Logik II

Sequenzenkalkül, Natürliches Schließen, Resolution

Peter Schroeder-Heister

Sommersemester 2000

Skriptum: Thomas Piecha

Universität Tübingen
Philosophisches Seminar und Wilhelm-Schickard-Institut für Informatik

Vorwort

Dieses Skriptum basiert auf einer zweistündigen Vorlesung im Sommersemester 2000, die sich an Studierende der Philosophie richtete. Vorausgegangen war als „Logik I“ eine ebenfalls zweistündige Vorlesung, die die Grundlagen der klassischen Junktoren- und Quantorenlogik behandelt und die Tableaumethode als Beweisverfahren eingeführt hatte. Die vorliegende Vorlesung entstand unter dem Gesichtspunkt, innerhalb der Logikausbildung in der Philosophie auch einmal eine beweistheoretisch orientierte Vorlesung anzubieten, als Alternative zur Modal- und Zeitlogik, die ich normalerweise unter dem Titel „Logik II“ ankündige. Ein erster Versuch dazu war eine im Sommersemester 1996 gehaltene Vorlesung mit dem Titel „Konsequenzenlogik“. Die damalige Vorlesung behandelte, anders als die jetzige, noch Schnittelimination, Sequenzenregeln für beliebige Junktoren und Quantoren, das intuitionistische natürliche Schließen nebst Normalisierung sowie einen Termkalkül, der die Curry-Howard-Korrespondenz zu behandeln erlaubte. Diese Themen sind in der neueren Vorlesung durch solche ersetzt, die sich auf Resolution und Logikprogrammierung beziehen.

Herrn Thomas Piecha bin ich für die Ausarbeitung des Skriptums besonders dankbar. In einigen Teilen konnte er sich auf den Entwurf eines Skriptums stützen, das Frau Birgit Munz für die „Konsequenzenlogik“ erstellt hatte.

Das Skriptum ist von mir durchgesehen. Somit bin ich für alle Fehler verantwortlich. Als Hilfsmittel zum Gebrauch neben Vorlesungen kann es natürlich kein durchformuliertes, ein Gebiet abdeckendes Lehrbuch ersetzen.

Peter Schroeder-Heister

Inhaltsverzeichnis

1 Tableaukalkül und klassischer Sequenzenkalkül

Definition 1.1 (Sprache)
Vokabular:

(i) Individuenkonstanten (beliebig viele, metasprachliche Variablen: $k, k_1, k_2, k', \ldots$),
Prädikatenkonstanten jeweils mit Stelligkeit (metaspr.: $P, Q, R, \ldots$),
(keine Funktionskonstanten)

(ii) Individuenparameter oder „freie“ Variablen (metaspr.: $a, b, c, \ldots$; abzählbar unendlich viele); gebundene Variablen (metaspr.: $x, y, z, \ldots$; abzählbar unendlich viele)

(iii) Logische Zeichen ($\forall, \exists, \wedge, \vee, \neg, \rightarrow, \leftrightarrow$ und Hilfszeichen).

Substitution in beliebigen Zeichenreihen:
$\alpha[{}^{\beta}\!/_{a}]$: Ersetzung des Parameters a in α durch β
Terme: Konstanten und Parameter: $t, s, \ldots$
Atomformeln: $P(t_1, \ldots, t_n)$ auch $Pt_1 \ldots t_n$
Formeln: werden aus Atomformeln gebildet: $A \wedge B$, $A \vee B$, $A \rightarrow B$, $\neg A$, $\exists x A[{}^{x}\!/_{a}]$, $\forall x A[{}^{x}\!/_{a}]$, wobei x nicht in A vorkommt.
(Abkürzende Schreibweise: $\exists x A(x)$, $\forall x A(x)$)
Klammerung: wie üblich
Signierte Formeln: $\mathrm{w}\, A$, $\mathrm{f}\, A$ für Formeln A (metaspr.: $\sigma, \tau, \ldots$).
Endliche Mengen signierter Formeln: $\Sigma, \Gamma, \Delta, \ldots$

Bemerkungen.

- $\forall x(Px \wedge \exists x Qx)$ ist nicht wohlgeformt.
  $\forall x Px \wedge \exists x Qx$ ist wohlgeformt.
- Die Objektzeichen selbst treten nicht in Erscheinung. Nur metasprachliche Variablen werden verwendet. Wir reden nicht über konkrete Formeln, sondern nur über Formeln von bestimmter Form.

Definition 1.2
Seien $U = \{A_1, \ldots, A_n\}$ und $V = \{B_1, \ldots, B_m\}$ $(n, m \geq 0)$ endliche Formelmengen. Ein Tableau für $\langle U, V \rangle$ ist eine Baumstruktur von signierten Formeln, die mit

$$\begin{array}{c} \sigma_1 \\ \sigma_2 \\ \vdots \\ \sigma_{n+m} \end{array}$$

beginnt, wobei $\{\sigma_1, \ldots, \sigma_{n+m}\} = \{\mathrm{w}\, A_1, \ldots, \mathrm{w}\, A_n, \mathrm{f}\, B_1, \ldots, \mathrm{f}\, B_m\} = \mathrm{w}\, U \cup \mathrm{f}\, V$, und die nach folgenden Regeln erzeugt wird:

(w ¬) $$\begin{array}{c} \mathrm{w}\,\neg A \\ | \\ \mathrm{f}\,A \end{array}$$

(f ¬) $$\begin{array}{c} \mathrm{f}\,\neg A \\ | \\ \mathrm{w}\,A \end{array}$$

(w ∧) $$\begin{array}{c} \mathrm{w}\,A \land B \\ | \\ \mathrm{w}\,A \\ \mathrm{w}\,B \end{array}$$

(f ∧) $$\begin{array}{c} \mathrm{f}\,A \land B \\ \diagup \quad \diagdown \\ \mathrm{f}\,A \qquad \mathrm{f}\,B \end{array}$$

(w ∨) $$\begin{array}{c} \mathrm{w}\,A \lor B \\ \diagup \quad \diagdown \\ \mathrm{w}\,A \qquad \mathrm{w}\,B \end{array}$$

(f ∨) $$\begin{array}{c} \mathrm{f}\,A \lor B \\ | \\ \mathrm{f}\,A \\ \mathrm{f}\,B \end{array}$$

(w →) $$\begin{array}{c} \mathrm{w}\,A \to B \\ \diagup \quad \diagdown \\ \mathrm{f}\,A \qquad \mathrm{w}\,B \end{array}$$

(f →) $$\begin{array}{c} \mathrm{f}\,A \to B \\ | \\ \mathrm{w}\,A \\ \mathrm{f}\,B \end{array}$$

(w ∀) $$\begin{array}{c} \mathrm{w}\,\forall x A(x) \\ | \\ \mathrm{w}\,A(t) \end{array}$$

(f ∀) $$\begin{array}{c} \mathrm{f}\,\forall x A(x) \\ | \\ \mathrm{f}\,A(a) \\ a \text{ neuer Parameter} \end{array}$$

(w ∃) $$\begin{array}{c} \mathrm{w}\,\exists x A(x) \\ | \\ \mathrm{w}\,A(a) \\ a \text{ neuer Parameter} \end{array}$$

(f ∃) $$\begin{array}{c} \mathrm{f}\,\exists x A(x) \\ | \\ \mathrm{f}\,A(t) \end{array}$$

$\begin{array}{c} \sigma \\ | \\ \tau \end{array}$ bzw. $\begin{array}{c} \sigma \\ | \\ \tau_1 \\ \tau_2 \end{array}$ bedeutet: Ein Zweig, in dem σ (an irgendeiner Stelle) vorkommt, darf mit τ bzw. $\begin{array}{c} \tau_1 \\ \tau_2 \end{array}$ fortgesetzt werden.

$\begin{array}{c} \sigma \\ \diagup \quad \diagdown \\ \tau_1 \qquad \tau_2 \end{array}$ bedeutet: Ein Zweig in dem σ (an irgendeiner Stelle) vorkommt, darf zu τ_1 und τ_2 verzweigt werden. τ bzw. τ_1 und τ_2 müssen also nicht *unmittelbar* auf σ folgen. Die Parameterbedingung

$$\begin{array}{c} \sigma \\ | \\ \tau \end{array} \quad a \text{ neu}$$

besagt: a darf vor τ im Zweig nicht vorkommen.
Ein Tableau für $\langle U, V \rangle$ heißt *geschlossen*, falls in jedem Zweig sowohl $\mathrm{w}\,A$ als auch $\mathrm{f}\,A$ für ein A vorkommt.

Beispiel.
$\langle\{A \to B, \neg B\}, \{\neg A\}\rangle$

$$
\begin{array}{ccc}
 & \mathrm{w}\, A \to B & \\
 & | & \\
 & \mathrm{w}\, \neg B & \\
 & | & \\
 & \mathrm{f}\, \neg A & \\
 / & & \backslash \\
\mathrm{f}\, A & & \mathrm{w}\, B \\
| & & | \\
\mathrm{w}\, A & & \mathrm{f}\, B
\end{array}
$$

Definition 1.3 (Kumulatives Tableau)
Ein *kumulatives Tableau* für $\langle U, V\rangle$ ist eine Baumstruktur von einer endlichen Menge signierter Formeln, die mit $\mathrm{w}\, U \cup \mathrm{f}\, V$ beginnt und nach folgenden Regeln erzeugt wird:

$$(\mathrm{w}\, \neg)\ \frac{\Sigma \cup \{\mathrm{w}\, \neg A\}}{\Sigma \cup \{\mathrm{w}\, \neg A\} \cup \{\mathrm{f}\, A\}}$$

$$(\mathrm{f}\, \neg)\ \frac{\Sigma \cup \{\mathrm{f}\, \neg A\}}{\Sigma \cup \{\mathrm{f}\, \neg A\} \cup \{\mathrm{w}\, A\}}$$

$$(\mathrm{w}\, \wedge)\ \frac{\Sigma \cup \{\mathrm{w}\, A \wedge B\}}{\Sigma \cup \{\mathrm{w}\, A \wedge B\} \cup \{\mathrm{w}\, A\} \cup \{\mathrm{w}\, B\}}$$

$$(\mathrm{f}\, \wedge)\ \frac{\Sigma \cup \{\mathrm{f}\, A \wedge B\}}{\Sigma \cup \{\mathrm{f}\, A \wedge B\} \cup \{\mathrm{f}\, A\} \quad \Sigma \cup \{\mathrm{f}\, A \wedge B\} \cup \{\mathrm{f}\, B\}}$$

$$(\mathrm{w}\, \vee)\ \frac{\Sigma \cup \{\mathrm{w}\, A \vee B\}}{\Sigma \cup \{\mathrm{w}\, A \vee B\} \cup \{\mathrm{w}\, A\} \quad \Sigma \cup \{\mathrm{w}\, A \vee B\} \cup \{\mathrm{w}\, B\}}$$

$$(\mathrm{f}\, \vee)\ \frac{\Sigma \cup \{\mathrm{f}\, A \vee B\}}{\Sigma \cup \{\mathrm{f}\, A \vee B\} \cup \{\mathrm{f}\, A\} \cup \{\mathrm{f}\, B\}}$$

$$(\mathrm{w}\, \to)\ \frac{\Sigma \cup \{\mathrm{w}\, A \to B\}}{\Sigma \cup \{\mathrm{w}\, A \to B\} \cup \{\mathrm{f}\, A\} \quad \Sigma \cup \{\mathrm{w}\, A \to B\} \cup \{\mathrm{w}\, B\}}$$

$$(\mathrm{f}\, \to)\ \frac{\Sigma \cup \{\mathrm{f}\, A \to B\}}{\Sigma \cup \{\mathrm{f}\, A \to B\} \cup \{\mathrm{w}\, A\} \cup \{\mathrm{f}\, B\}}$$

$$(\mathrm{w}\, \forall)\ \frac{\Sigma \cup \{\mathrm{w}\, \forall x A(x)\}}{\Sigma \cup \{\mathrm{w}\, \forall x A(x)\} \cup \{\mathrm{w}\, A(t)\}}$$

$$(\mathrm{f}\,\forall)\ \frac{\Sigma \cup \{\mathrm{f}\,\forall x A(x)\}}{\Sigma \cup \{\mathrm{f}\,\forall x A(x)\} \cup \{\mathrm{f}\,A(a)\}} \quad a \text{ nicht in } \Sigma \cup \{\mathrm{f}\,\forall x A(x)\}$$

$$(\mathrm{w}\,\exists)\ \frac{\Sigma \cup \{\mathrm{w}\,\exists x A(x)\}}{\Sigma \cup \{\mathrm{w}\,\exists x A(x)\} \cup \{\mathrm{w}\,A(a)\}} \quad a \text{ nicht in } \Sigma \cup \{\mathrm{f}\,\exists x A(x)\}$$

$$(\mathrm{f}\,\exists)\ \frac{\Sigma \cup \{\mathrm{f}\,\exists x A(x)\}}{\Sigma \cup \{\mathrm{f}\,\exists x A(x)\} \cup \{\mathrm{f}\,A(t)\}}$$

Hier bedeutet $\dfrac{\Sigma}{\tau}$, daß τ *unmittelbar* unter Σ steht und $\dfrac{\Sigma}{\tau_1 \quad \tau_2}$, daß Σ sich *unmittelbar* zu τ_1 und τ_2 verzweigt. (Wir benutzen waagerechte Schlußstriche, um auszudrücken, daß die Konsequenz *unmittelbar* auf die Prämisse folgt.)
Ein kumulatives Tableau heißt *geschlossen*, wenn jedes Blatt einen Widerspruch enthält.

Satz 1.1
Zu jedem geschlossenen Tableau für $\langle U, V\rangle$ gibt es ein geschlossenes kumulatives Tableau für $\langle U, V\rangle$ und umgekehrt.

Beweis.
Nach Konstruktion. $\square$

Beispiel.
Kumulatives Tableau:

$$\begin{array}{ll} (\mathrm{w}\,\vee) & \dfrac{\{\mathrm{w}\,\neg A \vee B, \mathrm{w}\,A, \mathrm{f}\,B\}}{\{\mathrm{w}\,\neg A \vee B, \mathrm{w}\,A, \mathrm{f}\,B, \mathrm{w}\,\neg A\} \qquad \{\mathrm{w}\,\neg A \vee B, \mathrm{w}\,A, \underline{\mathrm{f}\,B}, \underline{\mathrm{w}\,B}\}} \\ (\mathrm{w}\,\neg) & \dfrac{}{\{\mathrm{w}\,\neg A \vee B, \underline{\mathrm{w}\,A}, \mathrm{f}\,B, \mathrm{w}\,\neg A, \underline{\mathrm{f}\,A}\}} \end{array}$$

Tableau geschlossen, da in beiden Mengen Widerspruch!

Definition 1.4 (Modifiziertes kumulatives Tableau)
Ein *modifiziertes kumulatives Tableau* für $\langle U, V\rangle$ ist wie in Definition 1.3 eine Baumstruktur von Mengen signierter Formeln, die mit $\mathrm{w}\,U \cup \mathrm{f}\,V$ beginnt, jedoch nach folgenden Regeln erzeugt wird:

$$(\mathrm{w}\,\wedge)\ \frac{\Sigma \cup \{\mathrm{w}\,A \wedge B\}}{\Sigma \cup \{\mathrm{w}\,A\} \cup \{\mathrm{w}\,B\}} \qquad (\mathrm{f}\,\wedge)\ \frac{\Sigma \cup \{\mathrm{f}\,A \wedge B\}}{\Sigma \cup \{\mathrm{f}\,A\} \quad \Sigma \cup \{\mathrm{f}\,B\}}$$

$$(\mathrm{w}\,\vee)\ \frac{\Sigma \cup \{\mathrm{w}\,A \vee B\}}{\Sigma \cup \{\mathrm{w}\,A\} \quad \Sigma \cup \{\mathrm{w}\,B\}} \qquad (\mathrm{f}\,\vee)\ \frac{\Sigma \cup \{\mathrm{f}\,A \vee B\}}{\Sigma \cup \{\mathrm{f}\,A\} \cup \{\mathrm{f}\,B\}}$$

$$\vdots \qquad\qquad\qquad \vdots$$

$$(\mathrm{w}\,\forall) \quad \frac{\Sigma \cup \{\mathrm{w}\,\forall x A(x)\}}{\Sigma \cup \{\mathrm{w}\,A(t)\}} \qquad\qquad (\mathrm{f}\,\forall) \quad \frac{\Sigma \cup \{\mathrm{f}\,\forall x A(x)\}}{\Sigma \cup \{\mathrm{f}\,A(a)\}} \quad a \text{ neu}$$

Bemerkung.
Modifizierte kumulative Tableaus unterscheiden sich von kumulativen Tableaus dadurch, daß die jeweils ausgewertete signierte Formel σ weggelassen werden *kann*. (Sie *muß* nicht weggelassen werden, da sie in Σ vorkommen kann.)
Achtung: Das Weglassen von signierten Formeln mit Quantoren kann dazu führen, daß die Allgemeingültigkeit mancher Formeln nicht mehr nachgewiesen werden kann, da bei quantorenlogischen Formeln bisweilen dasselbe σ mehrfach ausgewertet werden muß. Man weise z.B. die Allgemeingültigkeit von $\exists x(Px \rightarrow \forall y Py)$ nach.

