Einführung in die Logik

Klaus Madlener

2002

Vorwort

Dieses Skript entstand aus dem "alten" Vorlesungsskript "Einführung in die Logik und Korrektheit von Programmen", das sich über viele Jahre bewährte. Es umfasst nur den Teil über Logik – die Kapitel über Programmverifikation wurden weggelassen. Hauptziel ist es die Vorteile einer Formalisierung der Logik deutlich zu machen. Der Übergang von der Umgangssprache zu einer formalisierten Sprache, welcher erfahrungsgemäß gewisse Schwierigkeiten bereitet, wird besprochen und eingeübt.

Die Auswahl des Stoffes für eine solche Vorlesung bereitet immer Probleme, denn das Gebiet der Logik ist so umfangreich, dass hier nur die wichtigsten Ergebnisse vorgestellt werden können.

Die Aussagenlogik dient der Einführung der Haupttechniken der Logik. Die Formalisierung von Aussagen und die Einführung des semantisch begründeten Folgerungsbegriffes sind Vorbilder für die Vorgehensweise in anderen Logiken. Eine der wesentlichen Eigenschaften der Logik, dass man nämlich den Folgerungsbegriff durch nachweislich äquivalente formale Ableitungssysteme ersetzen kann, wird ausführlich diskutiert. Der Kompaktheitssatz für die Aussagenlogik dient ebenfalls als Vorbild für den entsprechenden Satz der Prädikatenlogik.

Ein wesentlicher Teil der Vorlesung, nämlich die Übungen, sind in diesem Skript nicht enthalten. Es sei darauf hingewiesen, dass das Skript ausgezeichnete Literatur auf diesem Gebiet nicht ersetzen kann, sondern eher die Stoffauswahl der Vorlesung umreißt und Anregung zu weiterem Studium geben soll.

Inhaltsverzeichnis

1 Aussagenlogik

1.1 Die Sprache der Aussagenlogik

Als Einleitung und zur Einführung in die Methoden der Logik geht es in diesem Kapitel um die Grundlagen der Aussagenlogik. Im Mittelpunkt steht die Frage nach der Bedeutung einer Aussage: Wann ist eine Aussage wahr und wann ist sie falsch? Darüber hinaus beschäftigt sich die Aussagenlogik mit dem Zusammensetzen mehrerer Aussagen zu neuen, komplexeren Aussagen und den Zusammenhängen von Aussagen untereinander. Um die Bedeutung einer Aussage zu bestimmen, müssen zunächst die Begriffe "wahr" und "falsch" geklärt werden. Dazu ist eine Formalisierung des Wahrheitsbegriffes nötig.

1.1 Definition (Sprache der Aussagenlogik)**.** Sei $\Sigma = V \cup O \cup K$ ein *Alphabet*, wobei $V = \{p_1, p_2, \ldots\}$ eine abzählbare Menge von *Aussagevariablen* ist, $O = \{\wedge, \vee, \neg, \rightarrow, \ldots\}$ eine Menge von *Verknüpfungen (Junktoren)* bezeichnet und $K = \{(,)\}$ die Klammern beinhaltet. Die Menge der *Aussageformen* $F \subseteq \Sigma^\star$ (auch *Formeln* der Aussagenlogik genannt) wird induktiv definiert durch

1. $V \subseteq F$, die Menge der atomaren Aussagen,
2. falls $A, B \in F$, dann $(\neg A), (A \wedge B), (A \vee B), (A \rightarrow B) \in F$ und
3. F ist die kleinste Menge von Elementen aus $\Sigma^\star$, die V enthält und 2. erfüllt.

Zur Vereinfachung der Schreibweise werden äußere Klammern gelegentlich weggelassen.

1.2 Bemerkung. Eigenschaften von F werden mit Hilfe der "strukturellen Induktion", das heißt einer Induktion über den Aufbau von F bewiesen. Beispiele für Eigenschaften von F sind:

1. Für $A \in F$ gilt: A ist atomar oder beginnt mit "(" und endet mit ")".
2. Sei $f(A, i)$ die Anzahl der "(" minus der Anzahl der ")" in den ersten i Buchstaben von $A \in F$. Dann gilt $f(A, i) > 0$ für $1 \leq i < |A|$ und $f(A, i) = 0$ für $i = |A|$.

Ist $U = \Sigma^\star$, so kann F auch als Erzeugnis einer Relation $R \subseteq U^\star \times U$ angegeben werden (F wird frei von einer solchen Relation erzeugt, d.h. für alle $u, v \in F^\star$ und $A \in F$ gilt: Aus uRA und vRA folgt $u = v$).
Man kann zeigen, dass es eine eindeutige kontextfreie Grammatik gibt, die F erzeugt. F ist entscheidbar.

1.3 Satz (Eindeutigkeitssatz)**.** Jede Aussageform $A \in F$ ist entweder atomar oder sie lässt sich eindeutig darstellen als $A \equiv (\neg A_1)$ oder $A \equiv (A_1 \star A_2)$ mit $\star \in \{\vee, \wedge, \rightarrow\}$ und $A_1, A_2 \in F$.

Beweis. Induktion über den Aufbau von F.

Ind. Anf.: Sei $A \in V$, d.h. A ist atomar. $\surd$

Ind. Schritt : Zu zeigen ist, dass man zu einer gegebenen Darstellung von $A \in F \backslash V$ keine weitere Darstellung von A finden kann, die mit dieser nicht übereinstimmt.

1. Fall: $A \equiv (\neg B), B \in F$.

   a) Ist $A \equiv (\neg C)$ mit $C \in F$ eine weitere Darstellung von A, dann ist die Darstellung von C nach Induktionsvoraussetzung gleich der Darstellung von B.

   b) Gäbe es eine andere Darstellung für A, müßte sie von der Gestalt $(D \star E)$ sein, mit $D, E \in F$ und $\star \in \{\wedge, \vee, \rightarrow\}$. Das zweite Zeichen "$\neg$" von A müßte dann auch erstes Zeichen von D sein. Das widerspricht aber der Definition von F, wonach D entweder eine Aussagevariable ist oder eine Formel, die mit "(" beginnt (s. 1.2). Diese Darstellung von A ist also eindeutig.

2. Fall: $A \equiv (B \star C)$ mit $B, C \in F, \star \in \{\wedge, \vee, \rightarrow\}$.

   a) Eine andere Darstellung von A der Gestalt $(\neg D), D \in F$, führt analog Fall 1.(b) zu einem Widerspruch zur Definition der Formeln.

   b) Sei $A \equiv (D \star_1 E)$ eine andere Darstellung für A mit $D, E \in F$ und $\star_1 \in \{\wedge, \vee, \rightarrow\}$. Falls $|D| = |B|$ ist, muss $\star_1 = \star$ sein und nach Induktionsvoraussetzung dann auch $D = B$ und $E = C$ gelten. Für den Fall $|D| \neq |B|$ sei o.B.d.A. $|B| > |D|$, so dass folgende Situation entsteht:

$$\begin{array}{lcl} A & \equiv & (\overset{B}{\rule{4cm}{0.5pt}} \star \overset{C}{\rule{2.5cm}{0.5pt}}) \\ A & \equiv & (\overset{D}{\rule{2.5cm}{0.5pt}} \star_1 \overset{E}{\rule{4cm}{0.5pt}}) \\ & & \qquad\qquad\qquad\quad \uparrow \end{array}$$

   Betrachte die Anzahl der nicht geschlossenen, öffnenden Klammern $f(A, 1 + |D|)$ in Formel A an der markierten Stelle (f aus 1.2). Es gilt offensichtlich $f(A, 1 + |D|) = f(D, |D|) + 1 = 1$. Da D Präfix von B ist und $f(B, |D|) \geq 1$, muss außerdem gelten $1 = f(A, 1+|D|) = 1+f(B, |D|) \geq 2$, was die Annahme einer zweiten Darstellung von A widerlegt.

■

1.2 Die Semantik der Aussagenlogik

1.4 Definition. Eine *Bewertung* der Aussageformen ist eine Funktion $\varphi : F \rightarrow \mathbb{B} := \{0, 1\}$ mit

1. $\varphi(\neg A) = 1 - \varphi(A)$,
2. $\varphi(A \vee B) = \max\{\varphi(A), \varphi(B)\}$,

3. $\varphi(A \wedge B) = \min\{\varphi(A), \varphi(B)\}$ und

4. $\varphi(A \to B) = \begin{cases} 0, & \text{falls } \varphi(A) = 1 \text{ und } \varphi(B) = 0, \\ 1, & \text{sonst.} \end{cases}$

Man sagt "A ist 'falsch' unter φ", falls $\varphi(A) = 0$ und " 'wahr' unter φ", falls $\varphi(A) = 1$. Eine mögliche Darstellung für eine Bewertung sind die sogenannten "Wahrheitstafeln":

$\varphi(A)$	$\varphi(\neg A)$
0	1
1	0

$\varphi(A)$	$\varphi(B)$	$\varphi(A \vee B)$	$\varphi(A \wedge B)$	$\varphi(A \to B)$
0	0	0	0	1
0	1	1	0	1
1	0	1	0	0
1	1	1	1	1

Eine *Belegung* der aussagenlogischen Variablen V ist eine Funktion $\varphi_0 : V \to \mathbb{B}$. Jede Bewertung auf F induziert eine Belegung von V. Umgekehrt gilt:

1.5 Lemma. Jede Belegung $\varphi_0 : V \to \mathbb{B}$ lässt sich auf genau eine Weise zu einer Bewertung $\varphi : F \to \mathbb{B}$ fortsetzen.

Der Beweis folgt unmittelbar aus Satz 1.3 und Definition 1.4.

1.6 Folgerung. Jede Bewertung $\varphi : F \to \mathbb{B}$ wird eindeutig durch die Werte von φ auf V festgelegt. Die Bewertung einer Aussageform $A \in F$ hängt nur von den Werten der in ihr vorkommenden Aussagevariablen aus V ab. Das heißt, will man $\varphi(A)$ berechnen, genügt es, die Werte $\varphi(p)$ zu kennen für alle Aussagevariablen p, die in A vorkommen.

Beispiel:
Sei $\varphi(p) = 1, \varphi(q) = 1, \varphi(r) = 0$. Dann kann $\varphi(A)$ iterativ berechnet werden:

$$A \equiv \underbrace{(\underbrace{(\underbrace{p}_{1} \to \underbrace{(\underbrace{q}_{1} \to \underbrace{r}_{0})}_{0})}_{0} \to \underbrace{(\underbrace{(\underbrace{p}_{1} \wedge \underbrace{q}_{1})}_{1} \to \underbrace{r}_{0})}_{0})}_{1}$$

Also gilt $\varphi(A) = 1$.

Es stellt sich die Frage, welche Werte $\varphi(A)$ annehmen kann, wenn φ <u>alle</u> Belegungen durchläuft. Ist etwa $\varphi(A) = 1$ für alle Belegungen φ? Um das nachzuprüfen, genügt es, die <u>endlich</u> vielen unterschiedlichen Belegungen der Variablen, die in A vorkommen, zu überprüfen. Kommen n Variablen in A vor, so gibt es 2^n verschiedene Belegungen:

Beispiel: Für die drei Variablen p, q und r aus A im obigen Beispiel gibt es 8 Belegungen, die betrachtet werden müssen:

$\varphi(p)$	$\varphi(q)$	$\varphi(r)$	$\varphi(q \to r)$	$\varphi(p \wedge q)$	$\varphi(p \to (q \to r))$	$\varphi((p \wedge q) \to r)$	$\varphi(A)$
0	0	0	1	0	1	1	1
0	0	1	1	0	1	1	1
0	1	0	0	0	1	1	1
0	1	1	1	0	1	1	1
1	0	0	1	0	1	1	1
1	0	1	1	0	1	1	1
1	1	0	0	1	0	0	1
1	1	1	1	1	1	1	1

Man sieht, dass $\varphi(A) = 1$ für jede Bewertung $\varphi : F \to \mathbb{B}$ gilt. Die Beobachtung, dass es Aussageformen gibt, die "wahr" sind, unabhängig davon wie ihre Variablen belegt werden, legt die folgende Definition nahe.

1.7 Definition. Sei $A \in F$, $\Sigma \subseteq F$.

1. a) A heißt *Tautologie*, falls $\varphi(A) = 1$ für jede Bewertung φ gilt. (Schreibweise "$\models A$")
   b) A ist *allgemeingültig*, falls A Tautologie ist.
   c) A ist *erfüllbar*, falls es eine Bewertung φ gibt, mit $\varphi(A) = 1$.
   d) A ist *widerspruchsvoll*, falls $\varphi(A) = 0$ für jede Bewertung φ.

2. a) Σ ist *erfüllbar*, falls es eine Bewertung φ gibt mit $\varphi(A) = 1$ für alle $A \in \Sigma$. ("*φ erfüllt Σ*")
   b) *Semantischer Folgerungsbegriff:* A ist *logische Folgerung* von Σ, falls $\varphi(A) = 1$ für jede Bewertung φ, die Σ erfüllt. Man schreibt "$\Sigma \models A$". Ist $\Sigma = \{A_1, \ldots, A_n\}$, ist die Kurzschreibweise "$A_1, \ldots, A_n \models A$" üblich.
   c) Die Menge $\text{Folg}(\Sigma)$ der Folgerungen aus Σ ist definiert durch:

$$\text{Folg}(\Sigma) := \{A \mid A \in F \text{ und } \Sigma \models A\}.$$

1.8 Bemerkungen und Beispiele.

1. $(p \vee (\neg p))$, $((p \to q) \vee (q \to r))$ und A aus Folgerung 1.6 sind Tautologien. $(p \wedge (\neg p))$ ist widerspruchsvoll. $(p \wedge q)$ ist erfüllbar jedoch keine Tautologie. Die Mengen

$$\text{Taut} := \{A \mid \ A \in F \text{ und } \models A\}$$

und

$$\text{Sat} := \{A \mid \ A \in F \text{ und } A \text{ ist erfüllbar}\}$$

sind entscheidbar.

2. a) Ist $\Sigma = \emptyset$, dann gilt $\Sigma \models A$ genau dann, wenn A Tautologie ist, d.h. $\text{Folg}(\emptyset) = \text{Taut}$.
   b) Ist Σ nicht erfüllbar, dann gilt $\Sigma \models A$ für alle $A \in F$, d.h. $\text{Folg}(\Sigma) = F$.

c) Sei $\Sigma = \{p\}$ und $A \equiv p \vee q$. Dann gilt $\Sigma \models A$, denn falls $\varphi(p) = 1$, dann auch $\varphi(p \vee q) = 1$. Jede Bewertung, die Σ erfüllt, erfüllt also auch A.

d) Sei $\Sigma \subseteq \Sigma'$. Ist Σ' erfüllbar, dann ist auch Σ erfüllbar.

e) Es gilt $\Sigma \subseteq \mathrm{Folg}(\Sigma)$ und $\mathrm{Folg}(\mathrm{Folg}(\Sigma)) = \mathrm{Folg}(\Sigma)$.

f) Falls $\Sigma \subseteq \Sigma'$, dann gilt $\mathrm{Folg}(\Sigma) \subseteq \mathrm{Folg}(\Sigma')$.

3. $\Sigma \models A$ gilt genau dann, wenn $\Sigma \cup \{\neg A\}$ nicht erfüllbar ist. Ist Σ endlich, dann ist es entscheidbar, ob Σ erfüllbar ist, und die Menge $\mathrm{Folg}(\Sigma)$ ist entscheidbar.

