektvariablen enthalten, Def. 2.3.1, S. 16
$Term$	Menge aller (Grund-)Terme einer Sprache, Def. 2.2.1, S. 11
$\mathcal{T}$	eine bestimmte Menge von Konstanten, Def. 3.8.8, S. 75
T	Vorzeichen (wahr), Def. 3.2.1, S. 27
T	Tableau, S. 27
$\mathcal{T}$	Theorie, S. 196
$\mathcal{T}_<$	Theorie partieller Ordnungen, S. 197
$\mathcal{T}_\approx$	Gleichheitstheorie, S. 197
uv	deutet an, daß es sich um eine Version mit universellen Variablen eines Kalküls, einer Signatur usw. handelt, S. 93
val	Auswertungsfunktion der Prädikatenlogik erster Stufe, Def. 2.3.2, S. 17
Var	Menge aller freien Variablen, Def. 2.2.2, S. 12
V, W	Menge von freien Variablen, S. 13
$\mathfrak{V}$	von-Neumann-Hierarchie von Mengen, Def. 2.6.4, S. 23
$\mathcal{V}$	eine bestimmte Menge von Termen, Def. 3.8.8, S. 75
w^0	initiale Welt, Def. 2.1.1, S. 9
W	Menge aller Welten eines Modells, Def. 2.1.1, S. 9
w	Gewichtsfunktion, S. 166
X, Y, Z	freie (starre) Variable, S. 12
x, y, z	Objektvariable, S. 12
$\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$	freie (universelle) Variable, S. 12

Index

Die Nummern von Seiten, auf denen ein Begriff definiert wird, sind fett gedruckt; wenn ein ganzer Abschnitt der Diskussion eines Begriffs oder Konzepts gewidmet ist, sind die Seitennummern dieses Abschnitts kursiv gesetzt.