Satz 1.2
Zu jedem geschlossenen Tableau für $\langle U, V\rangle$ gibt es ein geschlossenes modifiziertes kumulatives Tableau für $\langle U, V\rangle$ und umgekehrt.

Beweis.
„$\Longrightarrow$“: Jedes kumulative Tableau ist auch ein modifiziertes kumulatives Tableau.
„$\Longleftarrow$“: Durch Hinzufügen geeigneter signierter Formeln in jedem Schritt erweitern wir das gegebene Tableau zu einem kumulativen Tableau. $\square$

Definition 1.5 (Sequenz)
Eine Sequenz hat die Form $X \vdash Y$, wobei X und Y endliche Mengen von Formeln sind.
D.h.: $\{A_1, \ldots, A_n\} \vdash \{B_1, \ldots, B_m\} \quad n, m \geq 0$
Wir schreiben auch $A_1, \ldots, A_n \vdash B_1, \ldots, B_m \quad n, m \geq 0$ oder $X, A \vdash Y, B$

Definition 1.6 (Klassischer Sequenzenkalkül)
Der *klassische Sequenzenkalkül* LK ist wie folgt definiert:
Eine Ableitung in LK ist eine baumförmige Struktur von Sequenzen, die nach folgenden Regeln generiert wird:

Anfangssequenz: (I) $\dfrac{}{\mathrm{A,\ X{\vdash}Y,\ A}}$ (prämissenfreie Regel)

$$(\vdash \neg) \quad \frac{A, X \vdash Y}{X \vdash Y, \neg A} \qquad\qquad (\neg \vdash) \quad \frac{X \vdash Y, A}{\neg A, X \vdash Y}$$

$$(\wedge \vdash) \quad \frac{A, B, X \vdash Y}{A \wedge B, X \vdash Y} \qquad\qquad (\vdash \wedge) \quad \frac{X \vdash Y, A \quad X \vdash Y, B}{X \vdash Y, A \wedge B}$$

$$(\vee \vdash) \quad \frac{A, X \vdash Y \quad B, X \vdash Y}{A \vee B, X \vdash Y} \qquad\qquad (\vdash \vee) \quad \frac{X \vdash Y, A, B}{X \vdash Y, A \vee B}$$

$(\to\vdash)\quad \dfrac{X\vdash Y, A \quad B, X\vdash Y}{A\to B, X\vdash Y}$ $\qquad (\vdash\to)\quad \dfrac{A, X\vdash Y, B}{X\vdash Y, A\to B}$

$(\forall\vdash)\quad \dfrac{A(t), X\vdash Y}{\forall x A(x), X\vdash Y}$ $\qquad (\vdash\forall)\quad \dfrac{X\vdash Y, A(a)}{X\vdash Y, \forall x A(x)}$ a nicht in X, Y

$(\exists\vdash)\quad \dfrac{A(a), X\vdash Y}{\exists x A(x), X\vdash Y}$ a nicht in X, Y $\qquad (\vdash\exists)\quad \dfrac{X\vdash Y, A(t)}{X\vdash Y, \exists x A(x)}$

$X\vdash Y$ heißt ableitbar in LK ($X\vdash_{\mathrm{LK}} Y$), falls es eine Ableitung in LK gibt, die mit $X\vdash Y$ endet.

Satz 1.3
Zu jedem geschlossenen Tableau für $\langle U, V\rangle$ gibt es eine Ableitung in LK von $U\vdash V$ und umgekehrt.

Beweis.
„$\Longrightarrow$“

- Betrachte gemäß Satz 1.2 ein geschlossenes modifiziertes kumulatives Tableau für $\langle U, V\rangle$.
- Wir schreiben eine Menge Σ als $\{A : \mathrm{w}\,A \in \Sigma\}\vdash\{A : \mathrm{f}\,A \in \Sigma\}$.[1]
- Die Regeln ($\mathrm{w}\,*$) gehen in ($*\vdash$), die Regeln ($\mathrm{f}\,*$) in ($\vdash *$) über, wenn man sie rückwärts liest.
- Die Bedingung, daß das Tableau geschlossen ist, geht für alle Blätter in (I) über.
- Der Anfang $\mathrm{w}\,U \cup \mathrm{f}\,V$ geht in $U\vdash V$ über.
- Man kann das Tableau also als auf den Kopf gestellte Ableitung in LK verstehen.

„$\Longleftarrow$“

- Gegeben sei eine Ableitung in LK von $\langle U, V\rangle$.
- Wir schreiben in der Ableitung $X\vdash Y$ als $\mathrm{w}\,X \cup \mathrm{f}\,Y$.
- ($*\vdash$) geht in ($\mathrm{w}\,*$), ($\vdash *$) geht in ($\mathrm{f}\,*$) über, wenn es rückwärts gelesen wird.
- (I) besagt die Geschlossenheit eines Zweiges.
- Die Endsequenz wird zur Wurzel des Tableaus. □

[1]Beispiel: $\{\mathrm{w}\,A, \mathrm{f}\,B, \mathrm{f}\,C, \mathrm{w}\,D\}$ als $\{A, D\}\vdash\{B, C\}$

2 Einige Eigenschaften von LK

Bemerkungen.
In $X \vdash Y$ heißt X *Antezedens* und Y *Sukzedens.*
Unsere Motivation für $X \vdash Y$:

> Es ist nicht der Fall, daß alle Elemente von X wahr und alle Elemente von Y falsch sind.
> D.h.: Wenn alle Elemente von X wahr sind, dann ist mindestens ein Element von Y wahr.
> D.h.: X konjunktiv, Y disjunktiv.

Grenzfälle:

- $A_1, \ldots, A_n \vdash A$: Wenn $A_1, \ldots, A_n$ wahr, dann A wahr. (Konsequenz)
- $A_1, \ldots, A_n \vdash$: Nicht alle $A_1, \ldots, A_n$ sind wahr, d.h. mindestens ein A_i ist falsch.
- $\vdash A$: A ist logisch wahr.

Beispiele.

(a) $\dfrac{A \vdash A}{\neg A, A \vdash}\,(\neg\vdash)$ (b) $\dfrac{\dfrac{A \vdash A}{\vdash A, \neg A}\,(\vdash\neg)}{\vdash A \vee \neg A}\,(\vdash\vee)$

Historisches.
Der Sequenzenkalkül wurde von Gerhard Gentzen (1909-1945) 1934 eingeführt.

Beobachtungen.

(i) Für jedes logische Zeichen gibt es genau ein Paar von Regeln: Einführungsregeln links und Einführungsregeln rechts.

(ii) Jede Regel betrifft nur *ein* logisches Zeichen. (Separativität)

(iii) Es gibt nur *ein* Axiom. (Regellogik)

(iv) In den Prämissen einer Regel kommen nur die unmittelbaren Teilformeln der eingeführten Formel vor. Alle anderen Formeln bleiben identisch (parametrisch).
(Wir betrachten dabei $A(t)$ und $A(a)$ als Teilformel von $\forall x A(x)$ und $\exists x A(x)$.)

Definition 2.1 (Terminologie)
Das logisch zusammengesetzte Formelvorkommen, auf das sich eine Regelanwendung bezieht, heißt *Hauptformel* der Regelanwendung; die Teilvorkommen davon, auf die sich die Anwendung bezieht, heißen *Nebenformeln.* Die übrigen (parametrischen) Formelvorkommen heißen *Seitenformeln.*
Der Parameter einer Regelanwendung mit Parameterbedingung heißt *Eigenparameter.*
Mit $S, S_1, S_2, \ldots$ bezeichnen wir *Sequenzen.*

Satz 2.1 (Teilformelprinzip)

(i) Jede Ableitung für $X \vdash Y$ in LK enthält nur Teilformeln von Formeln in $X \vdash Y$.

(ii) In jeder Ableitung für $X \vdash Y$ werden nur Regeln für diejenigen logischen Zeichen angewendet, die in X oder Y vorkommen.

Beweis.
Durch Inspektion der Regeln. □

Bemerkung.
Dies ist nicht selbstverständlich: Für andere Logikkalküle gilt dies nicht.

Lemma 2.1 (Substitutionseigenschaft)
Sei eine Ableitung von $X(a) \vdash Y(a)$ gegeben[2]. Dann gibt es eine Ableitung von $X(t) \vdash Y(t)$ für einen beliebigen Term t. Diese Ableitung geht aus der ursprünglichen Ableitung nur durch Substitution hervor. Die Baumstruktur und die angewendeten Regeln bleiben unverändert.

Beweis.
Induktion über dem Aufbau von Ableitungen

(i) Mit $X(a) \vdash Y(a)$ ist auch $X(t) \vdash Y(t)$ Anfangssequenz.

(ii) Für alle Regelanwendungen ohne Eigenparameter gilt:

Mit $\dfrac{X_1(a) \vdash Y_1(a)}{X_2(a) \vdash Y_2(a)}$ ist auch $\dfrac{X_1(t) \vdash Y_1(t)}{X_2(t) \vdash Y_2(t)}$ eine Anwendung derselben Regel.

Entsprechendes gilt für Regeln mit zwei Prämissen.

(iii) Betrachte

$$(\vdash \exists)\ \frac{X_1(a) \vdash Y_1(a), A(a,b)}{X_1(a) \vdash Y_1(a), \exists x A(a,x)} \quad (b \text{ nicht in } X_1(a), Y_1(a))$$

Falls t von b verschieden, ist

$$(\vdash \exists)\ \frac{X_1(t) \vdash Y_1(t), A(t,b)}{X_1(t) \vdash Y_1(t), \exists x A(t,x)} \quad \text{korrekt.}$$

Falls $t = b$, erzeugen wir zunächst mit Induktionsvoraussetzung eine Ableitung von $X_1(a) \vdash Y_1(a), A(a,c)$ für ein neues c, das nicht in $X_1(a), Y_1(a), A(a,b)$ vorkommt und von t verschieden ist. Dann ist

$$(\vdash \exists)\ \frac{X_1(t) \vdash Y_1(t), A(t,c)}{X_1(t) \vdash Y_1(t), \exists x A(t,x)} \quad \text{eine korrekte Regelanwendung.}$$

Entsprechendes gilt für die anderen Quantorenregeln. □

[2] d.h., in X und Y kommt irgendwo der Parameter a vor.

Bemerkung.
Im folgenden können wir immer annehmen, daß Eigenparameter separiert sind, d.h. nur über einer Regelanwendung vorkommen und sonst nirgendwo in der Ableitung.

Lemma 2.2 (Verdünnung)
Sei eine Ableitung von $X \vdash Y$ gegeben, dann gibt es eine Ableitung von $X_1, X \vdash Y, Y_1$ für beliebige X_1, Y_1. Diese Ableitung hat dieselbe Baumstruktur wie die ursprüngliche.

Beweis.
Falls X_1, Y_1 Parameter enthalten, die in der Ableitung von $X \vdash Y$ als Eigenparameter vorkommen, ersetzt man zunächst mit Lemma 2.1 in der Ausgangsableitung diese Eigenparameter durch neue, die weder in der Ableitung noch in X_1, Y_1 vorkommen. Wenn man dann darin jede Sequenz $X' \vdash Y'$ durch $X', X_1 \vdash Y_1, Y'$ ersetzt, ergibt sich wieder eine Ableitung. □

Bemerkung.
Das drückt man auch so aus:

Die Regel $\quad (\mathrm{V})\ \dfrac{X \vdash Y}{X_1, X \vdash Y, Y_1} \quad$ ist *zulässig* in LK.

Definition 2.2 (Vertauschbarkeit von Regeln)
Das Regelpaar $\mathrm{R}_1|\mathrm{R}_2$ heißt *vertauschbar*, falls folgendes immer gilt:

Eine Ableitung ende mit einer Anwendung von R_2, eine von deren Prämissen mit einer Anwendung von R_1, wobei die Hauptformel von R_1 eine Seitenformel von R_2 sei. Dann lassen sich die Anwendungen von R_1 und R_2 so vertauschen, daß die Ableitung mit einer Anwendung von R_1 endet und eine von deren Prämissen mit einer Anwendung von R_2, wobei die Haupformel von R_2 eine Seitenformel von R_1 ist. Alle sonstigen Teilableitungen bleiben strukturell unverändert (d.h. haben dieselbe Baumstruktur).

Bemerkung.
Vertauschbarkeit von Regeln ist nicht symmetrisch.

Satz 2.2 (Kleene 1952)
Alle Regelpaare außer $\quad (\forall\vdash)|(\vdash\forall) \quad (\forall\vdash)|(\exists\vdash)$
$\quad (\vdash\exists)|(\vdash\forall) \quad (\vdash\exists)|(\exists\vdash)$
sind vertauschbar.

Beweis.
(i) Exemplarisches Gegenbeispiel zu $(\forall\vdash)|(\vdash\forall)$:
Zeige: $\forall x \neg\neg A(x) \vdash \forall x A(x)$

$$\dfrac{\dfrac{\dfrac{\dfrac{A(a)\vdash A(a)}{\vdash A(a),\neg A(a)}\,(\vdash\neg)}{\neg\neg A(a)\vdash A(a)}\,(\neg\vdash)}{\forall x\neg\neg A(x)\vdash A(a)}\,(\forall\vdash)}{\forall x\neg\neg A(x)\vdash\forall x A(x)}\,(\vdash\forall)$$

Die Reihenfolge der Anwendungen von $(\forall\vdash)$ und $(\vdash\forall)$ läßt sich nicht umkehren, da ansonsten die Eigenparameterbedingung verletzt wird.

(ii) Exemplarisches Beispiel für $(\vdash\wedge)|(\vee\vdash)$:

$$\dfrac{\dfrac{A,X\vdash Y,C \qquad A,X\vdash Y,D}{A,X\vdash Y,C\wedge D}\,(\vdash\wedge) \qquad B,X\vdash Y,C\wedge D}{A\vee B,X\vdash Y,C\wedge D}\,(\vee\vdash)$$

$$\dfrac{\dfrac{\dfrac{A,X\vdash Y,C}{A,X\vdash Y,C,C\wedge D}\,(\mathrm{V}) \qquad \dfrac{B,X\vdash Y,C\wedge D}{B,X\vdash Y,C,C\wedge D}\,(\mathrm{V})}{A\vee B,X\vdash Y,C,C\wedge D}\,(\vee\vdash) \qquad \dfrac{\dfrac{A,X\vdash Y,D}{A,X\vdash Y,D,C\wedge D}\,(\mathrm{V}) \qquad \dfrac{B,X\vdash Y,C\wedge D}{B,X\vdash Y,D,C\wedge D}\,(\mathrm{V})}{A\vee B,X\vdash Y,D,C\wedge D}\,(\vee\vdash)}{A\vee B,X\vdash Y,C\wedge D}\,(\vdash\wedge)$$

Andere Beispiele als Übung. Es gilt übrigens auch die Vertauschbarkeit mit sich selbst, z.B.: $(\vdash\wedge)|(\vdash\wedge)$. □

Bemerkung.
Vertauschbarkeit von $\mathrm{R}_1|\mathrm{R}_2$ ist nicht definiert, wenn die Hauptformel von R_1 eine Nebenformel (statt einer Seitenformel) von R_2 ist.