1.9 Lemma (Deduktionstheorem und Modus-Ponens-Regel)**.**

a) Deduktionstheorem: $\Sigma, A \models B$ gilt genau dann, wenn $\Sigma \models (A \rightarrow B)$ gilt.

(Σ, A ist Kurzschreibweise für $\Sigma \cup \{A\}$)

b) Modus-Ponens-Regel: Es gilt $\{A, A \rightarrow B\} \models B$. Insbesondere ist B eine Tautologie, falls A und $(A \rightarrow B)$ Tautologien sind.

Beweis. zu (a):

"$\Longrightarrow$": Voraussetzung: $\Sigma, A \models B$. Sei $\varphi : F \rightarrow \mathbb{B}$ eine Bewertung, die Σ erfüllt. Ist $\varphi(A) = 0$, dann ist $\varphi(A \rightarrow B) = 1$. Ist $\varphi(A) = 1$, dann gilt nach Voraussetzung $\varphi(B) = 1$, also auch $\varphi(A \rightarrow B) = 1$.

"$\Longleftarrow$": analog.

zu (b):

Ist φ eine Bewertung mit $\varphi(A) = \varphi(A \rightarrow B) = 1$, dann ist $\varphi(B) = 1$. ■

Übliche Notationen für Regeln der Form "$A_1, \dots, A_n \models B$" sind:

$$\begin{array}{c} A_1 \\ \vdots \\ A_n \\ \hline B \end{array} \quad \text{und} \quad \frac{A_1, \dots, A_n}{B}$$

1.10 Satz (Kompaktheitssatz der Aussagenlogik)**.** $\Sigma \subseteq F$ ist erfüllbar genau dann, wenn jede endliche Teilmenge von Σ erfüllbar ist. Insbesondere gilt $\Sigma \models A$ genau dann, wenn es eine endliche Teilmenge $\Sigma_0 \subseteq \Sigma$ gibt mit $\Sigma_0 \models A$.

Beweis. Σ heißt *endlich erfüllbar (e.e.)*, wenn jede endliche Teilmenge von Σ erfüllbar ist. Zu zeigen ist also: Σ ist endlich erfüllbar genau dann, wenn Σ erfüllbar ist.

"$\Longleftarrow$": folgt unmittelbar aus der Bemerkung 1.8 2.(d).

"$\Longrightarrow$": Sei Γ e.e. und sei $A \in F$. Dann ist $\Gamma \cup \{A\}$ oder $\Gamma \cup \{\neg A\}$ e.e. Denn angenommen $\Gamma \cup \{A\}$ ist nicht e.e., dann gibt es eine endliche Teilmenge $\Gamma_0 \subseteq \Gamma$, so dass $\Gamma_0 \cup \{A\}$ nicht erfüllbar ist.

Sei $\Gamma_1 \subseteq \Gamma \cup \{\neg A\}$ endlich. Dann ist $\Gamma_2 := (\Gamma_1 \cup \Gamma_0) \setminus \{\neg A\} \subseteq \Gamma$ endlich und also auch erfüllbar. Es gibt somit eine Belegung φ, die Γ_2 und damit auch Γ_0 erfüllt. Dann ist aber

$\varphi(A) = 0$, d.h. $\varphi(\neg A) = 1$, und φ erfüllt $\Gamma_2 \cup \{\neg A\}$. Damit erfüllt φ aber auch Γ_1, eine beliebige endliche Teilmenge von $\Gamma \cup \{\neg A\}$, also ist $\Gamma \cup \{\neg A\}$ endlich erfüllbar.

Konstruiere eine <u>maximale</u> endlich erfüllbare Menge, die Σ enthält wie folgt: Sei $\{A_n \mid n \in \mathbb{N}\}$ eine Aufzählung der Aussageformen. Für $n \in \mathbb{N}$ definiere Δ_n durch $\Delta_0 := \Sigma$ und

$$\Delta_{n+1} \quad = \quad \begin{cases} \Delta_n \cup \{A_{n+1}\}\ , & \text{falls } \Delta_n \cup \{A_{n+1}\} \text{ endl. erfüllbar,} \\ \Delta_n \cup \{\neg A_{n+1}\}, & \text{sonst.} \end{cases}$$

Δ_n ist e.e. für alle $n \geq 0$. Sei $\Delta = \bigcup_{n\geq 0} \Delta_n$. Dann gilt: (i) $\Sigma \subseteq \Delta$, (ii) Δ ist e.e. und (iii) für jede Aussageform A gilt $A \in \Delta$ oder $\neg A \in \Delta$, wegen der endlichen Erfüllbarkeit von Δ jedoch nicht beides gleichzeitig. Definiere $\varphi : V \to \mathbb{B}$ durch

$$\varphi(p) = \begin{cases} 1, & \text{falls } p \in \Delta \\ 0, & \text{falls } \neg p \in \Delta. \end{cases}$$

φ ist eine wohldefinierte Belegung. Nach Lemma 1.5 lässt sich φ eindeutig zu einer Bewertung, die wir auch mit φ bezeichnen, fortsetzen. Durch Induktion über den Aufbau von F macht man sich schnell klar:

Für alle $A \in F$ gilt genau dann $\varphi(A) = 1$, wenn $A \in \Delta$ gilt. (Beispiel: Falls $\varphi(A \to B) = 1$, dann ist $\varphi(A) = 0$ oder $\varphi(B) = 1$, d.h. $\neg A \in \Delta$ oder $B \in \Delta$, also wurde $(A \to B)$ in Δ aufgenommen. Ist andererseits $\varphi(A \to B) = 0$, dann ist $\varphi(A) = 1$ und $\varphi(B) = 0$, d.h. $A \in \Delta$ und $\neg B \in \Delta$, also $\neg(A \to B) \in \Delta$ und wegen der endlichen Erfüllbarkeit von Δ ist $(A \to B) \notin \Delta$.) Also gilt auch $\varphi(A') = 1$ für alle $A' \in \Sigma$, also ist Σ erfüllbar. ■

1.11 Beispiel. Sei $\Sigma \subseteq F$. Gibt es zu jeder Bewertung φ ein $A \in \Sigma$ mit $\varphi(A) = 1$, so gibt es $A_1, \ldots, A_n \in \Sigma$ $(n > 0)$ mit $\models A_1 \vee \ldots \vee A_n$.

Betrachte die Menge $\Sigma' := \{\neg A \mid\ A \in \Sigma\}$. Nach Voraussetzung ist sie unerfüllbar. Also gibt es eine endliche nichtleere Teilmenge $\{\neg A_1, \ldots, \neg A_n\}$ von Σ', die unerfüllbar ist. Also gibt es für jede Bewertung φ ein i mit $\varphi(\neg A_i) = 0$. Dann ist aber $\varphi(A_i) = 1$ und daher auch $\varphi(A_1 \vee \ldots \vee A_n) = 1$.

1.12 Definition. $A, B \in F$ heißen *logisch äquivalent* mit der Schreibweise $A \models\!\!=\!\!| B$, falls für jede Bewertung φ gilt: $\varphi(A) = \varphi(B)$. Insbesondere ist dann $A \models B$ und $B \models A$.

Einige Beispiele für logisch äquivalente Formeln:

1. $A \models\!\!=\!\!| \neg(\neg A)$,
2. $A \wedge B \models\!\!=\!\!| B \wedge A$ und $A \vee B \models\!\!=\!\!| B \vee A$,
3. $A \wedge (B \wedge C) \models\!\!=\!\!| (A \wedge B) \wedge C$ und $A \vee (B \vee C) \models\!\!=\!\!| (A \vee B) \vee C$,
4. $A \wedge (B \vee C) \models\!\!=\!\!| (A \wedge B) \vee (A \wedge C)$ und
   $A \vee (B \wedge C) \models\!\!=\!\!| (A \vee B) \wedge (A \vee C)$
5. $\neg(A \wedge B) \models\!\!=\!\!| (\neg A) \vee (\neg B)$ und $\neg(A \vee B) \models\!\!=\!\!| (\neg A) \wedge (\neg B)$

6. $A \to B \models\!\!=\!\!\mid (\neg A) \vee B$, $A \wedge B \models\!\!=\!\!\mid \neg(A \to (\neg B))$ und $A \vee B \models\!\!=\!\!\mid (\neg A) \to B$.

Man beachte, dass $\models\!\!=\!\!\mid$ reflexiv, transitiv und symmetrisch, d.h. eine Äquivalenzrelation ist.
Nimmt man als zusätzlichen Junktor "$\leftrightarrow$" hinzu mit

$$\varphi(A \leftrightarrow B) = \begin{cases} 1, & \text{falls } \varphi(A) = \varphi(B) \\ 0, & \text{sonst,} \end{cases}$$

sind offenbar

- $\models (A \leftrightarrow B)$,
- $A \models\!\!=\!\!\mid B$,
- $A \models B$ und $B \models A$ und
- $\mathrm{Folg}(A) = \mathrm{Folg}(B)$

äquivalent.

1.13 Folgerung. Zu jedem $A \in F$ gibt es $B, C, D \in F$ mit

1. $A \models\!\!=\!\!\mid B$, B enthält nur $\to$ und $\neg$ als log. Verknüpfungen
2. $A \models\!\!=\!\!\mid C$, C enthält nur $\wedge$ und $\neg$ als log. Verknüpfungen
3. $A \models\!\!=\!\!\mid D$, D enthält nur $\vee$ und $\neg$ als log. Verknüpfungen

1.14 Definition. Eine Menge $\mathrm{OP} \subseteq \{\neg, \vee, \wedge, \to, \leftrightarrow, \ldots\}$ von Operatoren (Junktoren) heißt *vollständig,* falls es zu jedem $A \in F$ eine logisch äquivalente Aussageform $B \in F(\mathrm{OP})$ gibt.

Vollständige Operatormengen für die Aussagenlogik sind z.B.: $\{\neg, \to\}$, $\{\neg, \vee\}$, $\{\neg, \wedge\}$, $\{\neg, \vee, \wedge\}$ und $\{\mathrm{false}, \to\}$. Dabei ist false eine Konstante (nullstelliger Operator) mit $\varphi(\mathrm{false}) = 0$ für jede Bewertung φ. Offenbar gilt

$$\neg A \models\!\!=\!\!\mid (A \to \mathrm{false}).$$

Jede Aussageform $A(p_1, \ldots, p_n)$ stellt in natürlicher Form eine Bool'sche Funktion $f_A : \mathbb{B}^n \to \mathbb{B}$ dar, nämlich durch die Festlegung $f_A(b_1, \ldots, b_n) = \varphi_{\vec{b}}(A)$ mit einer Bewertung $\varphi_{\vec{b}}$ mit $\varphi_{\vec{b}}(p_i) = b_i$ für $1 \leq i \leq n$.

1.3 Das deduktive System $\mathcal{F}_0$

Bisher wurde die Semantik einer Sprache der Aussagenlogik mit Hilfe einer Bewertungsfunktion φ erklärt, die jeder syntaktisch korrekten Formel einen Wahrheitswert zuordnete. Der nächste Abschnitt beschäftigt sich mit einem axiomatischen Aufbau der Aussagenlogik mittels eines "Deduktiven Systems". Eine syntaktisch korrekte Formel in einem Deduktiven System wird "Theorem" genannt, wenn sie durch rein mechanische Anwendungen der Regeln des Systems auf die Axiome des Systems "abgeleitet" werden kann. Man kann deduktive Systeme angeben, in denen aussagenlogische Formeln genau dann Theoreme sind, wenn sie auch Tautologien sind.

1.15 Definition. Ein *Deduktives System* $\mathcal{F} = \mathcal{F}(\mathrm{Ax}, R)$ besteht aus

- einem Alphabet Δ (hier $\Delta = V \cup K \cup \{\rightarrow, \neg\}$),
- $F \subseteq \Delta^\star$, einer Menge von (wohldefinierten) Formeln (hier die Aussageformen),
- $\mathrm{Ax} \subseteq F$, einer Menge von Axiomen und
- R, einer Menge von Regeln der Form $\frac{A_1, \ldots, A_n}{A}$, $n \geq 0$.

Die Mengen F, Ax und R sind im allgemeinen rekursiv.
Die Menge $T = T(\mathcal{F})$ der *Theoreme* ist definiert durch:

1. $\mathrm{Ax} \in T$ (d.h. alle Axiome sind Theoreme)
2. Sind $A_1, \ldots, A_n \in T$ und ist die Regel $\frac{A_1, \ldots, A_n}{A}$ in R, dann ist $A \in T$.
3. T ist die kleinste Menge von Formeln, die 1. und 2. erfüllt.

Man schreibt für $A \in T(\mathcal{F})$ auch $\vdash_{\mathcal{F}} A$ oder einfach $\vdash A$ und sagt "A ist in $\mathcal{F}$ herleitbar".
Sei $\Sigma \subseteq F,\ A \in F$, dann bedeutet $\Sigma \vdash_{\mathcal{F}(Ax,R)} A$ nichts anderes als $\vdash_{\mathcal{F}(Ax \cup \Sigma, R)} A$.

1.16 Bemerkungen.

1. Eigenschaften der Elemente von T werden durch strukturelle Induktion bewiesen. T wird von einer Relation $R' \subseteq F^\star \times F$ erzeugt. Eine Formel A ist ein Theorem oder ist in $\mathcal{F}$ herleitbar, falls es eine endliche Folge von Formeln $B_0, \ldots, B_n$ gibt mit $A \equiv B_n$ und für $0 \leq i \leq n$ gilt: $B_i \in \mathrm{Ax}$ oder es gibt l und $i_1, \ldots, i_l < i$ und eine Regel $\frac{B_{i_1}, \ldots, B_{i_l}}{B_i} \in R$. Die Folge $B_0, \ldots, B_n$ heißt auch *Beweis* für A in $\mathcal{F}$. Das bedeutet $\vdash A$ gilt genau dann, wenn es einen Beweis $B_0, \ldots, B_n$ mit $A \equiv B_n$ gibt.

2. Die Menge T der Theoreme ist rekursiv aufzählbar (denn Ax und R sind rekursiv). Die Menge der Beweise

$$\mathrm{Bew} := \{B_1 \star B_2 \star \ldots \star B_n \mid B_1, \ldots, B_n \text{ ist Beweis}\}$$

ist rekursiv.

$T = \{A \mid \vdash A\} = \{A \mid \text{Es gibt Beweis } B_1, \dots, B_n \text{ mit } B_n = A\}$.

Beachte: Beweise sind im allgemeinen nicht eindeutig. Es wird im allgemeinen nicht verlangt, dass T von R frei erzeugt wird.