Beispiel.

$$\dfrac{\dfrac{X\vdash Y,A(a)}{X\vdash Y,\forall xA(x)}\,(\vdash\forall) \qquad \dfrac{X\vdash Y,B(b)}{X\vdash Y,\forall xB(x)}\,(\vdash\forall)}{X\vdash Y,\forall xA(x)\wedge\forall xB(x)}\,(\vdash\wedge)$$

Satz 2.3 (Formen von Ableitungen pränexer Formeln, Gentzen 1934/35)
Sei eine Ableitung von $X\vdash Y$ gegeben. Seien die Formeln in X und Y alle in pränexer Normalform. Dann gibt es eine Ableitung von $X\vdash Y$ der Form:

$$
\left.\begin{array}{c}\bigtriangledown\\ X_1 \vdash Y_1\end{array}\right\}\text{nur junktorenlogische Regeln}
$$
$$
\left.\begin{array}{c}\vdots\\ X \vdash Y\end{array}\right\}\text{nur quantorenlogische Regeln}
$$

Beweis.
Bei jedem Paar $R_1|R_2$, wobei R_1 quantorenlogisch und R_2 junktorenlogisch ist, muß die Hauptformel von R_1 Seitenformel von R_2 sein, da in X und Y kein Quantor im Bereich eines Junktors steht. Also besteht Vertauschbarkeit. Durch Iteration erhält man die beschriebene Form. □

Korollar 2.1
Sei eine Ableitung von $\vdash \exists x A(x)$ gegeben, wobei $A(x)$ junktorenlogisch. Dann gibt es eine Ableitung von $\vdash \exists x A(x)$ der Form:

$$
\begin{array}{rcl}
& \left.\begin{array}{c}\bigtriangledown\\ \vdash A(t_1), \ldots, A(t_n)\end{array}\right\} & \text{junktorenlogisch}\\
(\vdash \exists) & \overline{\vdash \exists x A(x), A(t_2), \ldots, A(t_n)} & \text{für gewisse Terme } t_1, \ldots, t_n\\
(\vdash \exists) & \overline{\vdash \exists x A(x), A(t_3), \ldots, A(t_n)} & \\
& \vdots & \\
(\vdash \exists) & \vdash \exists x A(x) &
\end{array}
$$

Bemerkungen.

(i) Dies ist ein Spezialfall des Herbrandschen Satzes, der von fundamentaler Bedeutung für das automatische Beweisen ist.

(ii) $\forall$-Quantoren lassen sich durch sog. „Skolemisierung“ beseitigen (siehe Kap. 7). Die quantorenlogische Allgemeingültigkeit läßt sich somit auf Termfindung plus Junktorenlogik zurückführen.

3 Vollständigkeit, Hilberttypkalkül

$\mathscr{L}$ sei die betrachtete Sprache.
Wir setzen den Begriff der Wahrheit bei einer Interpretation $\mathcal{I}$ über dem Bereich $\mathcal{U}$ voraus. $\mathcal{I}$ ordnet den Individuenkonstanten Objekte von $\mathcal{U}$ und den Prädikatkonstanten Prädikate (Attribute) über $\mathcal{U}$ zu. Dabei bezieht sich $\mathcal{I}$ auf eine erweiterte Sprache $\mathscr{L}(\mathcal{U})$, die neben den Individuenkonstanten der Ausgangssprache $\mathscr{L}$ noch Namen $\underline{u}$ aller Elemente $u \in \mathcal{U}$ enthält, die durch $\mathcal{I}$ auch entsprechend interpretiert werden, d.h. $\mathcal{I}(\underline{u}) = u$.
Wahrheit wird nur für geschlossene Formeln erklärt. Eine Interpretation, bei der eine Formel A wahr ist, heißt *Modell von A*. Eine Interpretation, bei der alle Formeln von X wahr (falsch) werden, heißt *Modell (Gegenmodell) von X*.
Achtung: Nach unserer Terminologie falsifiziert ein Gegenmodell zu X *alle* Formeln in X. Es kann also sein, daß X weder ein Modell noch ein Gegenmodell besitzt.

Definition 3.1 (Term-Interpretation)
Eine *Term-Interpretation* ist eine Interpretation über dem Bereich $\mathcal{T}$ aller Terme, bei der Terme durch sich selbst interpretiert werden.
Es ist hier keine erweiterte Sprache notwendig: Alle Terme werden als sich selbst bezeichnende Konstanten aufgefaßt.

Beispiel.
$\forall x A(x)$ ist wahr über $\mathcal{U}$ gdw $A(\underline{u})$ ist wahr über $\mathcal{U}$ für jedes $u \in \mathcal{U}$.

Eine Term-Interpretation $\mathcal{I}$ ist ein *Gegenmodell* zur Sequenz $X \vdash Y$, falls es eine Substitutionsinstanz $X' \vdash Y'$ von $X \vdash Y$ gibt, so daß gilt:

$\mathcal{I}$ ist Modell von X' und Gegenmodell von Y'.

Bemerkung.
$X' \vdash Y'$ kann mit $X \vdash Y$ identisch sein. Die Parameter in $X' \vdash Y'$ werden bezüglich $\mathcal{I}$ als Individuenkonstanten aufgefaßt.

Beispiel.
$\mathcal{I}(P) = \{\langle 1, a\rangle\}$ ist ein Gegenmodell von $P(1,a) \vdash P(a,1)$, da in $\langle 1, a\rangle$ das a als Konstante aufgefaßt wird. D.h., a ist (als Term über dem Termuniversum) eine Konstante, die man für a (als Parameter) einsetzt.

Satz 3.1 (Korrektheit von LK)
Falls $X \vdash_{\mathrm{LK}} Y$, dann gilt für jede Interpretation $\mathcal{I}$ über $\mathcal{U}$ und jede geschlossene Substitutionsinstanz $X' \vdash Y'$ von $X \vdash Y$ bezüglich $\mathscr{L}(\mathcal{U})$:

Falls $\mathcal{I}$ Modell von X', dann $\mathcal{I}$ Modell von A für mindestens ein $A \in Y'$.

Abkürzung: Falls $X \vdash_{\mathrm{LK}} Y$, dann $X \vDash Y$.

Beweis.
Induktion über der Struktur einer Ableitung von $X \vdash Y$:

(I) und Junktorenlogik: klar.

($\forall\vdash$): Sei $\mathcal{I}$ Modell von $\{\forall x A'(x)\} \cup X'$. Dann ist $\mathcal{I}$ Modell von X', und für jeden geschlossenen Term t von $\mathscr{L}(\mathcal{U})$ gilt: $\mathcal{I}$ ist Modell von $A'(t)$.
Nach Induktionsvoraussetzung ist $\mathcal{I}$ Modell für ein B aus Y'.

($\vdash\forall$): Sei $\mathcal{I}$ Modell von X'. Nach Induktionsvoraussetzung ist $\mathcal{I}$ dann Modell einer Formel aus $Y' \cup \{A'(t)\}$ für jeden beliebigen geschlossenen Term t aus $\mathscr{L}(\mathcal{U})$. (Da a nicht in X und Y vorkommt, werden X' und Y' durch Substitution von a nicht verändert). Damit ist $\mathcal{I}$ Modell von $\forall x A(x)$, falls es nicht Modell eines Elementes aus Y' ist.

($\exists\vdash$) und ($\vdash\exists$) analog. □

Korollar 3.1
Falls $X \vdash_{\mathrm{LK}} A$ und X und A geschlossen, dann folgt A logisch aus X.

Falls $\vdash_{\mathrm{LK}} A$, dann ist A allgemeingültig (logisch wahr).

Falls $A \vdash_{\mathrm{LK}}$, dann ist A kontradiktorisch (logisch falsch).

Satz 3.2 (Vollständigkeit von LK)
Falls für jede Interpretation $\mathcal{I}$ über $\mathcal{U}$ und für jede geschlossene Substitutionsinstanz $X' \vdash Y'$ von $X \vdash Y$ bezüglich $\mathscr{L}(\mathcal{U})$ gilt:

Falls $\mathcal{I}$ Modell von X', dann $\mathcal{I}$ Modell von A für ein $A \in Y'$,

dann gilt: $X \vdash_{\mathrm{LK}} Y$.

Abkürzung: Falls $X \vDash Y$, dann $X \vdash_{\mathrm{LK}} Y$.

Beweis.
Sei $X \nvdash_{\mathrm{LK}} Y$. Dann gibt es kein geschlossenes Tableau für $\langle X, Y\rangle$. Damit gibt es ein systematisch entwickeltes Tableau für $\langle X, Y\rangle$, das nicht geschlossen ist. Jeder nichtgeschlossene Zweig eines solchen Tableaus bildet ein Gegenmodell zu $X \vdash Y$.
(Für systematisch entwickelte Tableaus und Einzelheiten des Vollständigkeitssatzes siehe Logik I.) □

Beispiel.
$\nvdash_{\mathrm{LK}} \forall x \exists y (Px \rightarrow \neg Py)$

$$\begin{array}{lrcl}
\text{Tableau:} & 1. & \mathrm{f}\, \forall x \exists y (Px \rightarrow \neg Py) & \\
 & 2. & \mathrm{f}\, \exists y (Pa_1 \rightarrow \neg Py) & (1) \\
 & 3. & \mathrm{f}\, Pa_1 \rightarrow \neg Pa_1) & (2) \\
 & 4. & \mathrm{w}\, Pa_1 & (3) \\
 & 5. & \mathrm{f}\, \neg Pa_1 & (3) \\
 & 6. & \mathrm{w}\, Pa_1 & (5) \\
 & 7. & \mathrm{f}\, (Pa_1 \rightarrow \neg Pa_2) & (2) \\
 & 8. & \mathrm{w}\, Pa_1 & (7) \\
 & 9. & \mathrm{f}\, \neg Pa_2 & (7) \\
 & 10. & \mathrm{w}\, Pa_2 & (9) \\
 & & \vdots &
\end{array}$$

Aus diesem unendlichen Zweig ergibt sich als Gegenmodell: $\mathcal{U} = \{a_1, a_2, a_3, \ldots\}$, $\mathcal{I}(P) = \mathcal{U}$.

Satz 3.3 (Schnitt)

Die Schnittregel $\dfrac{X \vdash Y, A \quad A, X' \vdash Y'}{X, X' \vdash Y, Y'}$ (Schnitt)

ist zulässig in LK.

Beweis.
Über Korrektheits- und Vollständigkeitssatz und: Falls $X \vDash Y, A$ und $A, X' \vDash Y'$, dann $X, X' \vDash Y, Y'$. □

Korollar 3.2
LK ist widerspruchsfrei, d.h. $\nvdash_{\mathrm{LK}} A \wedge \neg A$ für jedes A.

Beweis.
Eine Widerspruchsherleitung kann nur so aussehen:

$$\dfrac{\begin{array}{c}\vdots \\ \vdash A\end{array} \qquad \dfrac{\begin{array}{c}\vdots \\ A \vdash\end{array}}{\vdash \neg A}\,(\vdash \neg)}{\vdash A \wedge \neg A}\,(\vdash \wedge)$$

Da Schnitt zulässig ist, gewinnen wir:

$$\dfrac{\begin{array}{c}\vdots \\ \vdash A\end{array} \qquad \begin{array}{c}\vdots \\ A \vdash\end{array}}{\vdash}\,(\text{Schnitt})$$

Die leere Sequenz $\vdash$ ist jedoch nicht ableitbar (da es keine Regel dafür gibt): *Widerspruch.* □

Definition 3.2 (Hilberttypkalkül)
Der Hilberttypkalkül HK hat folgende Axiome und Regeln (vgl. Kleene 1952, S. 82):

Axiome:

$$\begin{cases} A \to (B \to A) \\ (A \to B) \to ((A \to (B \to C)) \to (A \to C)) \end{cases}$$

$$\begin{cases} A \to (B \to A \land B) \\ A \land B \to A \\ A \land B \to B \end{cases}$$

$$\begin{cases} A \to A \lor B \\ B \to A \lor B \\ (A \to C) \to ((B \to C) \to (A \lor B \to C)) \end{cases}$$

$$\begin{cases} (A \to B) \to ((A \to \neg B) \to \neg A) \\ \neg\neg A \to A \end{cases}$$

$$\left\{ \forall x A(x) \to A(t) \right.$$

$$\left\{ A(t) \to \exists x A(x) \right.$$

Regeln:

$$\text{(MP)} \quad \frac{A \quad A \to B}{B}$$

$$\left.\begin{array}{ll} (\forall) & \dfrac{C \to A(a)}{C \to \forall x A(x)} \\[2ex] (\exists) & \dfrac{A(a) \to C}{\exists x A(x) \to C} \end{array}\right\} a \text{ nicht in } C \text{ und in keiner Annahme}$$

Ableitungen werden nicht in Baumform, sondern linear geschrieben.

Eine Ableitung in HK von A aus der Menge von Annahmen X ist eine Folge von Formeln, in der jede Formel entweder zu X gehört oder ein Axiom ist, oder aus vorherigen Formeln durch Anwendung einer Regel hervorgeht.

$X \vdash_{\mathrm{HK}} A$ bedeutet: A ist aus X ableitbar.

Beispiele.
(a) $\vdash_{\mathrm{HK}} A \to A$

1. Axiom	1.	$A \to (A \to A)$
2. Axiom	2.	$(A \to \underbrace{(A \to A)}_{B}) \to ((A \to (\underbrace{(A \to A)}_{B} \to \underbrace{A}_{C})) \to (A \to \underbrace{A}_{C}))$
(MP) auf 1. 2.	3.	$(A \to ((A \to A) \to A)) \to (A \to A)$
Axiom	4.	$A \to (\underbrace{(A \to A)}_{B} \to A)$
(MP) auf 3. 4.	5.	$A \to A$

(b) $\forall x A(x) \vdash_{\mathrm{HK}} \exists x A(x)$

Axiom	1.	$A(a) \to \exists x A(x)$
Axiom	2.	$\forall x A(x) \to A(a)$
Axiom	3.	$(\underbrace{\forall x A(x)}_{A} \to \underbrace{A(a)}_{B}) \to ((\underbrace{\forall x A(x)}_{A} \to (\underbrace{A(a)}_{B} \to \underbrace{\exists x A(x)}_{C})) \to (\underbrace{\forall x A(x)}_{A} \to \underbrace{\exists x A(x)}_{C}))$
(MP) auf 2. 3.	4.	$(\forall x A(x) \to (A(a) \to \exists x A(x))) \to (\forall x A(x) \to \exists x A(x))$
Axiom	5.	$(\underbrace{A(a) \to \exists x A(x)}_{A}) \to (\underbrace{\forall x A(x)}_{B} \to (\underbrace{A(a) \to \exists x A(x)}_{A}))$
(MP) auf 1. 5.	6.	$\forall x A(x) \to (A(a) \to \exists x A(x))$
(MP) auf 4. 6.	7.	$\forall x A(x) \to \exists x A(x)$
Annahme	8.	$\forall x A(x)$
(MP) auf 7. 8.	9.	$\exists x A(x)$

Satz 3.4
Falls $X \vdash_{\mathrm{HK}} A$, dann $X \vdash_{\mathrm{LK}} A$.

Beweis.

(i) Für jedes Axiom C von HK gilt: $\vdash_{\mathrm{LK}} C$:
Beispiele:

(a) $$\dfrac{\dfrac{\dfrac{}{A, B \vdash A}\,(\mathrm{I})}{A \vdash B \to A}\,(\vdash\to)}{\vdash A \to (B \to A)}\,(\vdash\to)$$

(b) $$\dfrac{\dfrac{\dfrac{}{A \vdash A, B}\,(\mathrm{I})}{A \vdash A \lor B}\,(\vdash\lor)}{\vdash A \to (A \lor B)}\,(\vdash\to)$$

(c) $$\dfrac{\dfrac{\dfrac{\dfrac{}{A \vdash A}\,(\mathrm{I})}{\vdash A, \neg A}\,(\vdash\neg)}{\neg\neg A \vdash A}\,(\neg\vdash)}{\vdash \neg\neg A \to A}\,(\vdash\to)$$

(ii) Eine Annahme C in einer HK-Ableitung wird durch ein LK-Axiom $C, X \vdash C$ für geeignetes X ersetzt.

(iii) (MP) ist in LK herleitbar:

$$\dfrac{X \vdash A \to B \qquad \dfrac{X \vdash A \qquad \dfrac{}{X, B \vdash B}\,(\mathrm{I})}{X, A \to B \vdash B}\,(\to\vdash)}{X \vdash B}\,(\text{Schnitt})$$

($\forall$) ist in LK herleitbar:

$$\dfrac{\dfrac{\dfrac{X \vdash C \to A(a) \qquad \dfrac{\dfrac{}{X, C \vdash C}\,(\mathrm{I}) \qquad \dfrac{}{X, A(a) \vdash A(a)}\,(\mathrm{I})}{X, C, C \to A(a) \vdash A(a)}\,(\to\vdash)}{X, C \vdash A(a)}\,(\text{Schnitt})}{X, C \vdash \forall x A(x)}\,(\vdash\forall)}{X \vdash C \to \forall x A(x)}\,(\vdash\to)$$

(∃) analog. □

Bemerkung.
Hier sieht man, wie wichtig es ist, daß die Schnittregel in LK zulässig ist.

Lemma 3.1 (Deduktionstheorem)
Falls $X, A \vdash_{\mathrm{HK}} B$, dann $X \vdash_{\mathrm{HK}} A \to B$.

Beweis.
Sei eine Ableitung von B aus $X \cup \{A\}$ gegeben. Wir führen Induktion über dem Aufbau dieser Ableitung.

(i) B ist in X:

$$\begin{array}{ll} B & \text{(Annahme)} \\ B \to (A \to B) & \text{(Axiom)} \\ A \to B & \text{(MP)} \end{array}$$

(ii) B ist A:

$$\begin{array}{ll} \vdots & \\ A \to A & \text{(siehe Beispiel oben)} \end{array}$$

(iii) B ist Axiom:

$$\begin{array}{ll} B & \text{(Axiom)} \\ B \to (A \to B) & \text{(Axiom)} \\ A \to B & \text{(MP)} \end{array}$$

(iv) (MP) sei im letzten Schritt angewendet:

$$\left.\begin{array}{lc} & \vdots \\ n. & C \\ & \vdots \\ m. & C \to B \\ & \vdots \\ \text{(MP) auf } n, m & B \end{array}\right\} \Longrightarrow \begin{array}{ll} \left.\begin{array}{l} \vdots \\ A \to C \\ \vdots \\ A \to (C \to B) \end{array}\right\} \text{Induktionsvoraussetzung} & \\ (A \to C) \to ((A \to (C \to B)) \to (A \to B)) & \text{(Ax.)} \\ (A \to (C \to B)) \to (A \to B) & \text{(MP)} \\ A \to B & \text{(MP)} \end{array}$$

(v) (∀) sei im letzten Schritt angewendet:

$$\left.\begin{array}{c} \vdots \\ C \to D(a) \\ \\ \vdots \\ \\ \underbrace{C \to \forall x D(x)}_{B} \end{array}\right\} \Longrightarrow \begin{array}{ll} \left.\begin{array}{l} \vdots \\ A \to (C \to D(a)) \end{array}\right\} & \text{Induktionsvoraussetzung} \\ \quad\vdots & \\ (A \wedge C) \to D(a) & \text{Übungsaufgabe} \\ (A \wedge C) \to \forall x D(x) & (\forall) \\ \quad\vdots & \\ A \to (C \to \forall x D(x)) & \text{Übungsaufgabe} \end{array}$$

Parameterbedingung erfüllt

(vi) $(\exists)$ sei im letzten Schritt angewendet:

$$\left.\begin{array}{c} \vdots \\ D(a) \to C \\ \\ \vdots \\ \\ \underbrace{\exists x D(x) \to C}_{B} \end{array}\right\} \Longrightarrow \begin{array}{ll} \left.\begin{array}{l} \vdots \\ A \to (D(a) \to C) \end{array}\right\} & \text{Induktionsvoraussetzung} \\ \quad\vdots & \\ D(a) \to (A \to C) & \text{Übungsaufgabe} \\ \exists x D(x) \to (A \to C) & (\exists) \\ \quad\vdots & \\ A \to (\exists x D(x) \to C) & \text{Übungsaufgabe} \end{array}$$

Parameterbedingung erfüllt □

Satz 3.5
Falls $X \vdash_{\mathrm{LK}} A_1, \ldots, A_n$, dann $X \vdash_{\mathrm{HK}} A_1 \vee \ldots \vee A_n$.

Beweis.
Induktion über dem Aufbau von LK-Ableitungen.
Beispiele:

(I): zu zeigen: $A, X \vdash_{\mathrm{HK}} \bigvee Y \vee A$
($\bigvee Y$ steht für die Disjunktion der Elemente von Y).
Folgt mit $A \to \bigvee Y \vee A$ (Axiom) und (MP).

$(\neg\vdash)$: nach Induktionsvoraussetzung gilt:
$X \vdash_{\mathrm{HK}} \bigvee Y \vee A$.
Daraus:
$\neg A, X \vdash_{\mathrm{HK}} \bigvee Y$ (Übungsaufgabe)

$(\exists\vdash)$: nach Induktionsvoraussetzung gilt:
$A(a), X \vdash_{\mathrm{HK}} \bigvee Y$.

Daraus:

$X \vdash_{\mathrm{HK}} A(a) \to \bigvee Y$	(Deduktionstheorem)
$X \vdash_{\mathrm{HK}} \exists x A(x) \to \bigvee Y$	($\exists$), Parameterbedingung erfüllt
$Y, \exists x A(x) \vdash_{\mathrm{HK}} \exists x A(x)$	(Axiom)
$Y, \exists x A(x) \vdash_{\mathrm{HK}} \bigvee Y$	(MP)

usw. $\square$

3.1 Anhang: Schütte-Tait-Kalkül

Betrachten LK ohne $\to$.
Bringen alles auf eine Seite, z.B. nach links, d.h., $X, \neg Y \vdash$ statt $X \vdash Y$.

$(\wedge\vdash)$ wird zu $\quad \dfrac{A, B, X \vdash}{A \wedge B, X \vdash}$

$(\vdash\wedge)$ wird zu $\quad \dfrac{\neg A, X \vdash \quad \neg B, X \vdash}{\neg(A \wedge B), X \vdash}$

Mit der Abkürzung $\neg(A \wedge B) = \neg A \vee \neg B$

wird $(\vdash\wedge)$ Spezialfall von $\quad \dfrac{A, X \vdash \quad B, X \vdash}{A \vee B, X \vdash}\ (\vee\vdash)$

Generell wird jetzt $\neg$ nach innen gezogen, d.h., $\neg$ steht nur vor atomarer Formel:

$$\begin{aligned}
\neg\neg A &:= A \\
\neg(A \wedge B) &:= \neg A \vee \neg B \\
\neg(A \vee B) &:= \neg A \wedge \neg B \\
\neg\forall x A(x) &:= \exists x \neg A(x) \\
\neg\exists x A(x) &:= \forall x \neg A(x)
\end{aligned}$$

Regeln des Schütte-Tait-Kalküls:

(I) $\quad A, \neg A, X \quad$ (A atomar)

$(\wedge) \quad \dfrac{A, B, X}{A \wedge B, X} \qquad (\vee) \quad \dfrac{A, X \quad B, X}{A \vee B, X}$

$(\forall) \quad \dfrac{A(t), X}{\forall x A(x), X} \qquad (\exists) \quad \dfrac{A(a), X}{\exists x A(x), X} \qquad$ (a nicht in X)

Duale Variante:

$\left.\begin{array}{ccc} \wedge & | & \vee \\ \forall & | & \exists \end{array}\right\}$ vertauscht

Schnittregel: $\quad \dfrac{X, A \quad \neg A, Y}{X, Y}$

4 Natürliches Schließen

Der Kalkül geht auf Jaśkowski und Gentzen zurück.

Motivation.
1. Grundidee: keine Axiome, sondern nur Annahmen
2. Grundidee: Abhängigkeit von Annahmen kann gelöscht werden

$$\dfrac{\begin{matrix} A \\ \vdots \\ B \end{matrix}}{A \to B} \qquad \begin{matrix} B \text{ noch von } A \text{ abhängig} \\ A \to B \text{ nicht mehr von } A \text{ abhängig} \end{matrix}$$

Regeln für NK

	Einführung		Beseitigung
$(\wedge\,\mathrm{I})$	$\dfrac{A \quad B}{A \wedge B}$	$(\wedge\,\mathrm{E})$	$\dfrac{A \wedge B}{A} \qquad \dfrac{A \wedge B}{B}$
$(\vee\,\mathrm{I})$	$\dfrac{A}{A \vee B} \qquad \dfrac{B}{A \vee B}$	$(\vee\,\mathrm{E})$	$\dfrac{A \vee B \quad \begin{matrix} A \\ \vdots \\ C \end{matrix} \quad \begin{matrix} B \\ \vdots \\ C \end{matrix}}{C}$
$(\to\mathrm{I})$	$\dfrac{\begin{matrix} A \\ \vdots \\ B \end{matrix}}{A \to B}$	$(\to\mathrm{E})$	$\dfrac{A \to B \quad A}{B}$
		$(\bot)_c$	$\dfrac{\begin{matrix} \neg A \\ \vdots \\ \bot \end{matrix}}{A}$
$(\forall\,\mathrm{I})$	$\dfrac{A(a)}{\forall x A(x)}$ a in keiner Annahme, von der $A(a)$ abhängt	$(\forall\,\mathrm{E})$	$\dfrac{\forall x A(x)}{A(t)}$
$(\exists\,\mathrm{I})$	$\dfrac{A(t)}{\exists x A(x)}$	$(\exists\,\mathrm{E})$	$\dfrac{\exists x A(x) \quad \begin{matrix} A(a) \\ \vdots \\ C \end{matrix}}{C}$ a nicht in C und in keiner Annahme außer $A(a)$, von der C abhängt

NI (intuitionistisch) hat $\dfrac{\perp}{A}(\perp)$ statt $(\perp)_c$.
In NM (minimal) fehlt $(\perp)_c$ ersatzlos.

Zunächst einige
BEISPIELE.

(a) $A\to(B\to C)\vdash_{\mathrm{NK}}(A\to B)\to(A\to C)$

$$\dfrac{\dfrac{\dfrac{\dfrac{A\to(B\to C)\quad A^{(1)}}{B\to C}(\to\mathrm{E})\qquad \dfrac{A\to B^{(2)}\quad A^{(1)}}{B}(\to\mathrm{E})}{C}(\to\mathrm{E})}{A\to C}(\to\mathrm{I})_{(1)}}{(A\to B)\to(A\to C)}(\to\mathrm{I})_{(2)}$$

[(1)] und [(2)] markieren die zusätzlichen Annahmen, die in der Herleitung gelöscht werden.

(b) Gebundene Umbenennung: $\exists xPx\vdash_{\mathrm{NK}}\exists yPy$

$$\dfrac{\exists xPx\qquad \dfrac{Pa^{(1)}}{\exists yPy}(\exists\mathrm{I})}{\exists yPy}(\exists\mathrm{E})_{(1)}$$

Die Endformel hängt nicht mehr von der Annahme Pa ab.

(c) Assoziativgesetz für $\lor$: $A\lor(B\lor C)\vdash_{\mathrm{NK}}(A\lor B)\lor C$

$$\dfrac{A\lor(B\lor C)\qquad \dfrac{\dfrac{A^{(2)}}{A\lor B}(\lor\mathrm{I})}{(A\lor B)\lor C}(\lor\mathrm{I})\qquad \dfrac{B\lor C^{(2)}\qquad \dfrac{\dfrac{B^{(1)}}{A\lor B}(\lor\mathrm{I})}{(A\lor B)\lor C}(\lor\mathrm{I})\qquad \dfrac{C^{(1)}}{(A\lor B)\lor C}(\lor\mathrm{I})}{(A\lor B)\lor C}(\lor\mathrm{E})_{(1)}}{(A\lor B)\lor C}(\lor\mathrm{E})_{(2)}$$

[(1)] und [(2)] markieren die zusätzlichen Annahmen, die in der Herleitung im vorletzten und letzten Schritt gelöscht werden.

(d) Assoziativgesetz für $\land$: $A\land(B\land C)\vdash_{\mathrm{NK}}(A\land B)\land C$

$$\dfrac{\dfrac{\dfrac{A\land(B\land C)}{A}(\land\mathrm{E})\qquad \dfrac{\dfrac{A\land(B\land C)}{B\land C}(\land\mathrm{E})}{B}(\land\mathrm{E})}{A\land B}(\land\mathrm{I})\qquad \dfrac{\dfrac{A\land(B\land C)}{B\land C}(\land\mathrm{E})}{C}(\land\mathrm{E})}{(A\land B)\land C}(\land\mathrm{I})$$

(e) $\neg\neg A \vdash_{\mathrm{NK}} A$

Hierbei $\neg A := A \to \bot$. Die Behauptung bedeutet also: $(A \to \bot) \to \bot \vdash_{\mathrm{NK}} A$

$$\dfrac{\dfrac{(A\to\bot)\to\bot \qquad A\to\bot^{(1)}}{\bot}\,(\to\mathrm{E})}{A}\,(\bot)_c\ {\scriptstyle(1)}$$

$(\bot)_c$ entspricht somit der Beseitigung der doppelten Negation.

(f) $\vdash_{\mathrm{NK}} A \lor \neg A$

Wir führen die Negation zum Widerspruch (klassische *reductio ad absurdum*):

$$\dfrac{\dfrac{\dfrac{\dfrac{\neg(A\lor\neg A)^{(3)} \qquad \dfrac{A^{(1)}}{A\lor\neg A}\,(\lor\mathrm{I})}{\bot}\,(\to\mathrm{E})}{\neg A}\,(\to\mathrm{I})\ {\scriptstyle(1)} \qquad \dfrac{\dfrac{\neg(A\lor\neg A)^{(3)} \qquad \dfrac{\neg A^{(2)}}{A\lor\neg A}\,(\lor\mathrm{I})}{\bot}\,(\to\mathrm{E})}{A}\,(\bot)_c\ {\scriptstyle(2)}}{\bot}\,(\to\mathrm{E})}{A\lor\neg A}\,(\bot)_c\ {\scriptstyle(3)}$$

Jetzt die formelle Definition einer Ableitung. Wir betrachten eine Sprache mit $\bot$ als Grundzeichen. $\neg A$ ist definiert als $A \to \bot$.

Definition 4.1

- Eine Ableitung $\mathcal{D}$ ist ein Paar $\langle \mathcal{T}, f\rangle$, wobei $\mathcal{T}$ *Formelbaum* und f *Löschungsfunktion* auf $\mathcal{T}$ ist.
- Eine *Löschungsfunktion* auf $\mathcal{T}$ ist dabei eine partielle Funktion von der Menge der Blätter von $\mathcal{T}$ in die Knoten von $\mathcal{T}$ außer der Wurzel (Blätter sind Grenzfälle von Knoten).
- Eine *Teilableitung* einer Ableitung ist ein Teilbaum der Ableitung, wobei die Löschungsfunktion auf die Blätter des Teilbaums, deren Wert *nicht* die Wurzel des Teilbaums ist, eingeschränkt wird.
- Die Blätter einer Ableitung heißen *Annahmen*, die Wurzel heißt *Endformel*.
- Annahmen, für welche die Löschungsfunktion nicht definiert ist, heißen *offene Annahmen*, sonst *geschlossene Annahmen*.
- Metasprachliche Zeichen für Ableitungen: $\mathcal{D}, \mathcal{D}_1, \mathcal{D}', \ldots$

- *Notation von Löschungsfunktionen durch Ziffern*:

$$\dfrac{\begin{array}{c}A^{(n)}\ldots A^{(n)}\ldots A^{(n)}\\ \vdots\\ C\end{array} \qquad\qquad \begin{array}{c}B^{(n)}\ldots B^{(n)}\\ \vdots\\ D\end{array}}{E}\,{\scriptstyle (n)}$$

  Dies bedeutet: Den Blättern A wird C, den Blättern B wird D zugeordnet.