3. Im Zusammenhang mit Deduktiven Systemen liegt die Frage nach einer automatischen Beweisfindung nahe: Ist T rekursiv entscheidbar? Oder speziell: Gibt es ein deduktives System $\mathcal{F}_0$, so dass $\vdash_{\mathcal{F}_0} A$ genau dann gilt, wenn $\models A$ gilt? Hierzu werden Ax und R häufig endlich beschrieben durch Schemata. Beispielsweise beschreibt das Axiom $(A \to (B \to A))$ die Menge $\{A_0 \mid \text{es gibt } A, B \in F \text{ mit } A_0 \equiv (A \to (B \to A))\}$ und die Regel $\frac{A, A \to B}{B}$ beschreibt die Menge

$$\left\{ \frac{A_0, A_1}{B_0} \;\middle|\; \text{Es gibt } A, B \in F \text{ mit } A_0 \equiv A, B_0 \equiv B \text{ und } A_1 \equiv A \to B \right\}.$$

1.17 Definition. Das Deduktive System $\mathcal{F}_0$ für die Aussagenlogik besteht aus der Formelmenge F_0 der Formeln in $V, \neg, \to,$ (und). Die Axiomenmenge Ax wird durch folgende Axiomenschemata beschrieben:

Ax1: $A \to (B \to A)$

Ax2: $(A \to (B \to C)) \to ((A \to B) \to (A \to C))$

Ax3: $((\neg A) \to (\neg B)) \to (B \to A)$

Dabei beschreiben Ax1, Ax2 und Ax3 disjunkte Formelmengen. Die Regelmenge R wird beschrieben durch das Regelschema

$$\text{MP:} \quad \frac{A, A \to B}{B} \quad \text{(modus ponens)}.$$

Beachte: Es genügen Axiome für Formeln in $\to$ und $\neg$, da alle anderen Formeln zu einer Formel in $\to$ und $\neg$ logisch äquivalent sind ($\{\neg, \to\}$ ist vollständig):

$$\begin{aligned} A \wedge B &\models\!\!\dashv \neg(A \to (\neg B)) \\ A \vee B &\models\!\!\dashv (\neg A) \to B \\ A \leftrightarrow B &\models\!\!\dashv (A \to B) \wedge (B \to A) \end{aligned}$$

Die Menge der Theoreme von $\mathcal{F}_0$ wird nicht frei erzeugt. Die Modus-Ponens-Regel ist hochgradig nicht eindeutig. $\frac{A, A \to B}{B}$ und $\frac{A', A' \to B}{B}$ sind beides Regeln mit gleicher Folgerung. Das erschwert sehr das Finden von Beweisen.

Beispiel: Für jedes $A \in F_0$ gilt $\vdash (A \to A)$.

Beweis.

$B_0 \equiv (A \to ((A \to A) \to A)) \to ((A \to (A \to A)) \to (A \to A))$	Ax2	
$B_1 \equiv A \to ((A \to A) \to A)$	Ax1	
$B_2 \equiv (A \to (A \to A)) \to (A \to A)$	MP(B_0, B_1)	■
$B_3 \equiv A \to (A \to A)$	Ax1	
$B_4 \equiv A \to A$	MP(B_2, B_3)	

1.18 Definition (Axiomatischer Folgerungsbegriff)**.** Sei $\Sigma \subseteq F_0, A \in F_0$.

1. A ist aus Σ in $\mathcal{F}_0$ *herleitbar*, wenn A sich aus $\mathrm{Ax} \cup \Sigma$ mit den Regeln aus R herleiten lässt, d.h. A ist Theorem im Deduktiven System $\mathcal{F}$ mit Axiomenmenge $\mathrm{Ax} \cup \Sigma$ und gleicher Regelmenge wie $\mathcal{F}_0$. Schreibweise $\Sigma \vdash_{\mathcal{F}_0} A$ oder einfach $\Sigma \vdash A$.

   $B_0, \ldots, B_n$ ist ein *Beweis* für $\Sigma \vdash A$, falls $A \equiv B_n$ und für alle $0 \leq i \leq n$ gilt: $B_i \in \mathrm{Ax} \cup \Sigma$ oder es gibt $j, k < i$ mit $B_k \equiv B_j \to B_i$.

2. Σ heißt *konsistent*, falls für keine Formel $A \in F_0$ gilt $\Sigma \vdash A$ und $\Sigma \vdash \neg A$.

1.19 Folgerungen und Bemerkungen.

1. Gilt $\Sigma \vdash A$, so folgt unmittelbar aus der Definition 1.18, dass es eine endliche Teilmenge $\Sigma_0 \subseteq \Sigma$ gibt mit $\Sigma_0 \vdash A$. Das entspricht dem Kompaktheitssatz für "$\models$".

2. Ist Σ inkonsistent, dann gibt es eine endliche Teilmenge $\Sigma_0 \subseteq \Sigma$, die inkonsistent ist (denn ist $\Sigma \subseteq \Gamma$ und $\Sigma \vdash A$, dann gilt auch $\Gamma \vdash A$).

3. Aus $\Sigma \vdash A$ und $\Gamma \vdash B$ für alle $B \in \Sigma$ folgt $\Gamma \vdash A$. Gilt $\Gamma \vdash \Sigma$ (d.h. $\Gamma \vdash A$ für alle $A \in \Sigma$) und $\Sigma \vdash \Delta$, dann gilt $\Gamma \vdash \Delta$. Beweise lassen sich also zusammensetzen.

1.20 Satz (Deduktionstheorem)**.** Sei $\Sigma \subseteq F_0$ und seien $A, B \in F_0$.
Es gilt genau dann $\Sigma, A \vdash B$, wenn $\Sigma \vdash (A \to B)$ gilt.

Beweis.
"$\Longleftarrow$": klar wegen Modus Ponens.
"$\Longrightarrow$": Sei $B_0, \ldots, B_m$ ein Beweis für $\Sigma, A \vdash B$, also $B_m \equiv B$. Für $i = 0, \ldots, m$ gilt $\Sigma \vdash (A \to B_i)$. $(\star)$.
Aus $(\star)$ folgt die Behauptung.
Beweis von $(\star)$:

1. $B_i \equiv A$: Nach dem Beispiel zur Definition 1.17 gilt dann $\vdash A \to B_i$ und somit aber auch $\Sigma \vdash A \to B_i$.

2. $B_i \in \mathrm{Ax} \cup \Sigma$: dann gilt $\Sigma \vdash (A \to B_i)$ wegen der Beweissequenz:

B_i	Axiom ($B_i \in \mathrm{Ax}$) oder Hypothese ($B_i \in \Sigma$)
$B_i \to (A \to B_i)$	Axiom
$A \to B_i$	MP

3. B_i entsteht aus B_j, B_k mit MP, d.h. $B_k \equiv B_j \to B_i$ mit $j, k < i$, und $\Sigma \vdash A \to B_j$ und $\Sigma \vdash A \to (B_j \to B_i)$ sind bereits bekannt, dann ist ein Beweis für $\Sigma \vdash A \to B_i$:

$(A \to (B_j \to B_i)) \to ((A \to B_j) \to (A \to B_i))$	Ax2
$\vdots$ (Beweis für $\Sigma \vdash A \to (B_j \to B_i)$)	
$(A \to B_j) \to (A \to B_i)$	MP
$\vdots$ (Beweis für $\Sigma \vdash A \to B_j$)	
$A \to B_i$	MP

■

1.21 Beispiele. $B_1, \ldots, B_n$ heißt *abgekürzter Beweis* für $\Sigma \vdash B_n$, falls für jedes j mit $1 \leq j \leq n$ gilt: $\Sigma \vdash B_j$ oder es gibt $j_1, \ldots, j_r < j$ mit $B_{j_1}, \ldots, B_{j_r} \vdash B_j$. Gibt es einen abgekürzten Beweis für $\Sigma \vdash A$, dann gibt es auch einen Beweis für $\Sigma \vdash A$.

1. $\vdash (A \to A)$ folgt aus dem Deduktionstheorem, da $A \vdash A$ gilt.

2. Um $A \to B, B \to C \vdash A \to C$ zu zeigen, zeige $A, A \to B, B \to C \vdash C$.

3. $\neg\neg A \vdash A$
   Beweis:

$B_1 \equiv \neg\neg A$	Hypothese
$B_2 \equiv \neg\neg A \to (\neg\neg\neg\neg A \to \neg\neg A)$	Ax1
$B_3 \equiv \neg\neg\neg\neg A \to \neg\neg A$	MP(B_1, B_2)
$B_4 \equiv (\neg\neg\neg\neg A \to \neg\neg A) \to (\neg A \to \neg\neg\neg A)$	Ax3
$B_5 \equiv \neg A \to \neg\neg\neg A$	MP(B_3, B_4)
$B_6 \equiv (\neg A \to \neg\neg\neg A) \to (\neg\neg A \to A)$	Ax3
$B_7 \equiv \neg\neg A \to A$	MP(B_5, B_6)
$B_8 \equiv A$	MP(B_1, B_7)

4. $\vdash (A \to B) \to ((B \to C) \to (A \to C))$
   (zeige: $A \to B, B \to C \vdash A \to C$)

5. $\vdash B \to ((B \to A) \to A)$

6. $\vdash \neg B \to (B \to A)$ (zu zeigen: $\neg B, B \vdash A$)
   Beweis:

$B_1 \equiv \neg B$	Hypothese
$B_2 \equiv \neg B \to (\neg A \to \neg B)$	Ax1
$B_3 \equiv \neg A \to \neg B$	MP(B_1, B_2)
$B_4 \equiv (\neg A \to \neg B) \to (B \to A)$	Ax3
$B_5 \equiv B \to A$	MP(B_3, B_4)
$B_6 \equiv B$	Hypothese
$B_7 \equiv A$	MP(B_6, B_5)

7. $\vdash B \to \neg\neg B$

8. $\vdash (A \to B) \to (\neg B \to \neg A)$ und $\vdash (\neg B \to \neg A) \to (A \to B)$

9. $\vdash B \to (\neg C \to \neg(B \to C))$

10. $\vdash (B \to A) \to ((\neg B \to A) \to A)$

11. $\vdash (A \to B) \to ((A \to \neg B) \to \neg A)$

1.22 Satz (Korrektheit und Vollständigkeit von $\mathcal{F}_0$). Sei $A \in F_0$ eine Formel der Aussagenlogik.

a) Korrektheit: Aus $\vdash_{\mathcal{F}_0} A$ folgt $\models A$, d.h. nur Tautologien können als Theoreme in $\mathcal{F}_0$ hergeleitet werden.

b) Vollständigkeit: Aus $\models A$ folgt $\vdash_{\mathcal{F}_0} A$, d.h. alle Tautologien lassen sich in $\mathcal{F}_0$ herleiten.

Beweis.

Korrektheit: Alle Axiome (Schemata) sind Tautologien. Sind A und $A \to B$ Tautologien, so ist auch B eine Tautologie, d.h. Tautologien sind abgeschlossen gegenüber Modus-Ponens-Regel (s. Lemma 1.9).

Vollständigkeit: Zeige: Gilt $\models A$ dann ist A in $\mathcal{F}_0$ herleitbar, d.h. $\vdash A$.

Lemma: Sei $A \equiv A(p_1, \dots, p_n) \in F_0,\ n > 0$, wobei $p_1, \dots, p_n$ die in A vorkommenden Aussagevariablen sind. Sei φ eine Bewertung. Ist

$$P_i := \begin{cases} p_i, & \text{falls } \varphi(p_i) = 1 \\ \neg p_i, & \text{falls } \varphi(p_i) = 0 \end{cases} \qquad \text{und} \qquad A' := \begin{cases} A, & \text{falls } \varphi(A) = 1 \\ \neg A, & \text{falls } \varphi(A) = 0 \end{cases}$$

$(1 \leq i \leq n)$, dann gilt $P_1, \dots, P_n \vdash A'$.

Angenommen das Lemma gilt und sei $\models A$, d.h. $\varphi(A) = 1$ für alle Bewertungen φ.

Sei φ eine Bewertung mit $\varphi(p_n) = 1$. Es gilt $P_1, \dots, P_n \vdash A$ und wegen $P_n \equiv p_n$ gilt $P_1, \dots, P_{n-1}, p_n \vdash A$. Betrachtet man eine Bewertung φ' mit $\varphi'(p_n) = 0$, erhält man $P_1, \dots, P_{n-1}, \neg p_n \vdash A$.

Durch Anwenden des Deduktionstheorems entstehen daraus

$$P_1, \dots, P_{n-1} \vdash p_n \to A \text{ und } P_1, \dots, P_{n-1} \vdash (\neg p_n) \to A.$$

Gleichzeitig gilt nach dem 10. Beispiel von 1.21 auch

$$P_1, \dots, P_{n-1} \vdash ((p_n \to A) \to ((\neg p_n \to A) \to A)).$$

Durch zweimaliges Anwenden des Modus-Ponens entsteht

$$P_1, \dots, P_{n-1} \vdash A.$$

Das kann man für jedes i machen, d.h. in der Herleitung von A wird kein p_i verwendet, also $\vdash A$.

Beweis des Lemma. Induktion über den Aufbau der Formel A:

$A \equiv p_1 \in V$: $\varphi(A) = \varphi(p_1)$ und folglich $p_1 \vdash A'$ wegen $A' = A$.

$A \equiv \neg C$: Ist $\varphi(A) = 1$, dann ist $\varphi(C) = 0$ und $C' \equiv \neg C$. Nach Induktionsvoraussetzung gilt $P_1, \ldots, P_n \vdash C'$ mit $C' \equiv \neg C \equiv A$.

Ist $\varphi(A) = 0$, dann ist $\varphi(C) = 1$, $C' \equiv C$ und $A' \equiv \neg A \equiv \neg\neg C$. Nach Induktionsvoraussetzung gilt $P_1, \ldots, P_n \vdash C'$ und nach dem 7. Beispiel von 1.21 gilt $P_1, \ldots, P_n \vdash (C \to \neg\neg C)$. Durch Anwenden der Modus-Ponens Regel erhält man $P_1, \ldots, P_n \vdash \neg\neg C$.

$A \equiv B \to C$: Ist $\varphi(A) = 0$, dann ist $\varphi(B) = 1$ und $\varphi(C) = 0$. Ferner gilt dann $A' \equiv \neg A \equiv \neg(B \to C)$, $B' \equiv B$ und $C' \equiv \neg C$. Nach Induktionsvoraussetzung gilt $P_1, \ldots, P_n \vdash B'$ und $P_1, \ldots, P_n \vdash C'$. Zusammen mit $P_1, \ldots, P_n \vdash B \to (\neg C \to \neg(B \to C))$ von 9. Beispiel in 1.21 und zweimaligem Anwenden des MP folgt $P_1, \ldots, P_n \vdash A'$.

Ist $\varphi(A) = 1$ dann ist $\varphi(C) = 1$ oder $\varphi(B) = 0$. Ferner gilt dann $A' \equiv B \to C$. Nach Induktionsvoraussetzung gilt $P_1, \ldots, P_n \vdash C$ oder $P_1, \ldots, P_n \vdash \neg B$. Gilt $P_1, \ldots, P_n, B \vdash C$, folgt nach Deduktionstheorem auch $P_1, \ldots, P_n \vdash B \to C$. Ist andererseits $P_1, \ldots, P_n, \neg C \vdash \neg B$ und somit $P_1, \ldots, P_n \vdash (\neg C \to \neg B)$. Nach dem 8. Beispiel von 1.21 gilt dann $P_1, \ldots, P_n \vdash B \to C$.

∎

1.23 Folgerung. Sei $\Sigma \subseteq F_0, A \in F_0$.

1. $\Sigma \vdash_{\mathcal{F}_0} A$ gilt genau dann, wenn $\Sigma \models A$ gilt.
2. Σ ist genau dann konsistent, wenn Σ erfüllbar ist.

Beweis.

1.

$$\begin{aligned}
\Sigma \vdash A &\overset{1.19}{\iff} \text{Es gibt } A_1, \ldots, A_n \in \Sigma \text{ mit } A_1, \ldots, A_n \vdash_{\mathcal{F}_0} A \\
&\overset{\text{D.T.}}{\iff} \text{Es gibt } A_1, \ldots, A_n \in \Sigma \text{ mit } \vdash_{\mathcal{F}_0} (A_1 \to (A_2 \to \ldots (A_n \to A) \ldots) \\
&\overset{1.22}{\iff} \text{Es gibt } A_1, \ldots, A_n \in \Sigma \text{ mit } \models (A_1 \to (A_2 \to \ldots (A_n \to A) \ldots) \\
&\overset{\text{D.T.}}{\iff} \text{Es gibt } A_1, \ldots, A_n \in \Sigma \text{ mit } A_1, \ldots, A_n \models A \\
&\overset{\text{K.S.}}{\iff} \Sigma \models A
\end{aligned}$$

2. Σ ist konsistent. $\iff$
Es gibt kein A mit $\Sigma \vdash A$ und $\Sigma \vdash \neg A$. $\iff$
Es gibt kein A mit $\Sigma \models A$ und $\Sigma \models \neg A$. $\iff$
Σ ist erfüllbar.

∎

1.4 Das Gentzen-Sequenzenkalkül

Es gibt andere deduktive Systeme, für die Satz 1.22 gilt. Das deduktive System $\mathcal{F}_0$ wurde von S.C. Kleene eingeführt. Das folgende deduktive System geht auf G. Gentzen zurück.

1.24 Definition. Das *Gentzen-Sequenzenkalkül*: Eine Sequenz ist eine Zeichenreihe der Form $\Gamma \Rightarrow \Delta$ mit zwei endlichen Mengen von Formeln Γ und Δ. Seien $\Gamma, \Delta \subseteq F$ endliche Mengen von Formeln und $A, B \in F$.
$\Gamma \vdash_G \Delta$ wird definiert durch:

die Axiome Ax1: $\Gamma, A \Rightarrow A, \Delta$

Ax2: $\Gamma, A, \neg A \Rightarrow \Delta$

Ax3: $\Gamma \Rightarrow A, \neg A, \Delta$

die Regeln R1: $\dfrac{\Gamma, A, B \Rightarrow \Delta}{\Gamma, A \wedge B \Rightarrow \Delta}$

R2: $\dfrac{\Gamma \Rightarrow A, B, \Delta}{\Gamma \Rightarrow (A \vee B), \Delta}$

R3: $\dfrac{\Gamma, A \Rightarrow \Delta, B}{\Gamma \Rightarrow (A \rightarrow B), \Delta}$

R4a: $\dfrac{\Gamma, A \Rightarrow \Delta}{\Gamma \Rightarrow \neg A, \Delta}$

R4b: $\dfrac{\Gamma \Rightarrow A, \Delta}{\Gamma, \neg A \Rightarrow \Delta}$

R5: $\dfrac{\Gamma \Rightarrow A, \Delta; \quad \Gamma \Rightarrow B, \Delta}{\Gamma \Rightarrow (A \wedge B), \Delta}$

R6: $\dfrac{\Gamma, A \Rightarrow \Delta; \quad \Gamma, B \Rightarrow \Delta}{\Gamma, (A \vee B) \Rightarrow \Delta}$

R7: $\dfrac{\Gamma \Rightarrow A, \Delta; \quad \Gamma, B \Rightarrow \Delta}{\Gamma, (A \rightarrow B) \Rightarrow \Delta}$

$\Gamma \vdash_G \Delta$ bedeutet, dass es ein $r \in \mathbb{N}$ und eine Folge von Sequenzen

$$\Gamma_1 \Rightarrow \Delta_1, \ldots, \Gamma_r \Rightarrow \Delta_r$$

gibt, für die folgendes gilt:

1. $\Gamma_r \equiv \Gamma$ und $\Delta_r \equiv \Delta$.
2. Jedes $\Gamma_j \rightarrow \Delta_j$ mit $1 \leq j \leq r$ ist Axiom oder geht aus vorangehenden Folgegliedern aufgrund einer Regel hervor.

1.25 Bemerkung. Die Aussage $\Gamma \vdash_G \Delta$ kann wie folgt anschaulich interpretiert werden: Für jede Bewertung φ gibt es eine Formel $A \in \Gamma$ mit $\varphi(A) = 0$ oder es gibt eine Formel $B \in \Delta$ mit $\varphi(B) = 1$. Sind $\Gamma = \{A_1, \ldots, A_n\}$ und $\Delta = \{B_1, \ldots, B_m\}$ entspricht dies also der Formel $A_1 \wedge \cdots \wedge A_n \rightarrow (B_1 \vee \cdots \vee B_m)$.

1.26 Beispiel. $p \vee q, (\neg p) \vee r \vdash_G q \vee r$

Beweis:

$B_1 \equiv q, r \Rightarrow q, r$	Ax1
$B_2 \equiv q, \neg p \Rightarrow q, r$	Ax1
$B_3 \equiv q, (\neg p) \vee r \Rightarrow q, r$	R6(1,2)
$B_4 \equiv p, r \Rightarrow q, r$	Ax1
$B_5 \equiv \neg p, p \Rightarrow q, r$	Ax2
$B_6 \equiv p, (\neg p) \vee r \Rightarrow q, r$	R6(4,5)
$B_7 \equiv p \vee q, (\neg p) \vee r \Rightarrow q, r$	R6(3,6)
$B_8 \equiv p \vee q, (\neg p) \vee r \Rightarrow q \vee r$	R2(6)

1.27 Bemerkung.

1. Aus $\Gamma \vdash_G \Delta$ folgt $\Gamma \models \Delta$. (Korrektheit)
2. Aus $\Gamma \models A$ folgt $\Gamma \vdash_G A$. (Vollständigkeit)

1.5 Semantische Tableaux

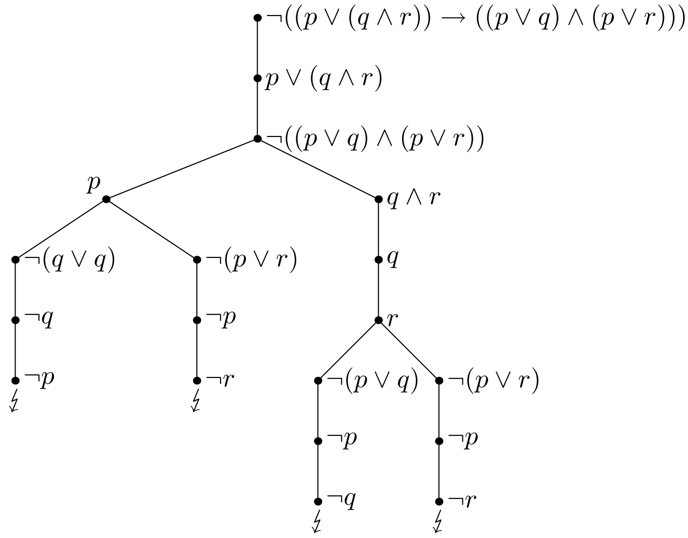
Wie erreicht man eine Systematisierung des Beweisens in einem deduktiven System? Gibt es eine Möglichkeit "automatisch" einen Beweis zu finden, ohne dazu alle Beweise der Reihe nach aufzuzählen?

Nach einer denotationalen und axiomatischen Definition der Semantik der Aussagenlogik wird im folgenden ein algorithmischer Aufbau betrachtet: Um zu entscheiden, ob $\Sigma \vdash A$ für eine gegebene (endliche) Formelmenge Σ und eine Aussageform A gilt, wird ein Algorithmus angegeben. Da in Σ und A nur endlich viele Variablen vorkommen, braucht der Algorithmus nur systematisch alle möglichen Belegungen zu prüfen.

1.28 Beispiel. Mit der Idee "$\Sigma \vdash A \iff \Sigma \cup \{\neg A\}$ ist nicht erfüllbar" versucht man, um die Allgemeingültigkeit einer Formel A zu zeigen, einen binären Baum für $\neg A$ zu konstruieren, dessen Knoten jeweils eine Klasse möglicher Belegungen repräsentieren. Die Wurzel des Baumes repräsentiert alle möglichen Belegungen und die Vereinigung der Klassen der Söhne eines inneren Knotens des Baumes ist die Klasse der Belegungen, die der Knoten repräsentiert. Gelingt es, einen solchen Baum derart zu konstruieren, dass sämtliche Blätter des Baumes zu einem Widerspruch zu der Annahme führen, eine Klasse von Belegungen, die durch Knoten auf dem Weg von der Wurzel zu dem entsprechenden Blatt repräsentiert wird, enthalte Belegungen, die $\neg A$ erfüllen, ist gezeigt, dass es keine Belegung gibt, die $\neg A$ erfüllt. Somit gilt, dass A Tautologie ist.

$\models (p \vee (q \wedge r) \rightarrow (p \vee q) \wedge (p \vee r))$ gilt genau dann, wenn $\neg((p \vee (q \wedge r)) \rightarrow ((p \vee q) \wedge (p \vee r)))$

unerfüllbar ist.



Da alle Äste zu Widersprüchen führen, gibt es keine Belegung, die die Formel erfüllt!

Man unterscheidet zwei Arten von Formeln, nämlich solche, die zu Verzweigungen führen (β-Formeln), und solche, die nicht zu Verzweigungen führen (α-Formeln).

Folgende Formeln sind $\underline{\alpha\text{-Formeln}}$, die zu Knoten mit den Markierungen α_1 und α_2 führen:

α	$\neg\neg A$	$A_1 \wedge A_2$	$\neg(A_1 \vee A_2)$	$\neg(A_1 \rightarrow A_2)$
α_1	A	A_1	$\neg A_1$	A_1
α_2	(A)	A_2	$\neg A_2$	$\neg A_2$

Folgende Formeln sind $\underline{\beta\text{-Formeln}}$, die zu Verzweigungen führen mit Knotenmarkierungen β_1 und

β		$\neg(A_1 \wedge A_2)$		$A_1 \vee A_2$		$A_1 \rightarrow A_2$	
β_1	β_2	$\neg A_1$	$\neg A_2$	A_1	A_2	$\neg A_1$	A_2

Beachte: Jede Aussageform ist entweder atomar (d.h. eine Variable) oder die Negation einer atomaren Formel oder eine α- oder eine β-Formel, und genau von einem dieser Typen.

Es gilt: Eine α-Formel ist genau dann erfüllbar, wenn beide Komponenten α_1 und α_2 erfüllbar sind. Eine β-Formel ist genau dann erfüllbar, wenn eine der Komponenten β_1 oder β_2 erfüllbar ist.

Insbesondere gilt für $\Gamma \subseteq F$ und α-Formel α mit Komponenten α_1 und α_2 und β-Formel β mit Komponenten β_1 und β_2: Genau dann ist $\Gamma \cup \{\alpha\}$ erfüllbar, wenn $\Gamma \cup \{\alpha_1, \alpha_2\}$ erfüllbar ist und genau dann ist $\Gamma \cup \{\beta\}$ erfüllbar, wenn $\Gamma \cup \{\beta_1\}$ oder $\Gamma \cup \{\beta_2\}$ erfüllbar ist.

1.29 Definition. *Tableaux* sind binäre Bäume mit Knoten, die mit Formeln aus F markiert sind. Sei $\Sigma \subseteq F$.

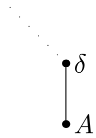
1. Die Menge der Tableaux τ_Σ für Σ wird induktiv definiert durch:

   a) $\tau_{\{A\}}$ ist der Baum mit einem Knoten, der mit A markiert ist. In diesem Fall schreibt man auch τ_A statt $\tau_{\{A\}}$. Graphisch:

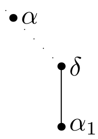
   $$\bullet A$$

   b) Ist τ Tableau für Σ und δ Marke eines Blattes von τ, so lässt sich τ wie folgt zu einem Tableau τ' für Σ fortsetzen: τ' entsteht aus τ indem man als Nachfolger von δ:

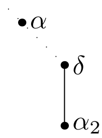
   (Σ) einen Knoten hinzufügt, der mit einer Formel $A \in \Sigma$ markiert ist. (A soll nicht bereits als Marke im Ast von δ vorkommen.) Graphisch:



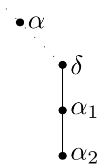
   (α) einen Knoten hinzufügt, der mit α_1 oder α_2 markiert ist, falls eine α-Formel α auf dem Ast zu δ vorkommt und α_1 und α_2 die Komponenten von α sind. Graphisch:



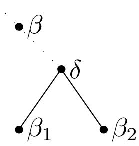
oder



   In der Praxis werden jedoch δ nacheinander Knoten für beide Komponenten hinzugefügt:



   (β) zwei Knoten hinzufügt, die mit den Komponenten β_1 bzw. β_2 einer β-Formel β markiert sind, falls β auf dem Ast zu δ vorkommt. Graphisch:



   Entsteht τ' aus τ durch Anwendung einer der Regeln $(\Sigma), (\alpha)$ oder (β), so heißt τ' *direkte Fortsetzung* von τ.

c) $\tau \in \tau_\Sigma$ genau dann, wenn $\tau = \tau_A$ für ein $A \in \Sigma$ oder es gibt eine Folge $\tau_0, \dots, \tau_n (= \tau)$, $n \in \mathbb{N}$, so dass τ_{j+1} eine direkte Fortsetzung von τ_j ist für $j = 0, \dots, n-1$ und $\tau_0 = \tau_A$ für ein $A \in \Sigma$.

2. Ein *Ast* eines Tableaus τ heißt *abgeschlossen*, falls er zwei konjugierte Formeln enthält (d.h. für ein $A \in F$ sowohl A als auch $(\neg A)$ enthält), sonst heißt der Ast *offen*. Ein Tableau τ heißt *abgeschlossen*, wenn jeder Ast von τ abgeschlossen ist. τ heißt *erfüllbar*, wenn τ einen *erfüllbaren Ast* (d.h. die Marken entlang des Ast bilden eine erfüllbare Formelmenge) enthält.

3. Sei $\Gamma \subseteq F, A \in F$. Dann ist A Tableau-Folgerung aus Γ ($\Gamma \vdash_\tau A$) genau dann, wenn für $\Sigma = \Gamma \cup \{\neg A\}$ jedes Tableau aus τ_Σ sich zu einem abgeschlossenen Tableau aus τ_Σ fortsetzen lässt.

1.30 Bemerkungen und Beispiele. Ziel ist es zu zeigen: $\Gamma \vdash_\tau A \iff \Gamma \models A$.

1. Abgeschlossene Äste sind nicht erfüllbar. Abgeschlossene Tableaux sind nicht erfüllbar.

2. Ist Γ erfüllbar, so ist jedes Tableau aus τ_Γ erfüllbar (und insbesondere nicht abgeschlossen).

3. Gilt $\Gamma \vdash_\tau A$, so ist $\Sigma = \Gamma \cup \{\neg A\}$ nicht erfüllbar. Insbesondere sind Tableau-Folgerungen korrekt (d.h. aus $\Gamma \vdash_\tau A$ folgt $\Gamma \models A$).

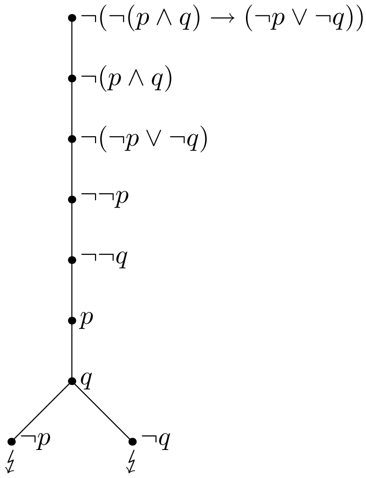
4. Gibt es ein abgeschlossenes Tableau in τ_Γ, so lässt sich jedes Tableau aus τ_Γ zu einem abgeschlossenen Tableau fortsetzen.