- *Schematische Darstellung von Regeln*:
  Das Beispiel:

$$\dfrac{C_1 \qquad \begin{array}{c}A\\ \vdots\\ C_2\end{array} \qquad\qquad \begin{array}{c}B_1 \qquad B_2\\ \vdots\\ C_3\end{array}}{D}$$

  bedeutet: Aus Ableitungen mit den Endformeln C_1, C_2, C_3 darf eine neue Ableitung mit der Endformel D erzeugt werden. Die Löschungsfunktion der neuen Ableitung enthält die der alten Ableitungen; ferner darf beliebig vielen Annahmen A das C_2 und beliebig vielen Annahmen B_1 und B_2 das C_3 zugeordnet werden.

- *Allgemeine Form einer Regel*:

$$\dfrac{\begin{array}{c}A_{11}\ldots A_{1k_1}\\ \vdots\\ B_1\end{array} \quad \ldots\ldots\ldots\ldots \quad \begin{array}{c}A_{n1}\ldots A_{nk_n}\\ \vdots\\ B_n\end{array}}{C}$$

- *Schematische Darstellung von Ableitungen*:

  (a) $\begin{array}{c}\mathcal{D}\\ B\end{array}$ „$\mathcal{D}$ endet mit B“

  (b) $\begin{array}{c}A_1,\ldots,A_n\\ \mathcal{D}\\ B\end{array}$ „$\mathcal{D}$ endet mit B, und $A_1,\ldots,A_n$ können offene Annahmen in $\mathcal{D}$ sein“

  (c) $\dfrac{\ldots \qquad \begin{array}{c}[A_1,\ldots,A_n]\\ \mathcal{D}\end{array} \qquad \ldots}{C}$ „In $\mathcal{D}$ können $A_1,\ldots,A_n$ als offene Annahmen vorkommen, die jedoch beim Übergang zu C gelöscht werden.“

(d) $$\begin{array}{c} \mathcal{D}_i \\ A_1, \ldots, A_i, \ldots, A_n \\ \mathcal{D} \end{array}$$ „Die offenen Annahmen A_i in $\mathcal{D}$ werden durch $\mathcal{D}_i$ ersetzt, wobei $\mathcal{D}_i$ mit A_i endet.“
(diese Notation bezieht sich auf diejenigen A_i, die im Kontext gemeint sind – nicht notwendigerweise auf alle vorkommenden A_i)

Definition 4.2

- Die Regeln für NK sind oben (S. 21) angegeben.
- NM hat dieselben Regeln wie NK ohne Regel für $\perp$.
- NI hat $\dfrac{\perp}{A}(\perp)$ als Regel für $\perp$.

Bemerkung.
$(\perp)$ ist ein Spezialfall von $(\perp)_c$.

- *Hauptprämisse* bei Eliminationsregeln ist diejenige Prämisse, in der das eliminierte logische Zeichen vorkommt. *Nebenprämissen* sind die anderen Prämissen (kommen bei $(\vee\,\mathrm{E})$, $(\rightarrow\mathrm{E})$ und $(\exists\,\mathrm{E})$ vor).
- $\mathcal{D}$ ist eine *Ableitung von A aus X*, falls A Endformel von $\mathcal{D}$ ist und jede offene Annahme aus $\mathcal{D}$ in X vorkommt.
- $\mathcal{D}$ ist eine *von X abhängige Ableitung von A*, falls $\mathcal{D}$ Ableitung von A aus X ist und jede Formel in X als offene Annahme in $\mathcal{D}$ vorkommt.
- A ist aus X *ableitbar*, falls es eine Ableitung von A aus X gibt.
  Notation: $X \vdash_{\mathrm{NK}} A$, bzw. $X \vdash_{\mathrm{NI}} A$, bzw. $X \vdash_{\mathrm{NM}} A$.

Theorem 4.1

$$\begin{array}{lll} X \vdash_{\mathrm{NK}} A & \text{gdw} & X \vdash_{\mathrm{HK}} A \\ & \text{gdw} & X \vdash_{\mathrm{LK}} A \end{array}$$

Beweis zur Übung.

Lemma 4.1

Die Stärke von NI wird nicht eingeschränkt, wenn man bei $(\perp)$ annimmt, daß A atomar ist. Dasselbe gilt für NK und $(\perp)_c$ für Formeln ohne $\vee$ und $\exists$.

Beweis zur Übung.

Lemma 4.2

Gegeben sei eine unendliche echte Teilmenge $\mathcal{P}$ der Menge der Parameter.
Jede Ableitung $\mathcal{D}$ von A aus X läßt sich in eine Ableitung $\mathcal{D}'$ von A aus X umformen, so daß gilt:

(i) Alle Eigenparameter von $\mathcal{D}'$ sind in $\mathcal{P}$.

(ii) Jeder Eigenparameter in $\mathcal{D}'$ ist Eigenparameter einer *einzigen* Anwendung von $(\forall\,\mathrm{I})$ oder $(\exists\,\mathrm{E})$.

(iii) Der Eigenparameter einer Anwendung von $(\forall\,\mathrm{I})$ kommt nur über dieser Anwendung vor.

(iv) Der Eigenparameter einer Anwendung von $(\exists\,\mathrm{E})$ kommt nur über den Nebenprämissen dieser Anwendung vor.

(v) Bis auf Umbenennung von Parametern unterscheiden sich $\mathcal{D}$ und $\mathcal{D}'$ nicht.

Beweis.
Vgl. Lemma 2.1 und Bemerkung dazu. $\square$

Korollar 4.1

$\begin{matrix}\mathcal{D}\\ A\end{matrix}$ erfülle die Bedingungen aus Lemma 4.2. Der Term t enthalte keine Parameter aus $\mathcal{P}$. Der Parameter a komme in A vor.

Dann ist $\begin{matrix}\mathcal{D}[{}^{t}\!/\!_{a}]\\ A[{}^{t}\!/\!_{a}]\end{matrix}$ eine Ableitung, wobei $\mathcal{D}[{}^{t}\!/\!_{a}]$ aus $\mathcal{D}$ durch Ersetzung *jedes* Vorkommens von a durch t entstehe.

5 Normalisierung für NK

Wir betrachten das System NK *ohne* $\vee, \exists$ (vgl. Aufgaben). Ferner habe $(\bot)_c$ atomare Konklusionen (vgl. Aufgabe). Parameter seien separiert (Lemma 4.2).

Definition 5.1
Ein Formelvorkommen einer Ableitung heißt *maximal*, wenn es Konklusion der Anwendung einer Einführungsregel und zugleich Hauptprämisse der Anwendung einer Beseitigungsregel ist.
Eine Ableitung heißt *normal*, wenn sie kein maximales Formelvorkommen enthält.

Definition 5.2
Wir definieren Umformungen von Ableitungen:

$$\text{(i)} \quad \left.\begin{array}{c} \mathcal{D}_1 \qquad \mathcal{D}_2 \\ \dfrac{\dfrac{A_1 \qquad A_2}{A_1 \wedge A_2}}{A_i} \end{array}\right\} \quad \rightsquigarrow \quad \begin{array}{c} \mathcal{D}_i \\ A_i \end{array} \qquad i \in \{1,2\}$$

$$\text{(ii)} \quad \left.\begin{array}{c} \begin{array}{cc} \begin{array}{c} [A] \\ \mathcal{D} \end{array} & \\ & \mathcal{D}_1 \end{array} \\ \dfrac{\dfrac{B}{A \to B} \qquad A}{B} \end{array}\right\} \quad \rightsquigarrow \quad \begin{array}{c} \mathcal{D}_1 \\ A \\ \mathcal{D} \\ B \end{array}$$

$$\text{(iii)} \quad \left.\begin{array}{c} \mathcal{D} \\ \dfrac{\dfrac{A(a)}{\forall x A(x)}}{A(t)} \end{array}\right\} \quad \rightsquigarrow \quad \begin{array}{c} \mathcal{D}[{}^t\!/\!_a] \\ A(t) \end{array}$$

$\mathcal{D} \rhd_1 \mathcal{D}'$ („$\mathcal{D}$ kontrahiert zu $\mathcal{D}'$“) falls $\mathcal{D}'$ aus $\mathcal{D}$ durch Ausführung eines Kontraktionsschrittes an einer Teilableitung von $\mathcal{D}$ sowie durch Umformung zur Herstellung der Parameterseparatiertheit (vgl. Lemma 4.2) hervorgeht.

$\mathcal{D} \rhd \mathcal{D}'$ („$\mathcal{D}$ reduziert zu $\mathcal{D}'$“) falls $\mathcal{D} = \mathcal{D}_1 \rhd_1 \ldots \rhd_1 \mathcal{D}_n = \mathcal{D}'$ (für $n \geq 1$).

Lemma 5.1
Ist $\mathcal{D}$ eine Ableitung von A aus X und $\mathcal{D} \rhd \mathcal{D}'$, dann ist $\mathcal{D}'$ ebenfalls eine Ableitung von A aus X.

Beweis.

(i) Annahmen gehen höchstens verloren; es kommen keine Annahmen hinzu.

(ii) Wegen Separationseigenschaft enthält t keine Eigenparameter von $\mathcal{D}$. □

Theorem 5.1 (Normalisierungssatz)
Zu jeder NK-Ableitung $\mathcal{D}$ gibt es ein $\mathcal{D}'$ mit $\mathcal{D} \rhd \mathcal{D}'$, so daß $\mathcal{D}'$ normal.
(D.h. jede Ableitung in NK hat eine Normalform.)

BEWEIS.

- Induktion über Anzahl $\langle g(\mathcal{D}), n(\mathcal{D})\rangle$, wobei
  $g(\mathcal{D})$ = größter Grad einer Maximalformel in $\mathcal{D}$
  $n(\mathcal{D})$ = Anzahl der Maximalformeln vom Grad $g(\mathcal{D})$ in $\mathcal{D}$.
  (Der Grad einer Formel ist die Anzahl der Vorkommen an logischen Zeichen.)

- Betrachte Teilableitung:

$$\mathcal{D}'' \left\{ \begin{array}{c} \dfrac{\mathcal{D}'}{A}\,\text{I-Regel} \quad \vdots \\ \hline B \end{array} \right. \text{E-Regel}$$

  von $\mathcal{D}$, so daß A Maximalformel von größtem Grad und oberhalb von B keine Maximalformel von größtem Grad ist.

- Reduziere $\mathcal{D}$ durch Kontraktion von $\mathcal{D}''$. Das Ergebnis sei $\mathcal{D}_1$. D.h. $\mathcal{D} \rhd_1 \mathcal{D}_1$.

- Die Maximalformel A verschwindet, und möglicherweise neu entstehende Maximalformeln haben einen kleineren Grad als A.
  D.h. es gilt: Entweder $g(\mathcal{D}_1) < g(\mathcal{D})$, oder $g(\mathcal{D}_1) = g(\mathcal{D})$, aber $n(\mathcal{D}_1) < n(\mathcal{D})$. □

BEMERKUNG.
Wir haben die sog. *schwache Normalisierung* gezeigt. D.h. wir haben ein spezifisches Reduktionsverfahren angegeben, das zu einer Normalform führt.
Es gilt auch die *starke Normalisierung*, wonach jedes *beliebige* Reduktionsverfahren zu einer Normalform führt.

Definition 5.3
Ein *Zweig* zu einer Deduktion geht von einem Blatt zu einer Endformel. Ein *Pfad* ist ein maximales Anfangsstück eines Fadens, das mit der ersten Nebenprämisse einer Anwendung von (→E) endet:

$$\frac{A \to B \quad \begin{array}{c}\vdots \\ A\end{array}}{B}$$

Ein Pfad hat Ordnung 0, wenn er mit der Endformel endet. Er hat Ordnung $n+1$, wenn er neben einer Formel endet, die zu einem Pfad der Ordnung n gehört.

Lemma 5.2
Sei ein Pfad $A_1, \ldots, A_n$ in einer normalen Ableitung gegeben.
Dann gibt es ein A_i $(1 \leq i \leq n)$, so daß gilt:

(i) Alle A_j mit $j < i$ sind Hauptprämissen von Beseitigungsregeln (E-Regeln).

(ii) A_i ist $\bot$ und Prämisse von $(\bot)_c$ oder A_i ist Prämisse einer Einführungsregel (I-Regel), falls $i \neq n$.

(iii) Alle A_j mit $i < j < n$ sind Prämissen von I-Regeln.

Beweis.
Hauptprämissen von E-Regeln müssen vor Prämissen von I-Regeln stehen, sonst kann die Ableitung nicht normal sein. Sie müssen auch vor Prämissen von $(\bot)_c$ stehen, da die Konklusion von $(\bot)_c$ atomar ist. Prämissen von $(\bot)_c$ müssen vor Prämissen von I-Regeln stehen, da $\bot$ nicht eine Konklusion einer I-Regel sein kann. Da $(\bot)_c$ atomare Konklusion hat, also auch nicht $\bot$, kann $(\bot)_c$ nur einmal vorkommen. Nebenprämissen von $(\rightarrow \mathrm{E})$ brauchen nicht berücksichtigt zu werden, da damit der Pfad endet.

Schema eines Pfads:

$$\begin{array}{l}
\left.\begin{array}{c} A_1 \\ A_2 \\ \vdots \\ A_i \end{array}\right\}\text{E-Regeln} \\
\underline{\hphantom{A_{i+1}A}}\ (\bot)_c \text{ oder I-Regel} \\
\left.\begin{array}{c} A_{i+1} \\ \vdots \\ A_n \end{array}\right\}\text{I-Regeln}
\end{array}$$

□

Theorem 5.2 (Teilformelprinzip)
Jede Formel in einer normalen Ableitung ist Teilformel der Endformel oder einer offenen Annahme mit Ausnahme von Annahmen $\neg A$, die bei einer Anwendung von $(\bot)_c$ gelöscht werden und Formeln $\bot$, die unmittelbar darunter stehen.

Beweis.
Wir beweisen die Behauptung für beliebige Pfade $A_1, \ldots, A_n$ durch Induktion über deren Ordnung.

(i) A_n ist entweder Endformel oder ist Teilformel von $A_n \rightarrow B$, das links daneben steht, und zu einem Pfad kleinerer Ordnung gehört. A_n erfüllt damit auf jeden Fall die Bedingung. Also erfüllen alle A_j $(i < j < n)$ ebenfalls die Bedingung (A_i ist diejenige Formel, die gemäß Lemma 5.2 den Pfad in einen Beseitigungs- und einen Einführungsabschnitt teilt).

(ii) Falls A_1 nicht bei einer Anwendung von $(\bot)_c$ gelöscht wird, dann ist A_1 offene Annahme oder wird im selben Pfad oder einem Pfad kleinerer Ordnung durch $(\rightarrow \mathrm{I})$ gelöscht.
Also erfüllt A_1 und damit alle A_j $(1 < j \leq i)$ die Bedingung.

(iii) Falls A_1 bei einer Anwendung von $(\bot)_c$ gelöscht wird, trifft einer der folgenden drei Unterfälle zu.

(a) A_1 ist Prämisse einer I-Regel. Dann ist (i) anwendbar.

(b) $A_1 \equiv A_n$. Dann ist (i) ebenfalls anwendbar.

(c) Es verbleibt der Fall $\dfrac{\neg B \quad B}{\perp}$ wobei $\neg B \equiv A_1$. Das ist genau die im Theorem formulierte Ausnahme. □

Beispiel.
$((A \to B) \to A) \to A$ (Peircesche Formel)

$$
\begin{prooftree}
\AxiomC{$\neg A^{\,(2)}$}
\AxiomC{$(A \to B) \to A^{\,(3)}$}
\AxiomC{$\neg A^{\,(2)}$}
\AxiomC{$A^{\,(1)}$}
\RightLabel{$\stackrel{\text{(Ordnung 3)}}{(\to \mathrm{E})}$}
\BinaryInfC{$\perp$}
\RightLabel{$(\perp)_c$}
\UnaryInfC{B}
\RightLabel{$(\to \mathrm{I})\ (1)$}
\UnaryInfC{$A \to B$}
\RightLabel{$\stackrel{\text{(Ordnung 2)}}{(\to \mathrm{E})}$}
\BinaryInfC{A}
\RightLabel{$\stackrel{\text{(Ordnung 1)}}{(\to \mathrm{E})}$}
\BinaryInfC{$\perp$}
\RightLabel{$(\perp)_c\ (2)$}
\UnaryInfC{A}
\RightLabel{$(\to \mathrm{I})\ (3)$ (Ordnung 0)}
\UnaryInfC{$((A \to B) \to A) \to A$}
\end{prooftree}
$$

„(Ordnung n)“ bedeutet, daß ein Pfad der Ordnung n dort endet.