5. Beispiele:

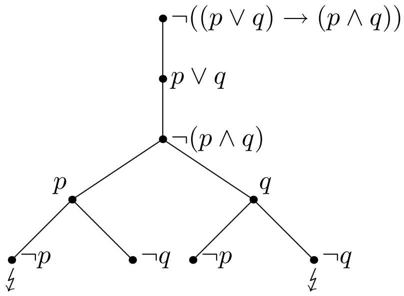
   a) $\vdash_\tau A \to (B \to A)$:

$$
\begin{array}{l}
\bullet\ \neg(A \to (B \to A)) \\
\bullet\ A \\
\bullet\ \neg(B \to A) \\
\bullet\ B \\
\bullet\ \neg A \\
\lightning
\end{array}
$$

b) $\vdash_\tau \neg(p \wedge q) \to (\neg p \vee \neg q)$:



c) $\vdash_\tau (p \vee q) \to (p \wedge q)$ gilt <u>nicht</u>:



Es gibt Belegungen, die $\neg((p \vee q) \to (p \wedge q))$ erfüllen, nämlich φ mit $\varphi(p) = 1$ und $\varphi(q) = 0$ und φ' $\varphi'(p) = 0$ und $\varphi'(q) = 1$. Also gilt nicht $\vdash_\tau (p \vee q) \to (p \wedge q)$.

1.31 Definition. Sei $\Sigma \subseteq F, \tau \in \tau_\Sigma$ und σ ein Ast von τ, bzw. die Menge der Formeln, die in dem Ast vorkommen. σ heißt *vollständig*, falls für jede α-Formel $\alpha \in \sigma$ stets $\{\alpha_1, \alpha_2\} \subseteq \sigma$ gilt und für jede β-Formel $\beta \in \sigma$ entweder $\beta_1 \in \sigma$ oder $\beta_2 \in \sigma$ gilt. τ heißt *vollständig*, falls gilt: Jeder Ast von τ ist entweder abgeschlossen oder "vollständig und enthält Σ". (Beachte: Falls Σ unendlich ist, sind unendliche Bäume zugelassen.)

1.32 Bemerkungen.

1. Ist Σ endlich, so lässt sich jedes Tableau aus τ_Σ zu einem vollständigen Tableau für Σ mit Hilfe von Σ-, α- und β-Regeln erweitern. Beachte, dass α- und β- Regeln nur Teilformeln einführen und dass eine Formel nur endlich viele Teilformeln enthalten kann.

2. Sei σ die Menge der Formeln eines <u>vollständigen offenen</u> Astes von τ. Dann gilt:

a) Es gibt kein $p \in V$ mit $\{p, \neg p\} \subseteq \sigma$.

b) Ist $\alpha \in \sigma$, so auch $\alpha_1, \alpha_2 \in \sigma$.

c) Ist $\beta \in \sigma$, so ist $\beta_1 \in \sigma$ oder $\beta_2 \in \sigma$.

1.33 Lemma. Jede Menge Σ von Formeln, die (a), (b) und (c) aus der Bemerkung 1.32.2 genügt, ist erfüllbar. Insbesondere sind vollständige offene Äste von Tableaux erfüllbar.

Beweis. Definiere:

$$\varphi(p) = \begin{cases} 0, & \neg p \in \Sigma \\ 1, & \text{sonst} \end{cases}$$

Offensichtlich ist φ wohldefiniert.

Beh.: Falls $A \in \Sigma$, dann $\varphi(A) = 1$.

Induktion über den Aufbau von A :

A atomare oder negierte atomare Formel $\surd$

Sei A eine α-Formel und $A \in \Sigma$, dann sind nach 1.32.2.a die Teilformeln α_1 und α_2 von A in Σ. Nach Induktionsvoraussetzung gilt $\varphi(\alpha_1) = \varphi(\alpha_2) = 1$ woraus $\varphi(A) = 1$ folgt.

Sei A eine β-Formel und $A \in \Sigma$. Dann ist die Teilformel β_1 oder β_2 nach 1.32.2.c in Σ. Nach Induktionsvoraussetzung gilt $\varphi(\beta_1) = 1$ oder $\varphi(\beta_2) = 1$. Somit folgt $\varphi(A) = 1$. ■

1.34 Satz. Sei $\Gamma \subseteq F$. Dann gilt:

1. Genau dann ist Γ nicht erfüllbar, wenn τ_Γ ein abgeschlossenes Tableau enthält.

2. Äquivalent sind

   - $\Gamma \models A$,
   - $\Gamma \vdash A$ und
   - $\tau_{\{\Gamma, \neg A\}}$ enthält ein abgeschlossenes Tableau.

3. Äquivalent sind

   - $\models A$,
   - $\vdash A$ und
   - $\tau_{\neg A}$ enthält ein abgeschlossenes Tableau.

Beachte: Der Kompaktheitssatz (1.10) folgt aus 1., denn ist Γ nicht erfüllbar, enthält τ_Γ ein abgeschlossenes Tableau und abgeschlossene Tableaux sind stets endliche Bäume, d.h. eine endliche Teilmenge von Γ ist nicht erfüllbar.

Beweis. zu 1.

“$\Longleftarrow$”: siehe 1.30.2.

“$\Longrightarrow$”: Jedes vollständige Tableau τ für Γ ist abgeschlossen, sonst enthält τ einen vollständigen offenen Ast, der ganz Γ enthält und somit wäre Γ nach 1.33 erfüllbar. ■

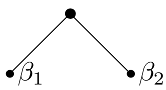
Gibt es immer vollständige Tableaux in τ_Γ für $\Gamma \subseteq F$? Falls Γ endlich ist, dann siehe 1.32.1. Was sind aber vollständige Tableaux, wenn Γ unendlich ist? Dazu zunächst ein systematisches Verfahren zur Tableaukonstruktion.

Systematische Tableaukonstruktion: Sei $\Gamma \subseteq F$, dann ist Γ abzählbar. Sei also $\Gamma = \{A_1, A_2, \ldots\}$. Konstruktion einer Folge von Tableaux τ_n $(n \in \mathbb{N})$:

1. $\tau_1 \equiv A_1$. Ist A_1 (negierte) atomare Formel, dann wird der Knoten markiert.
2. Sind alle Äste von τ_n abgeschlossen, dann Stopp!
3. τ_{n+1} entsteht aus τ_n wie folgt:

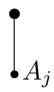
   a) Ist Y die erste unmarkierte α-Formel in τ_n, durch die ein offener Ast geht, so markiere Y und erweitere jeden offenen Ast, der durch Y geht, um die Teilformeln α_1 und α_2 von Y. α_1 und α_2 werden markiert, falls sie atomare oder negierte atomare Formeln sind. Dadurch werden möglicherweise Äste abgeschlossen. sonst:

   b) Ist Y die erste unmarkierte β-Formel in τ_n, durch die ein offener Ast geht, so markiere Y und erweitere jeden offenen Ast, der durch Y geht, um

   Markiere β_1 und/oder β_2, falls diese (negierte) atomare Formeln sind. Dadurch werden möglicherweise Äste abgeschlossen. sonst:

   c) Gibt es eine Formel $A_j \in \Gamma$, die noch nicht in jedem offenen Ast vorkommt, so erweitere alle diese Äste um:

   Falls möglich, Knoten markieren und Äste abschließen.

   Gehe zu Schritt (2).

Aus τ_n $(n \geq 1)$ erhält man kein weiteres Tableau, falls τ_n abgeschlossen ist oder alle Formeln von τ_n markiert sind und Γ endlich ist.

Setze $\tau_\infty := \bigcup_{n \in \mathbb{N}} \tau_n$. Dann ist τ_∞ ein binärer Baum. Behauptung: τ_∞ ist vollständig! Beweis:

1. $\tau_\infty = \tau_k$ für ein $k \in \mathbb{N}$.

   Ist τ_k abgeschlossen, gilt die Behauptung.

   Ist τ_k nicht abgeschlossen, ist τ_k vollständig: Alle Formeln sind markiert und Γ muss endlich sein. Alle Formeln von Γ sind in offenen Ästen von τ_k. Somit ist Γ nach Lemma 1.33 erfüllbar.

2. Es gibt kein $k \in \mathbb{N}$ mit $\tau_\infty = \tau_k$. Dann ist τ_∞ ein unendlicher Baum. Es gibt eine Folge von Knoten $\{Y_n\}, n \in \mathbb{N}$, die unendlich viele Nachfolger haben: Setze $Y_1 = A_1$, die Wurzel mit unendlich vielen Nachfolgerknoten. Ist Y_n bereits gefunden, dann hat Y_n entweder einen oder zwei direkte Nachfolger, von denen einer unendlich viele Nachfolger hat. Wähle als Y_{n+1} diesen Knoten. Dann ist der Ast $\{Y_n | n \in \mathbb{N}\}$ in τ_∞, offen, vollständig und enthält Γ, d.h. Γ ist erfüllbar.

1.35 Bemerkung und Folgerung.

1. Ist Γ eine rekursiv aufzählbare Menge, so ist das Hinzufügen einer Formel $A_n \in \Gamma$ zu einem Tableau effektiv, d.h. falls Γ rekursiv aufzählbar aber nicht erfüllbar ist, so stoppt die systematische Tableau-Konstruktion. Insbesondere stoppt die systematische Tableau-Konstruktion immer, wenn Γ endlich ist. Sie liefert dann entweder: Γ ist nicht erfüllbar, d.h. es gibt eine $n \in \mathbb{N}$, so dass τ_n abgeschlossen ist, oder: Γ ist erfüllbar und die (offenen) Äste von τ_n liefern alle Belegungen, die Γ erfüllen.

   Die systematische Tableau-Konstruktion liefert also für endliche Mengen alle Belegungen der wesentlichen Variablen, die Γ erfüllen.

2. Zur Vereinfachung der systematischen Tableau-Konstruktion für eine Menge $\Gamma = \{A_1, \ldots, A_n\}$ beginne mit

$$\begin{array}{l} \bullet A_1 \\ | \\ \bullet A_2 \\ | \\ \bullet A_3 \\ \vdots \\ \bullet A_{n-1} \\ | \\ \bullet A_n \end{array}$$

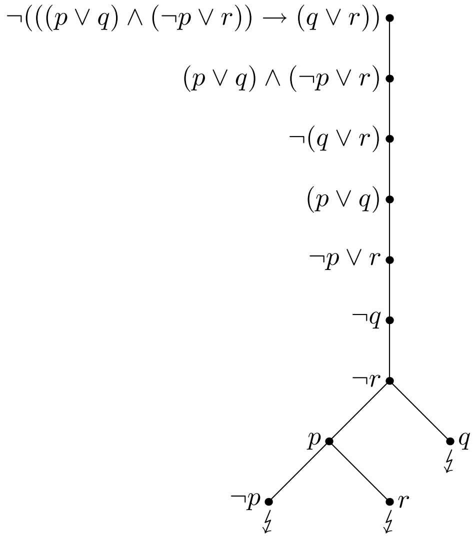
   als Anfangsbaum.

3.

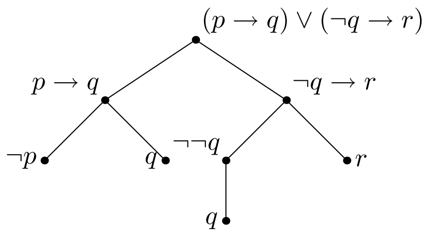
$$\begin{aligned} \Gamma \models A &\iff \Gamma \cup \{\neg A\} \text{ unerfüllbar} \\ &\iff \tau_{\{\Gamma, \neg A\}} \text{ enthält abgeschlossenes Tableau} \\ &\iff \Gamma \vdash_\tau A \end{aligned}$$

4. Beispiele:

a) $\models ((p \vee q) \wedge (\neg p \vee r)) \rightarrow (q \vee r)$ oder $(p \vee q) \wedge (\neg p \vee r) \models (q \vee r)$



b) Bestimme alle Belegungen, die $A \equiv (p \rightarrow q) \vee (\neg q \rightarrow r)$ erfüllen!



Demnach ist $\{\varphi \mid \varphi$ ist Bewertung mit $\varphi(p) = 0$ oder $\varphi(q) = 1$ oder $\varphi(r) = 1\}$ die Menge aller Belegungen, die A erfüllen. An den Blättern des Baumes lässt sich auch die äquivalente Disjunktive Normalform (DNF) zur Formel A ablesen, nämlich $\neg p \vee q \vee r$.

2 Prädikatenlogik – Formalisierung der Beziehungen zwischen Eigenschaften von Elementen, Funktionen und Prädikaten

Es zeigt sich, dass die Aussagenlogik nicht ausreicht, alle logischen Schlüsse zu beschreiben, insbesondere nicht solche Schlüsse, die die Eigenschaften von Elementen und deren Beziehungen untereinander betreffen.

Ein Beispiel einer Anwendung von logischen Formeln, die keine Formeln der Aussagenlogik sind, ist Hilberts 10. Problem: Gibt es ein Verfahren um zu entscheiden, ob eine diophantische Gleichung eine Lösung hat?

Gegeben eine Polynomgleichung in mehreren Veränderlichen mit Koeffizienten in $\mathbb{Z}$, z.B. $3x^2 - 2xy^3 + 5z^3 + 6 = 0$. Frage: Gibt es für diese Gleichung eine Lösung (x, y, z)?

Zunächst wird das Problem in einer geeigneten Sprache formuliert. Ein geeignetes Alphabet für diese Sprache besteht aus den folgenden Mengen von Zeichen:

- für ganze Zahlen die Symbole: $\ldots, -2, -1, 0, 1, 2, \ldots$
- eine abzählbare Menge von Variablen: $x, y, z, \ldots$
- eine Menge von Funktionszeichen: $+, -, \star, \ldots$
- eine Menge von Prädikatszeichen: $<, >, \ldots$
- eine Menge von logischen Zeichen: $=, \wedge, \vee, \neg, \rightarrow, \ldots$
- Quantoren: $\forall, \exists$
- Trennzeichen: (und)

Zur Formalisierung einer Aussage der Form: "Jede ganze Zahl ist gerade oder ungerade!" oder "Die Gleichung $3x^2 - 2xy^3 + 5z^3 + 6 = 0$ hat eine Lösung!" werden für die Eigenschaften von Elementen Prädikate definiert. Für die Eigenschaft von x gerade oder ungerade zu sein werden beispielsweise die Prädikate $U(x)$ und $G(x)$ eingeführt. Die Definitionen für diese Prädikate lauten in obiger Sprache dann $G(x) \equiv \exists z\ x = 2 \star z$ und $U(x) \equiv \exists z\ x = 2 \star z + 1$ und die oben erwähnten Aussagen lauten formal: $\forall x\ (U(x) \vee G(x))$ und

$$\exists x \exists y \exists z\ (((3 \star (x \star x)) - ((2 \star (x \star (y \star (y \star y)))) + ((5 \star (z \star (z \star z))) + 6))) = 0).$$

Derartige Aussagen werden Formeln genannt. Formal besteht eine Formel aus logischen Verknüpfungen von Termen. Ein Term wiederum kann bezüglich des obigen Alphabetes wie folgt definiert werden: Jede ganze Zahl ist ein Term, jede Variable ist ein Term und falls t_1 und t_2 Terme sind, dann sind auch $(t_1 + t_2), (t_1 - t_2)$ und $(t_1 \star t_2)$ Terme. Eine Formel hat entweder die Gestalt $t_1 = t_2$ oder $t_1 < t_2$, wobei t_1 und t_2 Terme sind, oder die Gestalt $\exists x\ A$, $\forall x\ A$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$ oder $(\neg A)$, wobei A und B ihrerseits bereits Formeln sind und x eine Variable ist.

Der nächste Schritt ist es, einer Formel eine Bedeutung zuzuordnen. Welche Bedeutung hat eine Formel der oben definierten Sprache in der Menge $\mathbb{Z}$ der ganzen Zahlen? Wann ist sie falsch und wann wahr? Eine Formel wie $(x + 5) > y$ oder $\forall x\ (x + 5) = 0$ hat für sich allein noch keine Bedeutung. Erst wenn eine Menge bestimmt wird, deren Werte die Variablen und Konstanten annehmen können, und wenn darüber hinaus den Symbolen wie $5, -1$ und den "freien" Variablen (das sind Variablen, die nicht im Gültigkeitsbereich eines Quantors liegen) ein solcher Wert zugeordnet wird und den Prädikats- und Funktionszeichen eine Bedeutung zugewiesen wird, kann die Bedeutung einer Formel erklärt werden. Das Bestimmen einer solchen Menge, die auch Definitionsbereich genannt wird, sowie die Festlegung der Bedeutung der einzelnen Symbole heißt Interpretation. Durch eine Interpretation wird die Bedeutung einer Formel festgelegt.

Beispiel: Sei I_1 eine Interpretation, die als Definitionsmenge die Menge $\mathbb{Z}$ der ganzen Zahlen hat, den Zahlensymbolen ihre üblichen Werte aus $\mathbb{Z}$ und den Symbolen $+$ und $=$ die gewohnte arithmetische Bedeutung zuordnet. Für die Variablen x und y lege I_1 die Werte 6 bzw. 3 fest. Die Interpretation I_2 lege für die Variable x den Wert 0 und für y den Wert 5 fest und entspräche im übrigen der Interpretation I_1. Dann ist die Formel $(x+5) = y$ falsch unter der Interpretation I_1 und wahr unter I_2. Die Formel $\forall x\ (x+5) > 0$ dagegen ist unter beiden Interpretationen falsch. Diese Formel ist falsch, unabhängig davon, wie die Variable x interpretiert wird. Wählt man aber als Definitionsbereich statt $\mathbb{Z}$ die Menge $\mathbb{N}$, wird die Formel wahr, wiederum unabhängig davon, wie die Variable x interpretiert wird.

Es tauchen die Fragen auf, ob es zu einer bestimmten Formel Interpretationen gibt, die die Formel "wahr" machen, oder ob eine Formel für alle Interpretationen wahr ist oder für alle Interpretationen falsch ist. Darüber hinaus stellt sich auch die Frage nach der "logischen Äquivalenz" zweier Formeln. Und natürlich die Frage, ob diese Fragen im allgemeinen überhaupt beantwortet werden können.

Man unterscheidet zwischen Prädikatenlogik erster und zweiter Stufe. Sind bei Formeln der ersten Stufe Quantifizierungen nur über Individuenvariablen möglich, so dürfen Formeln der zweiten Stufe darüber hinaus auch Quantifizierungen von Funktions- und Prädikatsvariablen enthalten. Beispielsweise kann man obiges Alphabet erweitern um die Funktionsvariablen $\Delta, F, G, \dots$ und Prädikatsvariablen $P, Q, \dots$ Dann ist die Formel zweiter Stufe $\exists \Delta\ (4\Delta 5) = 20$ unter der üblichen arithmetischen Interpretation wahr, denn wird Δ als die Multiplikation auf $\mathbb{Z}$ interpretiert, ist die Formel $(4\Delta 5) = 20$ wahr.

2.1 Die Sprache der Prädikatenlogik

2.1 Definition (Allgemeine Sprache der Prädikatenlogik zweiter Stufe)**.**

1. Das *Alphabet* für eine Sprache der *Prädikatenlogik zweiter Stufe* besteht aus folgenden Teilalphabeten:

   a) *Wahrheitswerte*: W und F

   b) *Logische Symbole:*

i. *Junktoren:* $\neg, \rightarrow, \wedge, \vee, \leftrightarrow, \ldots,$ IF_THEN_ELSE_
ii. *Operatoren:* $=$, if_then_else
iii. *Quantoren:* $\forall$ *(Allquantor)* und $\exists$ *(Existenzquantor)*

c) *Variablensymbole:*
i. n-stellige *Funktionsvariablen* F_j^n ($j \geq 1, n \geq 0$). Die F_j^0 heißen *Individuenvariablen* und werden mit x_j bezeichnet.
ii. n-stellige *Prädikatsvariablen* P_j^n ($j \geq 1, n \geq 0$). Die P_j^0 heißen *aussagenlogische Variablen.*

d) *Konstantensymbole:*
i. n-stellige *Funktionskonstanten* f_j^n ($j \geq 1, n \geq 0$). Die f_j^0 heißen *Individuenkonstanten* und werden mit a_j bezeichnet.
ii. n-stellige *Prädikatskonstanten* p_j^n ($j \geq 1, n \geq 0$). Die p_j^0 werden als W oder F interpretiert.

e) 5. *Hilfssymbole:* (und) und ,.

Alle Zeichen sollen verschieden sein und kein Buchstabe soll Teilwort eines anderen sein. Die einzelnen Teilalphabete sollen entscheidbar sein. Spezielle Sprachen zweiter Stufe werden durch Festlegung der Konstanten ausgezeichnet.

2. *Ausdrücke* in Sprachen zweiter Stufe sind *Terme* und *Formeln*:

a) Die Menge Term der *Terme* ist rekursiv definiert:
i. Jede Individuenvariable x_j und jede Individuenkonstante a_j ($j \geq 1$) ist ein (atomarer) Term.
ii. Sind $t_1, \ldots, t_n$ ($n \geq 1$) Terme, dann sind auch $f_j^n(t_1, \ldots, t_n)$ und $F_j^n(t_1, \ldots, t_n)$ Terme für alle $j \geq 1$.
iii. Ist A eine Formel und sind t_1 und t_2 Terme, dann ist (if A then t_1 else t_2) ein Term.
iv. Term ist die kleinste Menge, die i) genügt und gegenüber ii) und iii) abgeschlossen ist.

b) Die Formeln teilen sich in atomare und zusammengesetzte Formeln auf.
Die Menge Aform der *atomaren Formeln* ist die kleinste Menge mit:
i. $W, F \in$ Aform
ii. $p_j^0, P_j^0 \in$ Aform ($j \geq 1$) (die aussagenlogischen Konstanten und Variablen)
iii. Sind $t_1, \ldots, t_n$ ($n \geq 1$) Terme, dann sind

$$p_j^n(t_1, \ldots, t_n), P_j^n(t_1, \ldots, t_n) \in \text{Aform } (j \geq 1).$$

iv. Sind t_1, t_2 Terme, dann ist $(t_1 = t_2) \in$ Aform.

Die Menge Form der *Formeln* ist die kleinste Menge mit

i. Aform $\subseteq$ Form

ii. Sind $A, B, C \in$ Form, dann sind auch $(\neg A)$, $(A \rightarrow B)$, $(A \vee B)$, $(A \wedge B)$, $(A \leftrightarrow B)$, $(\text{IF } A \text{ THEN } B \text{ ELSE } C) \in$ Form .

iii. Ist v eine (Funktions- oder Prädikats-) Variable und A eine Formel, dann sind $((\forall v)\ A), ((\exists v)\ A) \in$ Form . (Einfachheitshalber nehmen wir an, dass weder $(\forall v)$ noch $(\exists v)$ bereits in A vorkommt.)

3. *Geltungsbereich* eines Quantors : Ist $B \equiv ((\forall v)\ A)$ oder $B \equiv ((\exists v)\ A)$, so ist A der *Geltungsbereich* von $(\forall v)$ bzw. $(\exists v)$. Ein Vorkommen von v in A heißt *gebunden*. Variablen in einer Formel kommen gebunden vor, falls sie im Geltungsbereich eines Quantors vorkommen. Sonstige Vorkommen einer Variablen heißen *frei*. Eine Variable v heißt *freie Variable* einer Formel A, wenn es in A freie Vorkommen von v gibt. Formeln ohne freie Variablen heißen *abgeschlossen* oder *Aussagen*.

4. Ein *Teilterm* eines Terms t ist ein Teilwort von t, das selbst ein Term ist. Eine *Teilformel* einer Formel A, ist ein Teilwort von A, das selbst eine Formel ist.

2.2 Bemerkungen und Beispiele.

1. Die Mengen Term und Form sind rekursiv entscheidbar. Sie werden jeweils eindeutig erzeugt, d.h. zusammengesetzte Terme und Formeln lassen sich eindeutig zerlegen (Bemerkung 1.2 und Satz 1.3 gelten entsprechend).

2. Vereinbarungen: Stelligkeiten können bei Prädikaten und Funktionen weggelassen werden, wenn sie aus dem Kontext hervorgehen.

   Man benutzt $a, b, c, \ldots$ für Individuenkonstanten, $f, g, h, \ldots$ für Funktionskonstanten, $p, q, r, \ldots$ für Prädikatskonstanten, $x, y, z, \ldots$ für Individuenvariablen, $F, G, H, \ldots$ für Funktionsvariablen, $P, Q, R, \ldots$ für Prädikatsvariablen, $A, B, C, \ldots$ für Formeln und t für Terme. Die Symbole für Individuenkonstanten und -variablen sowie für Terme werden auch in indizierter Form benutzt.

   Wenn keine Verwechslungen möglich sind, können Klammern entfallen. Beispielsweise wird $\forall x\ A$ statt $((\forall x)\ A)$ und $\neg A$ statt $(\neg A)$ geschrieben. Um die Lesbarkeit einer Formel zu erhöhen werden gelegentlich die Klammern $[,]$ oder $\{,\}$ gebraucht. $\neg, \exists v$ und $\forall v$ binden am stärksten, d.h. sie werden auf die kleinst mögliche Teilformel angewandt. Zum Beispiel steht $\neg p \vee \neg q$ für $((\neg p) \vee (\neg q))$ und $\neg \forall x\ p(x) \leftrightarrow \exists x\ \neg p(x)$ für $((\neg((\forall x)\ p(x))) \leftrightarrow ((\exists x)\ (\neg(p(x)))))$. Statt $\neg(t_1 = t_2)$ wird häufig $t_1 \neq t_2$ geschrieben.

3. Beispiele:

   a)

$$
\begin{aligned}
(\exists F_1^1)\ ((F_1^1(a_1) = a_2) \wedge ((\forall x_1)\ (p_1^1(x_1) \\
\rightarrow (F_1^1(x_1) = f_1^2(x_1, F_1^1(f_1^1(x_1)))))) \in \text{Form}
\end{aligned}
$$

in vereinfachter Schreibweise:

$$\underbrace{\exists F\,\{\underbrace{\underbrace{(\underbrace{F(a)}_{\text{Term}} = \underbrace{b}_{\text{Term}})}_{\text{at.F.}} \wedge \underbrace{\forall x\,[\underbrace{\underbrace{p(\underbrace{x}_{\text{Term}})}_{\text{at.F}} \to \underbrace{(\underbrace{F(x)}_{\text{Term}} = \underbrace{g(x,F(f(x)))}_{\text{Term}})}_{\text{at.Formel}}}_{\text{Formel}}]}_{\text{Formel}}}_{\text{Formel}}\}}_{\text{Formel}}$$

Alle markierten Formeln und atomaren Formeln sind Teilformeln der obigen Formel.

b) In der folgenden Formel kommen die Variablen P und x vor. P kommt 4-mal vor und x kommt 4-mal vor:

$$\forall P\ \{[\underbrace{P}_{\text{geb.}}(a)\ \wedge\ \forall x\ [[\underbrace{x}_{\text{geb.}} \neq a\ \wedge\ \underbrace{P}_{\text{geb.}}(f(\underbrace{x}_{\text{geb.}}))] \to \underbrace{P}_{\text{geb.}}(\underbrace{x}_{\text{geb.}})]] \to \underbrace{P}_{\text{geb.}}(\underbrace{x}_{\text{frei}})\}$$

Die Formel ist nicht abgeschlossen, da eine Variable ein freies Vorkommen hat. Die Formel aus (a) ist abgeschlossen.

2.3 Eingeschränkte Teilsprachen. Durch Einschränkung der Konstanten und Variablen in Definition 2.1 entstehen folgende Teilsprachen:

1. Für die *Sprache der Prädikatenlogik erster Stufe* werden nur Individuenvariablen x_j $(j \geq 1)$ zugelassen, keine Funktions- und Prädikatsvariablen. Eine typische Formel lautet:

$$(x_1 \neq x_2) \wedge \forall x_2\ (\exists x_3\ p(x_1, f(x_2, x_3)) \to (p(x_2, x_1) \vee p(x_2, a)))$$

2. Im *Gleichheitskalkül* sind nur Individuenkonstanten a_j $(j \geq 1)$ und Individuenvariablen x_j $(j \geq 1)$ zugelassen. Es sind keine weietern keine Funktionsvariablen oder -konstanten und auch keine Prädikatskonstanten und -variablen zugelassen.

   Ein Term im Gleichheitskalkül ist eine Individuenkonstante oder eine Individuenvariable oder hat die Gestalt "if A then t_1 else t_2", wobei t_1 und t_2 Terme und A eine Formel ist.

   Die Menge Aform der atomaren Formeln im Gleichheitskalkül besteht nur aus den Wahrheitswerten W und F und aus allen Formeln der Form $t_1 = t_2$, wobei t_1 und t_2 Terme sind. Eine typische Formel lautet:

$$((\forall x_1)(\forall x_2((\forall x_3)\ (((x_1 = x_2) \wedge (x_2 = a)) \to (x_1 = a)))))$$

3. In der *quantifizierten Aussagenlogik* sind nur aussagenlogische Konstanten p_j^0 ($j \geq 1$) und aussagenlogische Variablen P_j^0 ($j \geq 0$) zugelassen. Es gibt keine Terme und atomare Formeln sind die aussagenlogischen Konstanten und Variablen, sowie die Wahrheitswerte W und F. Eine typische Formel lautet:

$$(((\forall P_1)\ (P_1 \leftrightarrow p)) \rightarrow ((\exists P_2)\ (P_2 \leftrightarrow W)))$$

4. In der *Aussagenlogik* sind nur aussagenlogische Konstanten p_j^0 ($j \geq 0$) zugelassen.

   Eine typische Formel lautet:

$$(\text{IF}\ p_1\ \text{THEN}\ p_2\ \text{ELSE}\ p_3) \leftrightarrow (\text{IF}\ \neg p_1\ \text{THEN}\ p_3\ \text{ELSE}\ p_2)$$

5. Spezielle Sprachen erster und zweiter Stufe werden durch Festlegung der Konstanten definiert. Da Funktionskonstanten und Prädikatskonstanten gleichsam als Parameter für eine bestimmte prädikatenlogische Sprache agieren und oft explizit erwähnt werden müssen, liegt folgende Definition nahe:

2.4 Definition. Eine *Basis* für die Prädikatenlogik ist ein Paar $B = (F, P)$ von Zeichenmengen, wobei F die Menge der Funktionskonstanten und P die Menge der Prädikatskonstanten ist.

2.5 Beispiel. Ist $B = (F, P)$ Basis für die *Sprache der Arithmetik*, einer Sprache erster Stufe, dann besteht die Menge F aus den 0-stelligen Konstanten 0 und 1 und der einstelligen Funktionskonstanten S (für Nachfolger) sowie den zweistelligen Funktionskonstanten $+$ und $\star$. Die Menge P enthält die zweistellige Prädikatskonstante $<$.

2.2 Die Semantik der Prädikatenlogik

Nachdem die Syntax einer Sprache der Prädikatenlogik definiert ist, muss die Semantik einer prädikatenlogischen Formel erklärt werden, nämlich, wann eine Formel "falsch" ist und wann sie "wahr" ist. Dazu wird der Begriff der *Interpretation* eingeführt. Unterschiedliche Interpretationen können zu unterschiedlichen Bedeutungen einer Formel führen. Eine Interpretation wird rekursiv analog der Definition von Formeln definiert.