Bemerkungen.

- Im NK haben wir also kein volles Teilformelprinzip; anders als in LK.
- Natürliches Schließen ist eher auf intuitionistische Logik zugeschnitten.

6 Junktorenlogische Resolution

Definition 6.1
Eine *Klausel* ist eine Sequenz $A_1, \ldots, A_n \vdash B_1, \ldots, B_m$, wobei $A_1, \ldots, A_n, B_1, \ldots, B_m$ Atome, d.h. atomare Formeln sind.

Bemerkung.
Die Sequenz steht für die Disjunktion $\neg A_1 \vee \ldots \vee \neg A_n \vee B_1 \vee \ldots \vee B_m$ von Literalen.

Definition 6.2
Die *junktorenlogische Resolutionsregel* lautet:

$$\frac{X_1 \vdash Y_1, A \qquad A, X_2 \vdash Y_2}{X_1, X_2 \vdash Y_1, Y_2} \, (\mathcal{R})$$

Die Konklusion heißt *Resolvente* der Prämissen.

Satz 6.1
$(\mathcal{R})$ ist semantisch korrekt.

Definition 6.3
Der junktorenlogische Resolutionskalkül erlaubt die Ableitung von Klauseln aus Klauseln (als Annahmen) gemäß der Regel $(\mathcal{R})$.

Definition 6.4
Eine Resolutionswiderlegung einer Menge M von Klauseln ist eine Ableitung der leeren Klausel $\vdash$ aus Klauseln aus M.

Bemerkung.
Da die Resolutionsregel korrekt ist, muß mindestens eine der Klauseln in M falsch sein, falls eine Resolutionswiderlegung von M vorliegt.

Bemerkung.
Formt man eine junktorenlogische Formel A in konjunktive Normalform (KNF) um, so entspricht A eine Klauselmenge $\mathrm{Kl}(A)$.

Definition 6.5
Ein Resolutionsbeweis für eine Formel A ist eine Resolutionswiderlegung von $\mathrm{Kl}(\neg A)$.

Bemerkung. (Vollständigkeit des Resolutionskalküls – hier nicht bewiesen)
Falls eine Menge von Klauseln M semantisch inkonsistent ist, gibt es eine Resolutionswiderlegung von M.

Beispiele.
Man muß KNF von $\neg A$ bilden. Hierzu wird kein semantisches Verfahren (Wahrheitstafeln) verwendet, da dieses die Formel entscheidet und das Resolutionsverfahren überflüssig macht.

Verfahren zur Bestimmung der KNF:

1. Schritt: Eliminiere logische Zeichen, die von $\wedge, \vee, \neg$ verschieden sind.
2. Schritt: Ziehe Negationen nach innen (de Morgan, doppelte Negationsbeseitigung).
3. Schritt: Ziehe $\wedge$ nach außen mit Hilfe der Distributivität:
   $(A \wedge B) \vee C \rightsquigarrow (A \vee C) \wedge (B \vee C)$.

(a) $\vDash A \rightarrow A \vee B$

KNF von $\neg(A \rightarrow A \vee B)$:

$$\begin{aligned} \neg(A \rightarrow A \vee B) &\rightsquigarrow \neg(\neg A \vee A \vee B) \\ &\rightsquigarrow A \wedge \neg A \wedge \neg B \\ &\rightsquigarrow \{\vdash A;\ A\vdash;\ B\vdash\} \end{aligned}$$

Resolutionswiderlegung: $\dfrac{\vdash A \qquad A\vdash}{\vdash}$

(b) $\vDash A \vee (B \vee C) \rightarrow (A \vee B) \vee C$

KNF von $\neg\big(A \vee (B \vee C) \rightarrow (A \vee B) \vee C\big)$:

$$\begin{aligned} \neg\big(A \vee (B \vee C) \rightarrow (A \vee B) \vee C\big) &\rightsquigarrow \neg\big(\neg(A \vee (B \vee C)) \vee ((A \vee B) \vee C)\big) \\ &\rightsquigarrow (A \vee (B \vee C)) \wedge \neg((A \vee B) \vee C) \\ &\rightsquigarrow (A \vee B \vee C) \wedge \neg A \wedge \neg B \wedge \neg C \\ &\rightsquigarrow \{\vdash A, B, C;\ A\vdash;\ B\vdash;\ C\vdash\} \end{aligned}$$

Resolutionswiderlegung: $\dfrac{\dfrac{\dfrac{\vdash A, B, C \qquad A\vdash}{\vdash B, C} \qquad B\vdash}{\vdash C} \qquad C\vdash}{\vdash}$

Alternativ: $A \vee (B \vee C) \vDash (A \vee B) \vee C$

KNF von $A \vee (B \vee C)$: $A \vee B \vee C$
KNF von $\neg\big((A \vee B) \vee C\big)$:

$$\begin{aligned} \neg\big((A \vee B) \vee C\big) &\rightsquigarrow \neg A \wedge \neg B \wedge \neg C \\ &\rightsquigarrow \text{Selbe Klauselmenge wie oben.} \end{aligned}$$

(c) $\vDash(\neg A \to \neg B) \to (B \to A)$

KNF von $\neg\big((\neg A \to \neg B) \to (B \to A)\big)$:

$$
\begin{aligned}
\neg\big((\neg A \to \neg B) \to (B \to A)\big) &\leadsto \neg\big((A \vee \neg B) \to (\neg B \vee A)\big)\\
&\leadsto \neg\big(\neg(A \vee \neg B) \vee \neg B \vee A\big)\\
&\leadsto \neg\big((\neg A \wedge B) \vee \neg B \vee A\big)\\
&\leadsto \neg(\neg A \wedge B) \wedge B \wedge \neg A\\
&\leadsto (A \vee \neg B) \wedge B \wedge \neg A\\
&\leadsto \{B \vdash A;\ \vdash B;\ A \vdash\}
\end{aligned}
$$

Resolutionswiderlegung:
$$
\dfrac{\vdash B \qquad \dfrac{B \vdash A \quad A \vdash}{B \vdash}}{\vdash}
$$

Alternativ: $\neg A \to \neg B \vDash B \to A$

KNF von $\neg A \to \neg B$: $A \vee \neg B$
KNF von $\neg(B \to A)$: $B \wedge \neg A$
ergibt dieselbe Klauselmenge.

(d) $\vDash((A \to B) \to A) \to A$

KNF von $\neg\big(((A \to B) \to A) \to A\big)$:

$$
\begin{aligned}
\neg\big(((A \to B) \to A) \to A\big) &\leadsto \neg\big(\neg(\neg(\neg A \vee B) \vee A) \vee A\big)\\
&\leadsto \neg\big(\neg((A \wedge \neg B) \vee A) \vee A\big)\\
&\leadsto \neg\big((\neg(A \wedge \neg B) \wedge \neg A) \vee A\big)\\
&\leadsto \neg\big(((\neg A \vee B) \wedge \neg A) \vee A\big)\\
&\leadsto ((A \wedge \neg B) \vee A) \wedge \neg A\\
&\leadsto ((A \vee A) \wedge (\neg B \vee A)) \wedge \neg A\\
&\leadsto \{\vdash A;\ B \vdash A;\ A \vdash\}
\end{aligned}
$$

Resolutionswiderlegung: $\dfrac{\vdash A \quad A \vdash}{\vdash}$

(e) $\vDash((A \to B) \wedge (B \to C)) \to (A \vee B \to C)$

KNF von $\neg\big(((A \to B) \wedge (B \to C)) \to (A \vee B \to C)\big)$:

$$
\begin{aligned}
\neg\big(((A \to B) \wedge (B \to C)) \to (A \vee B \to C)\big) &\leadsto \neg\big(((\neg A \vee B) \wedge (\neg B \vee C)) \to \neg(A \vee B) \vee C\big)\\
&\leadsto \neg\big(\neg((\neg A \vee B) \wedge (\neg B \vee C)) \vee \neg(A \vee B) \vee C\big)\\
&\leadsto \neg\big((A \wedge \neg B) \vee (B \wedge \neg C) \vee (\neg A \wedge \neg B) \vee C\big)\\
&\leadsto (\neg A \vee B) \wedge (\neg B \vee C) \wedge (A \vee B) \wedge \neg C\\
&\leadsto \{A \vdash B;\ B \vdash C;\ \vdash A, B;\ C \vdash\}
\end{aligned}
$$

Alternativ: $A \to B,\ B \to C \vDash A \vee B \to C$

KNF von $A \to B$: $\neg A \vee B$
KNF von $B \to C$: $\neg B \vee C$
KNF von $\neg((A \vee B) \to C)$:

$$\begin{aligned} \neg((A \vee B) \to C) &\leadsto \neg\neg((A \vee B) \wedge \neg C) \\ &\leadsto (A \vee B) \wedge \neg C \end{aligned}$$

Klauselmenge: wie vorhin

Resolutionswiderlegung:

$$\dfrac{\dfrac{\dfrac{\vdash A, B \qquad \dfrac{\dfrac{A \vdash B \quad B \vdash C}{A \vdash C} \qquad C \vdash}{A \vdash}}{\vdash B} \qquad B \vdash C}{\vdash C} \qquad C \vdash}{\vdash}$$

kürzer:

$$\dfrac{\dfrac{\dfrac{\vdash A, B \quad A \vdash B}{\vdash B} \qquad B \vdash C}{\vdash C} \qquad C \vdash}{\vdash}$$

Bemerkung.
Ein enormer Teil des Aufwands geht in die Bestimmung der KNF.

7 Skolemisierung

Definition 7.1
Eine Formel A einer Sprache $\mathscr{L}$ ist in *pränexer Normalform*, wenn sie die Form

$$Q_1 x_1 \dots Q_n x_n B[{}^{x_1}\!/\!{}_{a_1}, \dots, {}^{x_n}\!/\!{}_{a_n}]$$

hat, wobei $n \geq 0$, Q_i entweder $\forall$ oder $\exists$ und alle x_i paarweise verschieden.

Bemerkung.
Im Fall $n = 0$ ist A quantorenfrei und identisch mit B.

Definition 7.2
Eine Skolem-Normalform A_{S} zu einer Formel in pränexer Normalform A in einer geeigneten Erweiterungssprache $\mathscr{L}_{\mathrm{S}}$ von $\mathscr{L}$ wird wie folgt definiert:

(i) Falls $n = 0$, dann $A_{\mathrm{S}} \equiv A$.

(ii) Falls Q_1 ein Allquantor $\forall$, dann sei

$$A_{\mathrm{S}} \equiv \big(Q_2 x_2 \dots Q_n x_n B[{}^{x_2}\!/\!{}_{a_2}, \dots, {}^{x_n}\!/\!{}_{a_n}]\big)_{\mathrm{S}}\,.$$

(iii) Falls Q_1 ein Existenzquantor $\exists$, und $b_1, \dots, b_m$ die in A vorkommenden Parameter, dann sei

$$A_{\mathrm{S}} \equiv \big(Q_2 x_2 \dots Q_n x_n B[{}^{f(b_1,\dots,b_m)}\!/\!{}_{a_1}, {}^{x_2}\!/\!{}_{a_2}, \dots, {}^{x_n}\!/\!{}_{a_n}]\big)_{\mathrm{S}}\,,$$

wobei f ein neues (noch nicht in A vorkommendes) m-stelliges Funktionszeichen ist, bzw. eine neue Konstante k, falls $m = 0$.

Bemerkungen.

(i) Das Verfahren ist nichtdeterministisch, solange man nicht die Reihenfolge von Parametern und einzuführenden Zeichen normiert.

(ii) Die Skolem-Normalformen von A und $\forall x A[{}^{x}\!/\!{}_{a}]$ sind identisch (bis auf den in (i) erwähnten Indeterminismus). Das entspricht dem Verständnis von Parametern als Zeichen, die Allgemeinheit ausdrücken.

Beispiele.

(a) $\exists x P(x) \rightsquigarrow P(k)$

(b) $\exists x \forall y P(x, y) \rightsquigarrow P(k, a)$

(c) $\forall x \exists y P(x, y) \rightsquigarrow P(a, f(a))$

(d) $\exists x \forall y \exists z P(x, y, z) \rightsquigarrow P(k, a, f(a))$

(e) $\forall x \exists y \forall z \exists u P(x, y, z, u) \rightsquigarrow P(a, f(a), b, g(a, b))$

Definition 7.3
$\forall A$ sei der *All-Abschluß* einer Formel A, d.h. die Formel

$$\forall x_1 \ldots \forall x_n A[{}^{x_1}/_{a_1}, \ldots, {}^{x_n}/_{a_n}],$$

wobei $a_1, \ldots, a_n$ *alle* Parameter in A und $x_1, \ldots, x_n$ Variablen sind, die in A nicht vorkommen.
(Wieder nichtdeterministische Definition).

Satz 7.1
$\forall A$ ist erfüllbar unter einer Interpretation, gdw. gilt: $\forall A_{\mathrm{S}}$ ist erfüllbar in einer geeigneten erweiterten Interpretation, in der die durch Skolemisierung eingeführten Konstanten und Funktionszeichen interpretiert sind.

Folgerungen.

(i) Um zu zeigen, daß $\forall A$ unerfüllbar ist, d.h. um $\forall A$ zu widerlegen, reicht es aus, $\forall A_{\mathrm{S}}$ zu widerlegen.

(ii) Sei $B_1 \wedge \ldots \wedge B_m$ KNF von A_{S}.
Um zu zeigen, daß $\forall A$ unerfüllbar ist, reicht es aus, $\forall B_1 \wedge \ldots \wedge \forall B_m$ zu widerlegen.

$B_1, \ldots, B_m$ kann man als Klauselmenge schreiben. Sie sei bezeichnet mit A_{SK} (Skolem-Normalform in Klauselform). Definiert man ein Widerlegungsverfahren für Klauseln so, daß man dabei die Parameter in Klauseln *universell* versteht (siehe nächstes Kapitel), dann gilt: Um zu zeigen, daß $\forall A$ unerfüllbar ist, genügt es, A_{SK} nach diesem Verfahren zu widerlegen.

D.h. um zu zeigen, daß eine geschlossene Formel (= Formel ohne Parameter) A allgemeingültig ist, genügt es, $(\neg A)_{\mathrm{SK}}$ nach diesem Verfahren zu widerlegen.
D.h. 3 Schritte:

1. Bilde Skolem-Normalform zu $\neg A$.
2. Bilde daraus KNF.
3. Wende Widerlegungsverfahren an.

oder

1. Bilde pränexe Normalform zu $\neg A$ mit Kern in KNF.
2. Skolemisiere.
3. Wende Widerlegungsverfahren an.

Im Fall von Folgerungsbehauptungen $A_1, \ldots, A_n \vDash A$:
Bilde $(A_1)_{\mathrm{SK}}, \ldots, (A_n)_{\mathrm{SK}}$ und $(\neg A)_{\mathrm{SK}}$.

Beispiele
für Skolemisierung.

(a) $\models \exists x \forall y A(x,y) \to \forall y \exists x A(x,y)$

$$\begin{aligned}
\neg\big(\exists x \forall y A(x,y) \to \forall y \exists x A(x,y)\big) &\rightsquigarrow \neg\big(\forall x \exists y \forall z \exists u (A(x,y) \to A(u,z))\big)\\
&\rightsquigarrow \exists x \forall y \exists z \forall u \neg(A(x,y) \to A(u,z))\\
&\rightsquigarrow \exists x \forall y \exists z \forall u (A(x,y) \wedge \neg A(u,z))\\
&\rightsquigarrow A(k,a) \wedge \neg A(b,f(a))\\
&\rightsquigarrow \{\vdash A(k,a);\; A(b,f(a)) \vdash\}
\end{aligned}$$

(b) $\exists x \forall y A(x,y) \models \forall y \exists x A(x,y)$

$$\exists x \forall y A(x,y) \rightsquigarrow A(k,a)$$

$$\begin{aligned}
\neg \forall y \exists x A(x,y) &\rightsquigarrow \exists y \forall x \neg A(x,y)\\
&\rightsquigarrow \neg A(b,k')
\end{aligned}$$

Man beachte: Obwohl die Ausgangsbehauptungen logisch gleichwertig sind, hat (b) nur die Konstante k' statt des Funktionszeichens f. Dies spielt keine Rolle, da Parameter in jeder Klausel generalisiert verstanden werden.