2.6 Definition. Sei L eine Sprache der Prädikatenlogik zweiter Stufe (festgelegt durch die Konstantensymbole).

1. Eine *Interpretation* I für L ist ein Tripel $I = (D, I_c, I_v)$ mit
   - $D \neq \emptyset$, dem Individuenbereich oder Definitionsbereich der Interpretation,
   - I_c, einer *Belegung* (Abbildung) *der Konstanten,* die jedem n-stelligen Funktionssymbol f eine Funktion $I_c(f) : D^n \rightarrow D$ und jedem m-stelligen Prädikatssymbol p ein Prädikat $I_c(p) : D^m \rightarrow \mathbb{B}$ zuordnet ($n, m \geq 0,\ \mathbb{B} = \{0, 1\}$) und

- I_v, einer *Belegung der Variablen*, die jeder n-stelligen Funktionsvariablen F eine Funktion $I_v(F) : D^n \to D$ und jeder m-stelligen Prädikatsvariablen P ein m-stelliges Prädikat $I_v(P) : D^m \to \mathbb{B}$ zuordnet $(n, m \geq 0)$.

2. Die Interpretation von Termen und Formeln ist eine Fortsetzung von I auf Term und Form: I ordnet jedem $t \in$ Term ein $I(t) \in D$ und jedem $A \in$ Form ein $I(A) \in \mathbb{B}$ zu.

   a) Bewertung der Terme:
      - $I(a) = I_c(a)$ für ein 0-stelliges Konstantensymbol a
      - $I(x) = I_v(x)$ für ein 0-stelliges Variablensymbol x
      - $I(f(t_1, \ldots, t_n)) = I_c(f)(I(t_1), \ldots, I(t_n))$ für eine n-stellige Funktionskonstante f und Terme $t_1, \ldots, t_n$ $(n \geq 1)$
      - $I(F(t_1, \ldots, t_n)) = I_v(F)(I(t_1), \ldots, I(t_n))$ für eine n-stellige Funktionsvariable F und Terme $t_1, \ldots, t_n$ $(n \geq 1)$
      - Für eine Formel A und Terme t_1, t_2 ist

$$I(\text{if } A \text{ then } t_1 \text{ else } t_2) = \begin{cases} I(t_1), & \text{falls } I(A) = 1, \\ I(t_2), & \text{falls } I(A) = 0. \end{cases}$$

      Beachte: $I(t)$ hängt für einen Term t nur von den Belegungen der in t vorkommenden Variablen und Konstanten ab.

   b) Bewertung der Formeln:
      - $I(W) = 1, I(F) = 0$
      - $I(p) = I_c(p)$ für eine 0-stellige Prädikatskonstante p
      - $I(P) = I_v(P)$ für eine 0-stellige Prädikatsvariable P
      - $I(p(t_1, \ldots, t_n)) = I_c(p)(I(t_1), \ldots, I(t_n))$ für eine n-stellige Prädikatskonstante p und Terme $t_1, \ldots, t_n$ $(n \geq 1)$
      - $I(P(t_1, \ldots, t_n)) = I_v(P)(I(t_1), \ldots, I(t_n))$ für eine n-stellige Prädikatsvariable P und Terme $t_1, \ldots, t_n$ $(n \geq 1)$
      - Für alle $t_1, t_2 \in$ Term ist

$$I(t_1 = t_2) = \begin{cases} 1, & \text{falls } I(t_1) = I(t_2), \\ 0, & \text{sonst.} \end{cases}$$

      - Die Interpretationen $I(\neg A)$, $I(A \vee B)$, $I(A \wedge B)$, $I(A \to B)$, $I(A \leftrightarrow B)$ und $I(\text{IF } A \text{ THEN } B \text{ ELSE } C)$ sind wie in der Aussagenlogik definiert.
      - $I(\forall x\ A) = \begin{cases} 1, & \text{falls } I^{x,d}(A) = 1 \text{ für alle } d \in D \text{ gilt}, \\ 0, & \text{sonst.} \end{cases}$
      - $I(\exists x\ A) = \begin{cases} 1, & \text{falls es ein } d \in D \text{ mit } I^{x,d}(A) = 1 \text{ gibt}, \\ 0, & \text{sonst.} \end{cases}$

Dabei ist $I^{x,d} = (D, I_c, I'_v)$ mit

$$I'_v(y) = \begin{cases} d & , \quad \text{falls } y \equiv x \\ I_v(y), & \text{sonst} \end{cases}$$

und $I'_v(F) = I_v(F)$ und $I'_v(P) = I_v(P)$ für n-stellige Funktions- bzw. Prädikatsvariablen F und P ($n \geq 1$).
Analog wird $I(\forall v\ A)$ bzw. $I(\exists v\ A)$ definiert, wenn v eine n-stellige ($n \geq 1$) Funktionsvariable oder eine m-stellige ($m \geq 0$) Prädikatsvariable ist. Es werden statt $d \in D$ dann Funktionen $\psi : D^n \to D$ bzw. Prädikate $\pi : D^n \to \mathbb{B}$ eingesetzt.

Jede Interpretation $I = (D, I_c, I_v)$ induziert durch (a) und (b) eine Bewertung aller Terme und Formeln, die die bewerteten Konstanten und Variablen als freie Variablen enthalten. Umgekehrt wird jede Bewertung I, die (a) und (b) genügt, eindeutig durch eine solche Interpretation induziert.

3. Gilt $I(A) = 1$, so ist A *wahr* in der Interpretation I oder I *erfüllt* A. Schreibweise: $\models_I A$ oder $I \models A$.

2.7 Bemerkungen und Beispiele.

1. Um die Bewertung einer Formel A zu bestimmen, genügt es, die Bewertung der Konstanten und der in A frei vorkommenden Variablen zu kennen. Die Bewertung von A unter $I = (D, I_c, I_v)$ hängt nur von diesen Werten ab. Ist A abgeschlossen (d.h. A ist ohne freie Variablen), so genügt es, eine Interpretation der Form $I = (D, I_c)$ zu betrachten. Solche Interpretationen heißen auch *algebraische Strukturen* oder *Relationalsysteme*. Man beschreibt ein Relationalsystem (algebraische Struktur) häufig als $< A;\ f_1, \ldots, f_n;\ R_1, \ldots, R_m >$, wobei der Definitionsbereich A die Menge der Objekte bezeichnet, über denen die Funktionen $f_j : A^{h_j} \to A$ ($1 \leq j \leq n,\ h_j \geq 0$) und die Relationen R_k ($1 \leq k \leq m$) definiert sind.

2. Beachte, dass in der Definition der Bewertung Redundanzen vorkommen. Zum Beispiel gleicht die Bewertung der Formel $\neg A$ der Bewertung von IF A THEN F ELSE W, die von $A \wedge B$ der von IF A THEN B ELSE F, die von $A \vee B$ der von IF A THEN W ELSE B, die von $A \to B$ der von IF A THEN B ELSE W, die von $A \leftrightarrow B$ der von IF A THEN B ELSE $\neg B$, die von $\forall v\ A$ der von $\neg\exists v\ \neg A$, die von $\exists v\ A$ der von $\neg\forall v\ \neg A$ und die von $t_1 = t_2$ der von $\forall P\ [P(t_1) \leftrightarrow P(t_2)]$.

3. Beispiele:

   a) $\exists x \forall y\ (p(y) \to x = y)$ bedeutet "Es gibt ein x, so dass für alle y immer $x = y$ gilt, wenn $p(y)$ zutrifft" oder "Es gibt höchstens ein x, so dass $p(x)$ wahr ist". Diese Formel ist wahr in allen Interpretationen, in denen $p(d)$ für höchstens ein Element $d \in D$ wahr ist (z.B. $D = \mathbb{N}$ und $I(p(x))$ genau dann, wenn $I(x) = 0$).

b) $\exists x\ [p(x) \wedge \forall y\ [p(y) \rightarrow x = y]]$ bedeutet "Es gibt genau ein x, so dass $p(x)$ wahr ist".

c) $\forall z \exists u \exists v\ ((z = u \vee z = v) \wedge u \neq v) \wedge \forall x \forall y \forall P\ [x \neq y \vee (P(x,x) \vee \neg P(y,y))]$
Beh.: Diese Formel ist wahr in jeder Interpretation mit $|D| \geq 2$ und falsch für $|D| = 1$.

d) Seien $A \equiv \forall x\ p(x, f(x)), A_1 \equiv p(x, f(x)), I = (\mathbb{N}, I_c, I_v)$ und sei $I_c(p)$ das $\leq$-Prädikat und $I_c(f) = \text{quad}$ mit $\text{quad} : n \mapsto n^2$. Dann gilt: $I(A) = I^{x,n}(A_1) = (n \leq n^2) = W$ für alle $n \in \mathbb{N}$.

4. Sind I_1 und I_2 Interpretationen mit den gleichen Definitionsbereichen und ist A eine Formel, dann gilt "$I_1 \models A$ genau dann, wenn $I_2 \models A$", falls I_1 und I_2 in allen Konstanten und freien Variablen, die in A vorkommen übereinstimmen.

2.8 Definition. Sei L eine Sprache der Prädikatenlogik zweiter Stufe.

1. Eine Formel $A \in \text{Form}$ heißt *allgemeingültig*, falls $I(A) = 1$ für alle Interpretationen I für L, d.h. $I \models A$ für alle Interpretationen I. Schreibweise: $\models A$.

2. Eine Formel $A \in \text{Form}$ heißt *erfüllbar*, falls es eine Interpretation I mit $I(A) = 1$ gibt. I heißt dann auch *Modell* für A. Gibt es keine solche Interpretation, so heißt A *unerfüllbar*.

3. $\Sigma \subseteq \text{Form}$ heißt *erfüllbar*, falls es eine Interpretation I gibt, die alle Formeln von Σ erfüllt.

2.9 Bemerkungen und Beispiele.

1. Ist A nicht allgemeingültig, so ist $\neg A$ erfüllbar. Ist $\neg A$ unerfüllbar, dann ist A allgemeingültig. Um die Allgemeingültigkeit (Erfüllbarkeit) einer Formel zu prüfen, genügt es solche Interpretationen zu betrachten, die die Konstanten und freien Variablen der Formel belegen. Das sind, im Gegensatz zur Aussagenlogik, immer noch unendlich viele, denn die Definitionsmenge kann eine beliebige nichtleere Menge sein.

2. Jede Tautologie der Aussagenlogik ist allgemeingültig. Genauer:

   Tautologie-Theorem: Sei A eine Formel der Aussagenlogik mit den Aussagevariablen $p_1, \ldots, p_n$ $(n \geq 1)$, d.h. $A \equiv A(p_1, \ldots, p_n)$. A' entstehe aus A durch Ersetzen von p_j $(1 \leq j \leq n)$ in A durch eine Formel B_j der Prädikatenlogik zweiter Stufe. Ist A eine Tautologie (d.h. allgemeingültig), so ist auch A' allgemeingültig. Allgemeingültig sind beispielsweise die Formeln $A_1 \vee \neg A_1$ und $A_1 \rightarrow (A_2 \rightarrow A_1)$.

3. Beispiele:

   a) $\forall x \forall y \forall P\ (x \neq y \vee (P(x,x) \vee \neg P(y,y)))$ ist allgemeingültig. Da die Formel weder Konstanten noch freie Variablen enthält, genügt es eine Interpretation $I = (D)$ mit $|D| \geq 1$ zu betrachten. Der Fall $|D| = 1$ ist klar, denn entweder gilt $P(x,x)$ oder $\neg P(x,x)$. Ist $|D| > 1$, dann wähle $x \equiv d_1$ und $y \equiv d_2$ mit $d_1, d_2 \in D$ und $d_1 \neq d_2$. Dann ist $I(x \neq y) = 1$ und somit erfüllt I die Formel.

Die Formel $A \equiv \exists P \forall x \exists y\ (P(x,x) \wedge \neg P(x,y))$ ist weder allgemeingültig noch unerfüllbar. Sei $I = (D)$ eine Interpretation. Ist $|D| = 1$, dann ist $I(A) = 0$. Ist $|D| \geq 2$, dann ist $I(A) = 1$.

Die Formel $\exists P \exists x \exists y\ ((P(x,x) \wedge \neg P(x,y)) \wedge x = y)$ ist unerfüllbar, die Formeln $t = t, \forall x\ A \leftrightarrow \neg \exists x\ (\neg A)$ und $\exists x\ A \leftrightarrow \neg \forall x\ (\neg A)$ sind allgemeingültig.

b) Sei p eine 2-stellige Prädikatskonstante.

$A_1 \equiv \forall x \forall y \forall z\ ((p(x,y) \wedge p(y,z)) \rightarrow p(x,z))$	(transitiv)
$A_2 \equiv \forall x \forall y\ (p(x,y) \vee p(y,x) \vee x = y)$	(Trichotomie)
$A_3 \equiv \forall x\ \neg p(x,x)$	(antireflexiv)
$A_4 \equiv \forall x \forall y\ (p(x,y) \rightarrow \exists z\ (p(x,z) \wedge p(z,y)))$	(dicht)
$A_5 \equiv \forall x \exists y\ p(x,y)$	(kein letztes Element)
$A_6 \equiv \forall x \exists y\ p(y,x)$	(kein erstes Element)

Die Formeln $A_1, \ldots, A_3$ beschreiben eine lineare Ordnung auf der Definitionsmenge, die Formeln $A_1, \ldots, A_4$ eine dichte lineare Ordnung (DLO).