8 Quantorenlogisches Resolutionsverfahren und Unifikation

Sei σ eine Substitution, auch geschrieben $[{}^{t_1}/_{a_1}, \ldots, {}^{t_n}/_{a_n}]$.

Definition 8.1
σ heißt *Unifikator* von zwei Atomen A und B, wenn gilt: $A\sigma \equiv B\sigma$.

Beispiel.
Gegeben $P(k_1, a)$ und $P(b, k_2)$.

Für $\sigma \equiv [{}^{k_2}/_a, {}^{k_1}/_b]$ gilt $P(k_1, a)\sigma \equiv P(b, k_2)\sigma$.

Definition 8.2
Atome A und B heißen *unifizierbar*, wenn es einen Unifikator σ von A und B gibt.

Definition 8.3
Die *allgemeine Resolutionsregel* lautet wie folgt:

$$\frac{X_1 \vdash Y_1, A \qquad B, X_2 \vdash Y_2}{(X_1, X_2 \vdash Y_1, Y_2)\sigma}\,(\mathcal{R})$$

falls σ ein Unifikator von A und B ist. Hierbei wird vorausgesetzt, daß die Parametermengen der beiden Prämissen disjunkt sind.

Beispiel.

$$\frac{Q(a) \vdash P(k_1, a) \qquad P(b, k_2) \vdash R(b)}{Q(k_2) \vdash R(k_1)}\,[{}^{k_2}/_a, {}^{k_1}/_b]$$

Motivation.
Die Parameter in den Prämissen werden universell verstanden. Also dürfen sie beliebig substituiert werden. Nach einer Substitution, die A und B unifiziert, hat man die normale junktorenlogische Resolutionsregel.

$$\begin{array}{cc}
X_1 \vdash Y_1, A & B, X_2 \vdash Y_2 \\
\Big\downarrow{\scriptstyle \text{Substitution}} & \Big\downarrow{\scriptstyle \text{Substitution}} \\
\end{array}$$

$$\frac{X_1\sigma \vdash Y_1\sigma, A\sigma (\equiv B\sigma) \qquad B\sigma(\equiv A\sigma), X_2\sigma \vdash Y_2\sigma}{X_1\sigma, X_2\sigma \vdash Y_1\sigma, Y_2\sigma}$$

Beispiel.

$$\dfrac{\begin{array}{c} Q(a) \vdash P(k_1,a) \\ \big\downarrow{\scriptstyle \sigma} \\ Q(k_2) \vdash P(k_1,k_2) \end{array} \qquad \begin{array}{c} P(b,k_2) \vdash R(b) \\ \big\downarrow{\scriptstyle \sigma} \\ P(k_1,k_2) \vdash R(k_1) \end{array}}{Q(k_2) \vdash R(k_1)}$$

Das entspricht den Bemerkungen zum Widerlegungsverfahren hinter Satz 7.1 (S. 36).

Da Parameter universell verstanden werden, kann man bei $(\mathcal{R})$ zu Recht verlangen, daß die Parametermengen der beiden Prämissen *disjunkt* sind. Dies wird durch Umbenennung erreicht. (Vgl. Folgerung (ii) auf S. 36). („Separierung von Parametern“).

Jetzt kann das Beispiel von S. 37 fortgesetzt werden:

$\{\vdash A(k,a);\ A(b,f(a)) \vdash\} \rightsquigarrow \{\vdash A(k,a);\ A(b,f(c)) \vdash\}$ (Separierung der Parameter)

Resolutionswiderlegung: $\dfrac{\vdash A(k,a) \qquad A(b,f(c)) \vdash}{\vdash}\ [{}^{f(c)}/_{a}, {}^{k}/_{b}]$

Weitere
Beispiele.

(a) $\vDash \big((\forall x A(x) \to B) \to \exists x (A(x) \to B)\big)$

$$\begin{aligned} \neg\big((\forall x A(x) \to B) \to \exists x (A(x) \to B)\big) &\rightsquigarrow \neg\big(\exists x (A(x) \to B) \to \exists y (A(y) \to B)\big) \qquad (*) \\ &\rightsquigarrow \neg\big(\forall x \exists y ((A(x) \to B) \to (A(y) \to B))\big) \\ &\rightsquigarrow \exists x \forall y \big((A(x) \to B) \land \neg(A(y) \to B)\big) \\ &\rightsquigarrow \exists x \forall y \big((\neg A(x) \lor B) \land (A(y) \land \neg B)\big) \\ &\rightsquigarrow (\neg A(k) \lor B) \land (A(a) \land \neg B) \\ &\rightsquigarrow \{A(k) \vdash B;\ \vdash A(a);\ B \vdash\} \end{aligned}$$

Resolutionswiderlegung:

$$\dfrac{\vdash A(a) \qquad \dfrac{A(k) \vdash B \qquad B \vdash}{A(k) \vdash}}{\vdash}\ [{}^{k}/_{a}]$$

Alternativ:

$$\begin{aligned} \forall x A(x) \to B &\rightsquigarrow \exists x (\neg A(x) \lor B) \\ &\rightsquigarrow \neg A(k) \lor B \end{aligned}$$

$$\begin{aligned} \neg \exists x (A(x) \to B) &\rightsquigarrow \forall x (A(x) \land \neg B) \\ &\rightsquigarrow A(a) \land \neg B \end{aligned}$$

ergibt dieselbe Klauselmenge.

(Dieses Beispiel dient nur Demonstrationszwecken. Es ist insofern nicht wirklich ernstzunehmen, als die in ihm behauptete Gesetzmäßigkeit bei der Bildung der pränexen Normalform schon benutzt wird und überdies die Formel an der Stelle (∗) die trivial widerlegbare Form $\neg(A \to A)$ hat (bis auf gebundene Umbenennung).)

(b) $\text{Bocardo}_{\text{III}}$

$$\begin{array}{ccc} M & \text{o} & P \\ M & \text{a} & S \\ \hline S & \text{o} & P \end{array} \qquad \begin{array}{c} \exists x(M(x) \wedge \neg P(x)) \\ \forall x(M(x) \to S(x)) \\ \hline \exists x(S(x) \wedge \neg P(x)) \end{array}$$

$$\begin{array}{lll} \exists x(M(x) \wedge \neg P(x)) & \forall x(M(x) \to S(x)) & \neg\exists x(S(x) \wedge \neg P(x)) \\ \rightsquigarrow \{P(k)\vdash;\ \vdash M(k)\} & \rightsquigarrow \forall x(\neg M(x) \vee S(x)) & \rightsquigarrow \forall x(\neg S(x) \vee P(x)) \\ & \rightsquigarrow M(a) \vdash S(a) & \rightsquigarrow S(b) \vdash P(b) \end{array}$$

(man kann separat skolemisieren, da Parameter separiert werden können)

$\rightsquigarrow \{P(k)\vdash;\ \vdash M(k);\ M(a)\vdash S(a);\ S(b)\vdash P(b)\}$

Resolutionswiderlegung:

$$\dfrac{\dfrac{\vdash M(k) \qquad \dfrac{M(a)\vdash S(a) \qquad S(b)\vdash P(b)}{M(b)\vdash P(b)}[b/a]}{\vdash P(k)}[k/b] \qquad P(k)\vdash}{\vdash}$$

Zur eigentlichen Resolutionsregel kommt noch die sog. „Faktorisierungsregel“ dazu, die es erlaubt, durch Unifikation mehrere Atome einer Klausel zu identifizieren. Wenn man sie als zwei Regeln, jeweils für Antezedens und Sukzedens, formuliert, lautet sie wie folgt:

$$\dfrac{X, A, B \vdash Y}{(X, A \vdash Y)\sigma}(\mathcal{F}) \qquad\qquad \dfrac{X \vdash A, B, Y}{(X \vdash A, Y)\sigma}(\mathcal{F})$$

falls σ ein Unifikator von A und B ist. De facto handelt es sich um Substitutionsregeln: Durch die Substitution σ werden A und B zu $A\sigma$ (was dasselbe ist wie $B\sigma$) verschmolzen. Die Substitution ist offensichtlich dadurch gerechtfertigt, daß Parameter in Klauseln universell verstanden werden.

Beispiel.
Mithilfe der Faktorisierung ist es möglich, eine Resolutionswiderlegung von

$$\{\vdash P(x_1), P(x_2)\ ;\ P(y_1), P(y_2)\vdash\}$$

zu erzeugen:

$$\dfrac{\dfrac{\vdash P(x_1), P(x_2)}{\vdash P(x_1)}(\mathcal{F}) \qquad \dfrac{P(y_1), P(y_2)\vdash}{P(y_1)\vdash}(\mathcal{F})}{\vdash}(\mathcal{R})$$

Es ist allerdings möglich, $(\mathcal{R})$ und $(\mathcal{F})$ zu einer einzigen *verallgemeinerten Resolutionsregel* zu kombinieren, wenn man erlaubt, in einem Schritt mehrere Paare von Formeln zu unifizieren und zu resolvieren. Im Beispiel würde man zugleich $P(x_1)$ und $P(y_1)$ sowie $P(x_2)$ und $P(y_2)$ unifizieren und resolvieren, d.h. aus den Prämissen $\vdash P(x_1), P(x_2)$ und $P(y_1), P(y_2) \vdash$ in *einem* Schritt $\vdash$ erhalten.

Um Vollständigkeit zu erreichen, ist es wichtig, daß man optimal unifiziert.
Hier gilt:

Satz 8.1
Gegeben zwei Atome A und B. Es gibt einen Algorithmus, der folgendes liefert:

Wenn A und B unifizierbar sind, den allgmeinsten Unifikator (im unten beschriebenen Sinne).
Wenn A und B nicht unifizierbar sind, eine Feststellung dieses Sachverhalts.

Definition 8.4
Die *Komposition* $\sigma\tau$ von Substitutionen ist wie folgt definiert:

$$A\sigma\tau :\equiv (A\sigma)\tau\,.$$

(Bemerkung.
$\sigma\tau$ läßt sich auch explizit als Substitution schreiben (siehe Aufgaben).)

Definition 8.5
σ ist *allgemeinster Unifikator* (*mgu* – „most general unifier“) von A und B, wenn für alle Unifikatoren τ von A und B gilt: $\tau = \sigma\rho$ für eine Substitution ρ.

Satz 8.2
Mehrere allgemeinste Unifikatoren von A und B unterscheiden sich nur durch Umbenennung.

Unifikationsalgorithmus
Gesucht: *mgu* der Atome A und B.
Voraussetzung: Die Parametermengen von A und B sind disjunkt.

Wenn A und B mit verschiedenen Prädikatzeichen beginnen, sind A und B nicht unifizierbar.
Wenn A und B identische 0-stellige Prädikatzeichen (Aussagenkonstanten) sind, sind sie durch die leere Substitution unifizierbar.
Ansonsten führe schrittweise für alle Argumentstellen n von A und B folgende Schritte durch:

- Das n-te Argument von A sei ein Parameter a und das n-te Argument von B ein Term t:
  Falls a in t nicht vorkommt, ersetze in A und B a durch t, füge $[{}^{t}\!/_{a}]$ zur bisher erhaltenen Substitution hinzu und gehe zum $(n+1)$-ten Argument von A über;
  Falls a in t vorkommt, sind A und B nicht unifizierbar.

- Das n-te Argument von A sei kein Parameter, das n-te Argument von B sei ein Parameter: wie vorhin, mit vertauschten Rollen.
- Die n-ten Argumente von A und B seien keine Parameter: Falls sie identische Konstanten sind, gehe zum $(n+1)$-ten Argument von A über. Falls sie identische Funktionszeichen sind, gehe zum ersten Argument des Funktionszeichens über und verfahre schrittweise analog zu vorhin. Falls sie weder identische Konstanten sind noch mit identischen Funktionszeichen beginnen, sind A und B nicht unifizierbar.

Beispiele.

(a) $P(f(a),k)$ und $P(c,f(d))$ sind nicht unifizierbar:

$$\begin{aligned}\langle P(f(a),k),P(c,f(d))\rangle &\rightsquigarrow \langle P(f(a),\underline{k}),P(f(a),\underline{f}(d))\rangle \quad [{}^{f(a)}/_{c}]\\ &\rightsquigarrow \texttt{fail}\end{aligned}$$

(b) $P(f(a),c)$ und $P(b,k)$ sind unifizierbar:

$$\begin{aligned}\langle P(f(a),c),P(b,k)\rangle &\rightsquigarrow \langle P(f(a),c),P(f(a),k)\rangle \quad [{}^{f(a)}/_{b}]\\ &\rightsquigarrow \langle P(f(a),k),P(f(a),k)\rangle \quad [{}^{k}/_{c}]\end{aligned}$$

$$mgu\langle P(f(a),c),P(b,k)\rangle = [{}^{f(a)}/_{b}][{}^{k}/_{c}] = [{}^{f(a)}/_{b},{}^{k}/_{c}]$$

(c) $P(a,a)$ und $P(b,f(b))$ sind nicht unifizierbar:

$$\begin{aligned}\langle P(a,a),P(b,f(b))\rangle &\rightsquigarrow \langle P(b,\underline{b}),P(b,\underline{f(b)})\rangle \quad [{}^{b}/_{a}]\\ &\rightsquigarrow \texttt{fail} \quad (\textit{occur check})\end{aligned}$$

(d) $P(k,a,h(g(c)))$ und $P(d,h(b),h(b))$ sind unifizierbar:

$$\begin{aligned}\langle P(k,a,h(g(c))),P(d,h(b),h(b))\rangle &\rightsquigarrow \langle P(k,a,h(g(c))),P(k,h(b),h(b))\rangle \quad [{}^{k}/_{d}]\\ &\rightsquigarrow \langle P(k,h(b),h(g(c))),P(k,h(b),h(b))\rangle \quad [{}^{h(b)}/_{a}]\\ &\rightsquigarrow \langle P(k,h(g(c)),h(g(c))),P(k,h(g(c)),h(g(c)))\rangle\\ &\qquad [{}^{g(c)}/_{b}]\end{aligned}$$

$$mgu\langle P(k,a,h(g(c))),P(d,h(b),h(b))\rangle = [{}^{k}/_{d}][{}^{h(b)}/_{a}][{}^{g(c)}/_{b}] = [{}^{k}/_{d},{}^{h(g(c))}/_{a},{}^{g(c)}/_{b}]$$

Bemerkung.

Im letzten Beispiel sieht man den fundamentalen Unterschied zwischen simultaner Substitution

$$[{}^{t_1}/_{a_1},\dots,{}^{t_n}/_{a_n}]$$

und der Hintereinanderausführung von Substitutionen

$$[{}^{t_1}/_{a_1}]\dots[{}^{t_n}/_{a_n}].$$

Es gilt nämlich im Beispiel *nicht*: $[{}^{k}/_{d}][{}^{h(b)}/_{a}][{}^{g(c)}/_{b}] = [{}^{k}/_{d},{}^{h(b)}/_{a},{}^{g(c)}/_{b}]$.

9 SLD-Resolution

Die SLD-Resolution ist eine eingeschränkte Form des Resolutionsverfahrens. Bei dieser benutzt man jedoch die berechneten Unifikatoren, um zusätzliche Information zu erlangen.

Definition 9.1
Eine *definite Klausel* (auch *definite Hornklausel*) ist eine Klausel mit höchstens einem positiven Literal, d.h. eine Klausel der Form $X \vdash$ oder $X \vdash A$. Man schreibt auch $\leftarrow X$ bzw. $A \leftarrow X$. Eine Klausel der Form $\leftarrow X$ heißt auch *Zielklausel*, eine Klausel der Form $A \leftarrow X$ auch *Programmklausel*. A heißt *Kopf*, X heißt *Rumpf* von $A \leftarrow X$.

Motivation für die Schreibweise.
$X \vdash A$ kann man als „A, falls X" lesen, $X \vdash$ als „X ist falsch", d.h. „falsum, falls X".

Definition 9.2
Ein *SLD-Resolutionsschritt* hat die Form

$$\frac{\leftarrow X, A \qquad B \leftarrow Y}{(\leftarrow X, Y)\sigma}\,\sigma$$

falls σ A und B unifiziert. Die Prämissen sollen keine Parameter gemeinsam haben.

Definition 9.3
Sei eine Menge von Programmklauseln $\mathcal{P}$ gegeben (auch *Programm* genannt) und eine Zielklausel $\leftarrow X$. Eine *SLD-Widerlegung von* $\leftarrow X$ *relativ zu* $\mathcal{P}$ ist eine Folge von SLD-Resolutionsschritten, die mit $\leftarrow X$ als linker oberster Annahme beginnt, ansonsten nur Klauseln aus $\mathcal{P}$ als Annahmen benutzt, und mit der leeren Klausel $\leftarrow$ endet.

Definition 9.4
Ein *SLD-Beweis* für X aus $\mathcal{P}$ ist eine SLD-Widerlegung von $\leftarrow X$ relativ zu $\mathcal{P}$.

Bemerkung.
Ein *SLD-Beweis* für X_1 aus $\mathcal{P}$ sieht also wie folgt aus:

$$\begin{array}{l} \dfrac{\dfrac{\leftarrow X_1 \quad A_1 \leftarrow Y_1}{\leftarrow X_2}\,\sigma_1 \qquad\qquad A_2 \leftarrow Y_2}{\leftarrow X_3}\,\sigma_2 \\ \qquad\qquad\qquad \ddots \\ \qquad\qquad\qquad\quad \leftarrow \end{array}$$

σ_1, σ_2 sind dabei die im jeweiligen Schritt berechneten Substitutionen (im optimalen Fall allgemeinste Unifikatoren).

S: „selection" (Auswahl des zu unifizierenden Atoms in der Zielklausel)
L: „linear" (Form des Beweises)
D: „definite" Klauseln benutzt.

Falls man die Programmklauseln durchnumeriert hat, schreibt man auch:

$$X_1 \xrightarrow{(n_1),\sigma_1} X_2 \xrightarrow{(n_2),\sigma_2} X_3 \cdots\cdots\cdots X_k \xrightarrow{(n_k),\sigma_k} X_{k+1} \cdots\cdots\cdots$$

Bemerkung.
Einen SLD-Beweis für X kann man als Beweis für $\neg\forall\neg\bigwedge X$ auffassen, da Parameter universell verstanden werden; falls X nur aus einem Atom A besteht, also als Beweis von $\neg\forall\neg A$. Dies ist ein Beweis für $\exists A$.
Nun sind die „unterwegs“ benutzten Unifikatoren gerade die Spezialfälle, für die $\forall\neg A$ falsch wird, d.h. für die $\exists A$ wahr wird. Wir wissen also nicht nur, daß $\exists A$ gilt, sondern auch, für welche Substitutionen der Parameter A gilt. Ein SLD-Beweis für A liefert also zugleich eine Substitution σ, so daß $A\sigma$ gilt. σ ist die Substitution, die sich aus der sukzessiven Anwendung der SLD-Regel ergibt. Ein SLD-Beweis antwortet also auf die Frage: „Für welche Substitution σ ist A gültig relativ zu $\mathcal{P}$?“, formal:

$$?\sigma : \quad \mathcal{P} \models A\sigma$$

bzw. allgemein für X:

$$?\sigma : \quad \mathcal{P} \models A\sigma \quad \text{für alle } A \in X\,.$$

Damit ergibt sich eine gänzlich andere Stoßrichtung von Resolution. Nicht für beliebige Formeln, die in Skolem-Normalform gebracht werden, sondern für spezifische Formeln, die von vornherein in bestimmter Form sind:

- Programmklauseln, welche die Gesetzmäßigkeiten eines Gegenstandsbereichs beschreiben.
- Zielklauseln, die etwas beschreiben, was man über den Gegenstandsbereich wissen will.

Die entsprechende Logik ist unentscheidbar, aber maschinell behandelbar: Logik-Programmierung.

Eine andere Intuition besteht darin, SLD-Derivationen von oben nach unten als „Rückwärts-Schließen“ zu interpretieren.
Gegeben als Ziel: X, A (notiert als $\leftarrow X, A$).
Wir wissen: $B\sigma$ falls $Y\sigma$ für alle σ (notiert als $B \leftarrow Y$).
Ferner wissen wir: $A\sigma \equiv B\sigma$.
Als neues Ziel ergibt sich damit: $X\sigma, Y\sigma$.
D.h., wenn wir X, A für die Substitution σ nachweisen wollen, reicht es aus, $X\sigma, Y\sigma$ nachzuweisen.
„*Zielgerichtetes Räsonieren*“.

Dies schließt die Abfrage von Daten ein. Das Ziel enthält freie Variablen für die zu suchenden Daten, das Programm enthält die Daten in Form von Fakten. Es gibt keinen prinzipiellen Unterschied zwischen Daten und Programmen.

Beispiel.

$$\frac{\leftarrow P(a,b) \quad P(1,2) \leftarrow}{\leftarrow} \, [1/a, 2/b]$$

Frage: „Für welche a, b gilt $P(a, b)$."
Antwort: „Für $a = 1$ und $b = 2$".

Etwas komplizierter:

Programm:
Geschwister$(a, b) \leftarrow E$ltern(a, c, d), Eltern(b, c, d)
Eltern$(a, b, c) \leftarrow M$utter(a, b), Vater(a, c)
Mutter(Hans, Anna) $\leftarrow$
Mutter(Karin, Anna) $\leftarrow$
Vater(Karin, Fritz) $\leftarrow$
Vater(Hans, Fritz) $\leftarrow$

Frage: ?– Geschwister(Hans, a)

Der Übersichtlichkeit halber wird folgender SLD-Beweis mit Pfeilen statt Schlußstrichen notiert. Die jeweils benutzten Substitutionen werden neben den Pfeilen vermerkt. Da Parameter in Programmklauseln jeweils so umbenannt werden müssen, daß sie von denen in den jeweiligen Zielklauseln verschieden sind, kann die bei einem Resolutionsschritt berechnete Substitution aufgespalten werden in die Substitution des Ziels und die der Programmklausel. Das bedeutet, daß sich die bei Schlußstrich-Notation neben dem Schlußstrich vermerkte Substitution σ zusammensetzt aus den beiden Substitutionen, die neben den von Ziel- bzw. Programmklauseln ausgehenden Pfeilen vermerkt sind.

$$
\begin{tikzcd}[column sep=tiny]
\leftarrow G(H,a) \arrow[dr, swap, "{[c/a]}"] & & G(b,c) \leftarrow E(b,e,f), E(c,e,f) \arrow[dl, "{[H/b]}"] & & & & & & \\
& \leftarrow E(H,e,f), E(c,e,f) \arrow[dr] & & E(g_1,g_2,g_3) \leftarrow M(g_1,g_2), V(g_1,g_3) \arrow[dl, "{[H/g_1, e/g_2, f/g_3] \quad (*)}"] & & & & & \\
& & \leftarrow E(c,e,f), M(H,e), V(H,f) \arrow[dr, swap, "{[A/e]}"] & & M(H,A) \leftarrow \arrow[dl, "(\dagger)"] & & & & \\
& & & \leftarrow E(c,A,f), V(H,f) \arrow[dr, swap, "{[F/f]}"] & & V(H,F) \leftarrow \arrow[dl] & & & \\
& & & & \leftarrow E(c,A,F) \arrow[dr] & & E(h_1,h_2,h_3) \leftarrow M(h_1,h_2), V(h_1,h_3) \arrow[dl, "{[c/h_1, A/h_2, F/h_3]}"] & & \\
& & & & & \leftarrow M(c,A), V(c,F) \arrow[dr, swap, "{(**) \quad [K/c]}"] & & M(K,A) \leftarrow \arrow[dl] & \\
& & & & & & \leftarrow V(K,F) \arrow[dr] & & V(K,F) \leftarrow \arrow[dl] \\
& & & & & & & \leftarrow &
\end{tikzcd}
$$

Die für das Ausgangsziel *berechnete Antwortsubstitution* ergibt sich aus den Substitutionen längs der Pfeile, die von Zielen ausgehen: $[c/a][A/e][F/f][K/c]$. Daraus ergibt sich als für das Ausgangsziel *relevante Substitution*: $[c/a][K/c]$, was gleichwertig ist mit: $[K/a]$.

Die Antwort lautet also: *G*eschwister(*H*ans, *K*arin) bzw. $a = K$arin.

Zum Auswertungsverfahren:

An der Stelle $(*)$ gab es eine Wahlmöglichkeit. Es wurde $E(H,e,f)$ mit $E(g_1,g_2,g_3)$ unifiziert. Man hätte auch $E(c,e,f)$ mit $E(g_1,g_2,g_3)$ unifizieren können. In die Auswertung eines aus mehreren Atomen bestehenden Ziels geht also bei jedem Schritt eine *Auswahl* desjenigen Atoms ein, das mit dem Kopf einer Programmklausel unifiziert werden soll. Diese Auswahl kann man von einer *Auswahlfunktion* („selection function“) vorgeben lassen, die bei jedem Schritt bestimmt, wie die Auswahl zu treffen ist. (SLD-Resolution).

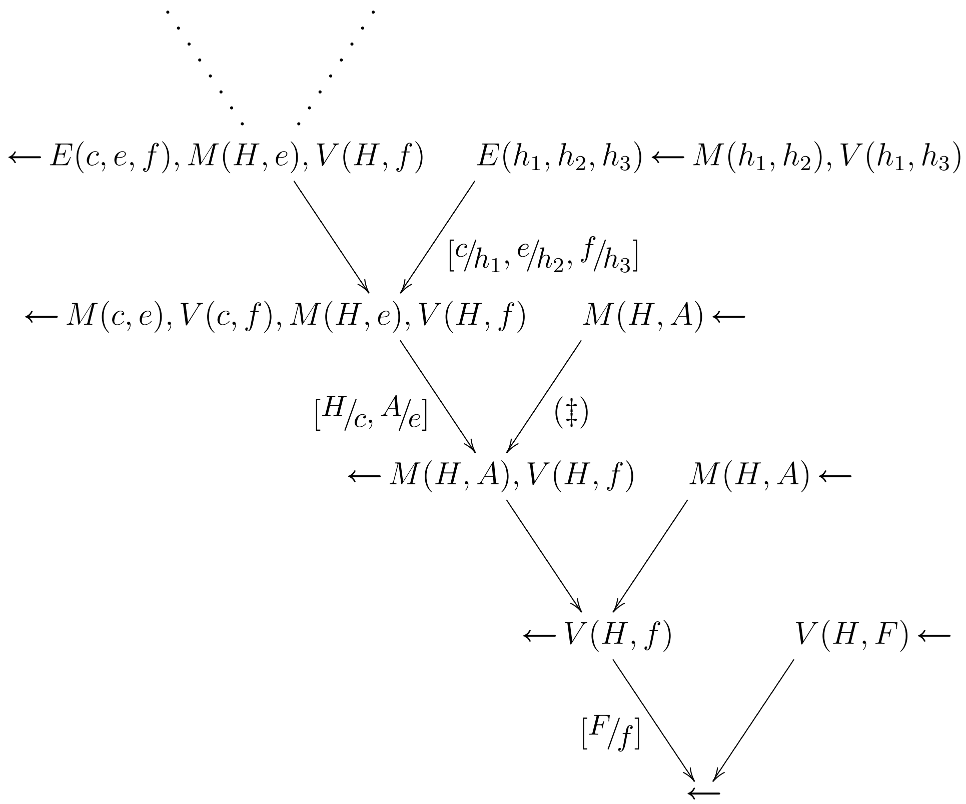
Es läßt sich zeigen, daß die Menge der Antwortsubstitutionen, die sich aus einer Abfrage gewinnen läßt, nicht davon abhängt, welche Auswahlfunktionen benutzt werden. In der Regel benutzt man das linkeste Atom.

Eine zweite Quelle des Indeterminismus liegt in der Auswahl der Programmklausel, die man anwendet. Diese bestimmt, welche Antwortsubstitutionen sich berechnen lassen, wobei es gegebenenfalls mehr als eine gibt. Bei falscher Wahl kann sich auch eine nichtdeterministische Berechnung ergeben. Z.B. hätten wir an der Stelle ($**$) auch die Substitution $[{}^{H}\!/_{c}]$ zusammen mit der Programmklausel $M(H, A)$ wählen können. Der SLD-Beweis hätte dann so ausgesehen:

$$
\begin{tikzcd}
\leftarrow M(c,A), V(c,F) \arrow[dr, "{[{}^{H}\!/_{c}]}"'] & & M(H,A) \leftarrow \arrow[dl] & \\
& \leftarrow V(H,F) \arrow[dr] & & V(H,F) \leftarrow \arrow[dl] \\
& & \leftarrow &
\end{tikzcd}
$$

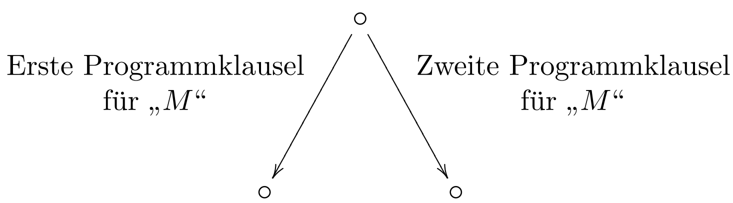
Als Antwort hätte sich damit ergeben: *G*eschwister(*H*ans, *H*ans) bzw. $a = H$ans.
Diese Antwort ist (neben *G*eschwister(*H*ans, *K*arin) bzw. $a = K$arin) korrekt, da die Verschiedenheit der Argumente von *G*eschwister im Programm nicht gefordert wurde.

Zu dieser Antwort hätte man auch wie folgt kommen können: Nehmen wir an, wir hätten an der Stelle (†) $E(c, e, f)$ ausgewertet (wie es sein müßte, würde man immer das linkeste Atom einer Zielklausel auswählen). Dann könnte man wie folgt fortfahren:



An der Stelle (‡) hätte man hinwiederum die Wahl gehabt, die Programmklausel $M(K,A)$ zu benutzen, womit man die intendierte Antwort (Geschwister(Hans, Karin) bzw. $a = K$arin) erhalten hätte.

Eine Wahl der Programmklausel wie bei (‡) kann man als Verzweigung in einem Suchbaum, der alle Auswertungsstrategien erfaßt, beschreiben:



Ein solcher Suchbaum ist nicht zu verwechseln mit einem SLD-Beweis.

Auswertungssysteme der Programmiersprache PROLOG versuchen, alle korrekten Antworten systematisch zu erfassen, indem sie den Suchbaum in geeigneter Weise durchlaufen. Das Verfahren, im Falle des Scheiterns eines SLD-Resolutionsschritts zu einem früheren Verzweigungspunkt zurückzugehen und eine andere Alternative zu wählen, heißt *backtracking*.

Literaturempfehlungen

Gentzen, G. (1935). *Untersuchungen über das logische Schließen.* Mathematische Zeitschrift, 39, 176-210, 405-431. (elektronisch: `http://gdz.sub.uni-goettingen.de`).

Goltz H.-J., Herre, H. (1990). *Grundlagen der logischen Programmierung.* Berlin: Akademie-Verlag.

Kleene, S.C. (1996). *Introduction to Metamathematics.* Amsterdam: North-Holland.

Lloyd, J.W. (1987). *Foundations of Logic Programming.* Berlin: Springer.

Nienhuys-Cheng, S.-H., Wolf, R. de (1997). *Foundations of Inductive Logic Programming* (LNAI 1228). Berlin: Springer. (Part I: Logic).

Prawitz, D. (1965). *Natural Deduction. A Proof-Theoretical Study.* Stockholm: Almqvist & Wiksell.

Schöning, U. (2000). *Logik für Informatiker.* Heidelberg: Spektrum Akademischer Verlag.

Smullyan, R.M. (1995). *First-Order Logic.* New York: Dover.

Troelstra, A.S., Schwichtenberg, H. (2002). *Basic Proof Theory.* Cambridge University Press.