Keine der Formeln ist allgemeingültig. Sind sie erfüllbar? Betrachte die folgenden Interpretationen: $I_1 = (\{0,1,2\}, \ldots), I_2 = (\mathbb{N}, \ldots), I_3 = ([0,1], \ldots)$ und $I_4 = (\mathbb{Q}, \ldots)$ ($\mathbb{Q}$ bezeichne die Menge der Brüche) und in allen Interpretationen sei p das "kleiner"-Prädikat. A_1, A_2 und A_3 sind in allen vier Interpretationen wahr, $A_1, \ldots, A_4$ sind wahr in I_3 und I_4. In I_1 und I_2 ist A_4 falsch. $A_1, \ldots, A_6$ sind wahr in I_4, A_5 und A_6 sind falsch in I_1 und I_3. A_5 ist wahr in I_2 und A_6 ist falsch in I_2.

c) Allgemeingültige Formeln:

$$\begin{array}{lcl}
A_1 & \equiv & \forall x\ p(x) \rightarrow \exists x\ p(x) \\
A_2 & \equiv & p(x) \rightarrow p(x) \\
A_3 & \equiv & \forall x\ q(x) \rightarrow q(a) \\
A_4 & \equiv & p(a) \rightarrow \exists x\ p(x) \\
A_5 & \equiv & \forall x \forall y\ (p(x,y) \rightarrow p(x,y)) \\
A_6 & \equiv & \exists y \forall x\ p(x,y) \rightarrow \forall x \exists y\ p(x,y) \\
A_7 & \equiv & (\forall x\ p(x) \vee \forall x\ q(x)) \rightarrow \forall x\ (p(x) \vee q(x))
\end{array}$$

Nicht allgemeingültige Formeln:

$$\begin{array}{lcl}
B_1 & \equiv & \exists x\ p(x) \rightarrow \forall x\ p(x) \\
B_2 & \equiv & p(x) \rightarrow p(a) \\
B_3 & \equiv & q(a) \rightarrow \forall x\ q(x) \\
B_4 & \equiv & \exists x\ p(x) \rightarrow p(a) \\
B_5 & \equiv & \forall x \forall y\ (p_1(x,y) \rightarrow p_1(y,x))
\end{array}$$

$$
\begin{aligned}
B_6 &\equiv \forall x \exists y\, p_1(x,y) \rightarrow \exists y \forall x\, p_1(x,y) \\
B_7 &\equiv \forall x\, (p(x) \vee q(x)) \rightarrow (\forall x\, p(x) \vee \forall x\, q(x))
\end{aligned}
$$

Um die Allgemeingültigkeit der Formeln $A_1, \ldots, A_7$ zu beweisen, zeige: Es gibt kein Gegenbeispiel, d.h. $\neg A_j$ $(1 \leq j \leq 7)$ ist unerfüllbar. Beispiel: $A_6 \equiv \exists x \forall y\ p(x,y) \rightarrow \forall x \exists y\ p(x,y)$. Sei $I = (D, \ldots)$ eine Interpretation, die A_6 falsch macht, d.h. $\exists x \forall y\ p(x,y)$ muss wahr sein und $\forall x \exists y\ p(x,y)$ muss falsch sein unter I. Es gibt dann also ein $d_1 \in D$, so dass $I(p(d,d_1)) = W$ für alle $d \in D$. Weil $\forall x \exists y\ p(x,y)$ falsch ist, muss $\exists y\ p(d_2,y)$ für ein $d_2 \in D$ falsch sein. Da $I(p(d_2,d_1)) = W$ gilt, muss auch $\exists y\, p(d_2,y)$ wahr sein.
Als Gegenbeispiel für $B_1, \ldots, B_7$ wähle die Interpretation $I = (\mathbb{Z}, \ldots)$ mit

$$
\begin{aligned}
I(p(x)) &= 1 \text{ genau dann, wenn } x > 0, \\
I(q(x)) &= 1 \text{ genau dann, wenn } x \leq 0, \\
I(p_1(x,y)) &= 1 \text{ genau dann, wenn } x > y,
\end{aligned}
$$

$I(a) = 0$ und $I(x) = 1$.

d) Für Formeln der Arithmetik in den Konstanten $0, S, +$ und $\star$ ist die "natürliche" Interpretation $I = (\mathbb{N}, I_c, I_v)$, die den Ziffernsymbolen ihren Wert und den Operatoren ihre übliche Bedeutung zuordnet (für S wird auch der Operator $'$ benutzt, dem I die Funktion $n' = n+1$ zuordnet). Welche abgeschlossenen Formeln (erster und zweiter Stufe) sind wahr in I?

$$
\left.
\begin{array}{l}
\forall x \forall y\ (S(x) = S(y) \rightarrow x = y) \\
\forall x\ S(x) \neq 0 \\
\forall x\ x + 0 = x \\
\forall x \forall y\ (x + S(y)) = S(x+y) \\
\forall x\ x \star 0 = 0 \\
\forall x \forall y\ (x \star S(y)) = (x \star y) + x
\end{array}
\right\} \text{Formeln erster Stufe}
$$

$$\forall P\ [(P(0) \wedge \forall x\ (P(x) \rightarrow P(S(x)))) \rightarrow \forall x\ P(x)] \quad \text{(Induktionsprinzip)}$$

Als *Theorie* bezeichnet man eine Menge von Formeln, die bezüglich logischer Folgerung abgeschlossen ist. Frage: Ist die Theorie erster Stufe, die von obigen Formeln erster Stufe aufgespannt wird (d.h. die Menge, die diese Formeln und deren Folgerungen enthält) oder die Theorie zweiter Stufe, die von obigen Formeln einschließlich des Induktionsprinzipes aufgespannt wird, für I entscheidbar? Ist die Menge der allgemeingültigen abgeschlossenen Formeln überhaupt entscheidbar?

2.10 Lemma. Die Allgemeingültigkeit für Formeln der quantifizierten Aussagenlogik ist entscheidbar.

Beweisidee: Finde zu jeder Formel der quantifizierten Aussagenlogik eine äquivalente Formel der Aussagenlogik (Quantorenelimination):

Es gelten $\models \forall P^0_j\ B \leftrightarrow B_{P^0_j}[W] \wedge B_{P^0_j}[F]$ (P^0_j durch W bzw. F ersetzten) und $\models \exists P^0_j\ B \leftrightarrow B_{P^0_j}[W] \vee B_{P^0_j}[F]$, denn $I(\forall P^0_j\ B) = I(B_{P^0_j}[W] \wedge B_{P^0_j}[F])$ und $I(\forall P^0_j\ B) = I(B_{P^0_j}[W] \wedge B_{P^0_j}[F])$ (für alle I).

Somit lassen sich die Quantoren eliminieren. Es bleibt dann eine Formel der Aussagenlogik, deren Allgemeingültigkeit (z.B. mit der Tableaumethode) entscheidbar ist.

Beispiel:

$$\begin{aligned}
\forall P \exists Q\ ((P \leftrightarrow \neg Q) \vee (p \rightarrow Q)) &\quad \longleftrightarrow \\
\exists Q\ ((W \leftrightarrow \neg Q) \vee (p \rightarrow Q)) \wedge \exists Q\ ((F \leftrightarrow \neg Q) \vee (p \rightarrow Q)) &\quad \longleftrightarrow \\
(((W \leftrightarrow \neg W) \vee (p \rightarrow W)) \vee ((W \leftrightarrow \neg F) \vee (p \rightarrow F))) \wedge & \\
(((F \leftrightarrow \neg W) \vee (p \rightarrow W)) \vee ((F \leftrightarrow \neg F) \vee (p \rightarrow F))) &\quad \longleftrightarrow \\
F \vee (p \rightarrow W) \vee W \vee (p \rightarrow F) \wedge W \vee (p \rightarrow W) \vee F \vee (p \rightarrow F) &\quad \longleftrightarrow \\
W \qquad\qquad \wedge \qquad\qquad W &
\end{aligned}$$

2.11 Definition. Sei $A \in$ Form, $t, \underline{t} \in$ Term und x eine Individuenvariable. Die *Substitution* von x durch t in A (bzw. $\underline{t}$), $A_x[t]$ (bzw. $\underline{t}_x[t]$), ist die Formel (der Term), die aus A (bzw. $\underline{t}$) entsteht, wenn man jedes freie Vorkommen von x in A (bzw. $\underline{t}$) durch t ersetzt.

Analog ist die simultane Substitution mehrerer Variablen $x_1, \ldots, x_n$ in A (bzw. $\underline{t}$) ($n > 1$) durch $t_1, \ldots, t_n \in$ Term definiert:

$A_{x_1,\ldots,x_n}[t_1, \ldots, t_n]$ (bzw. $\underline{t}_{x_1,\ldots,x_n}[t_1, \ldots, t_n]$) ist die Formel (der Term), die aus A (aus $\underline{t}$) entsteht, wenn man jedes freie Vorkommen der x_j in A (bzw. $\underline{t}$) simultan für alle j ($1 \leq j \leq n$) durch t_j ersetzt.

Die Substitution heißt *erlaubt*, wenn keine Variable die in t (bzw. in $t_1, \ldots, t_n$) vorkommt, nach der Substitution gebunden in t (bzw. $t_1, \ldots, t_n$) vorkommt.

Man kann die Substitution auch rekursiv über den Aufbau der Terme und Formeln definieren.

2.12 Beispiele und Lemma.

1. Für $A \equiv \exists y\ x = 2 \star y$ und $t \equiv y + 1$ ist $A_x[t] \equiv \exists y\ y + 1 = 2 \star y$ keine erlaubte Substitution, dagegen ist die Substitution $A_y[t] \equiv \exists y\ x = 2 \star y$ erlaubt.

   Sei $I = (\mathbb{N}, \ldots)$ eine Interpretation, die $+$ und $\star$ als Addition bzw. Multiplikation interpretiert, und sei $I(x) = 3$ und $I(y) = 2$. Dann ist $I(t) = 3$ und $I(A) = I(\exists y\ x = 2 \star y) = 0$. Nach der nicht erlaubten Substitution ist aber $I(A_x[t]) = I(\exists y\ y + 1 = 2 \star y) = 1$.

   Sei $B \equiv \forall x\ (P(x, y) \rightarrow Q(x))$ und $t \equiv f(y, z)$, dann ist die Substitution $B_y[t] \equiv \forall x\ (P(x, f(y, z)) \rightarrow Q(x))$ erlaubt.

2. **Lemma.** Sei A ein Term oder eine Formel, x eine Individuenvarible, $t \in$ Term und $A_x[t]$ eine erlaubte Substitution. Dann gilt für jede Interpretation $I = (D, I_c, I_v)$:

$$I(A_x[t]) = I^{x,I(t)}(A)$$

Insbesondere gilt $I'(A) = I(A_x[t])$, falls I und I' Interpretationen sind, die sich höchstens in der Interpretation von x unterscheiden, mit $I'(x) = I(t)$.

Beweis. Induktion über den Aufbau von A als Term bzw. Formel: Sei $I_1 = I^{x,I(t)}$

$A \in$ Term : $A \equiv x$, $A_x[t] = t$, $I(t) = I^{x,I(t)}(x) = I(t)\surd \ldots$ (entsprechend für andere Terme)

$A \in$ Form : A **atomar:** $A \equiv p(t^1, \ldots, t^n)$ $(n \geq 1)$ mit $t^j \in$ Term für $1 \leq j \leq n$.
Dann ist $A_x[t] \equiv p(t^1_x[t], \ldots, t^n_x[t])$ und

$$\begin{aligned} I(A_x[t]) &= I_c(p)(I(t^1_x[t]), \ldots, I(t^n_x[t])) \\ &= I_c(p)(I_1(t^1), \ldots, I_1(t^n)) \\ &= I_1(p(t^1, \ldots, t^n)) = I_1(A). \end{aligned}$$

Die Fälle $A \equiv \neg B$, $A \equiv B \wedge C$, $A \equiv B \vee C$, $A \equiv V \rightarrow C$, $A \equiv V \leftrightarrow C$ und $A \equiv \text{IF } B \text{ THEN } C \text{ ELSE } D$ sind einfach zu zeigen.

Sei $A_x[t] \equiv \forall y\ B_x[t]$ eine erlaubte Substitution, d.h. y kommt nicht in t vor.
Dann ist $I(t) = d_0 = I^{y,d}(t)$ für alle $d \in D$ (da y nicht in t vorkommt).
Es gilt $(I^{x,d_1})^{y,d_2} = (I^{y,d_2})^{x,d_1}$ für alle $d_1, d_2 \in D$ (da x verschieden y).

$$\begin{aligned} & I(A_x[t]) = 1 \\ \iff\ & I(\forall y\ B_x[t]) = 1 \\ \iff\ & I^{y,d}(B_x[t]) = 1 \text{ für alle } d \in D \\ \iff\ & (I^{y,d})^{x,d_0}(B) = 1 \text{ für alle } d \in D \text{ (Ind.Vor.)} \\ \iff\ & (I^{x,d_0})^{y,d}(B) = 1 \text{ für alle } d \in D \\ \iff\ & I^{x,d_0}(\forall y\ B) = 1 \\ \iff\ & I_1(A) = 1 \end{aligned}$$

Für den Fall $A \equiv \exists y\ B$ folgt dies aus $\neg\forall y\ (\neg B) \leftrightarrow \exists y\ B$.

3. Folgerung. Sei $A_x[t]$ eine erlaubte Substitution.

   a) Ist A allgemeingültig, dann ist auch $A_x[t]$ allgemeingültig.

   b) $\forall x\ A \rightarrow A_x[t]$ ist allgemeingültig.

   c) Allgemeingültig sind $\forall x\ A \rightarrow A$ und $A \rightarrow \exists x\ A$ (Spezialfall $x \equiv t$). Ist die Substitution nicht erlaubt, braucht das nicht mehr zu gelten:
   Sei $A \equiv \exists y\ (S(x) = y)$. Dann ist $A_x[y] \equiv \exists y\ (S(y) = y)$. Für die "natürliche" Interpretation $I = (\mathbb{N}, \ldots)$ der Arithmetik gilt $I(A) = 1$, aber $I(A_x[y]) = 0$.

4. Beachte: $A(v_1, \ldots, v_n)$ (d.h. A enthält als freie Variablen nur $v_1, \ldots, v_n$) sei allgemeingültig. Dann ist auch der *universelle Abschluss*

$$B \equiv \forall v_1 \forall v_2 \ldots \forall v_n\ A(v_1, \ldots, v_n)$$

allgemeingültig. Ist $A(v_1, \ldots, v_n)$ erfüllbar, dann ist auch der *existenzielle Abschluss*

$$B \equiv \exists v_1 \exists v_2 \ldots \exists v_n \, A(v_1, \ldots, v_n)$$

erfüllbar.

Sind A und $A \to B$ allgemeingültige Formeln, dann ist auch B allgemeingültig.

2.13 Definition. Sei L eine (Teil-) Sprache der Prädikatenlogik zweiter Stufe, $\Gamma \subseteq$ Form und $A, B \in$ Form.

1. A ist *logische Folgerung* aus Γ, $\Gamma \models A$, wenn jede Interpretation, die Γ erfüllt, auch A erfüllt (d.h. jedes Modell von Γ ist Modell für A).
2. A und B sind *logisch äquivalent*, $A \models\!\!=\!\!| B$, falls $A \models B$ und $B \models A$.

2.14 Bemerkung und Beispiele.

1. $\Gamma \models A$ genau dann, wenn $\Gamma \cup \{\neg A\}$ nicht erfüllbar
2. $\emptyset \models A$ genau dann, wenn $\models A$
3. Γ ist genau dann nicht erfüllbar, wenn $\Gamma \models A$ für alle $A \in$ Form gilt.
4. $\Gamma \subseteq \Sigma$ und $\Gamma \models A$ impliziert $\Sigma \models A$
5. Äquivalent sind:
   - $A \models\!\!=\!\!| B$,
   - $\models A \leftrightarrow B$ und
   - $I(A) = I(B)$ für jede Interpretation I.

   Beispiele:

   a) $\forall x \, Q(x) \models Q(y)$ (Spezialfall von $\forall x \, A \models A_x[t]$ aus 2.12)
   b) $A(y) \not\models \forall y \, A(y)$: Sei $A(y) \equiv p(y)$ und $I = (\{0,1\}, I_c, I_v)$ eine Interpretation mit $I(p(x)) = 1 \iff I(x) = 0$ und $I(y) = 0$), dann ist $I(A(y)) = 1$ aber $I(\forall y \, A(y)) = 0$.
   c) $\models \exists x \, (Q(x) \to \forall x \, Q(x))$: Sei $I = (D, Q_I)$ eine Interpretation. $I(\exists x \, (Q(x) \to \forall x \, Q(x)) = 1 \overset{\text{Def}}{\iff}$ Es gibt $d \in D$, so dass für $I' = (D, Q_I, x \leftarrow d)$ gilt $I'(Q(x) \to \forall x \, Q(x)) = 1 \iff Q_I(d) = 0$ oder $I'(\forall x \, Q(x)) = 1 \iff$ Es gibt $d \in D$ mit $Q_I(d) = 0$ oder $Q_I(d) = 1$ für alle $d \in D$.
   d) Beispiele für äquivalente Formeln:
      - Je zwei allgemeingültige Formeln sind äquivalent.
      - $\neg\neg A \models\!\!=\!\!| A$,
      - $W \vee A \models\!\!=\!\!| A \vee W \models\!\!=\!\!| W$,

- $F \wedge A \models\!\!=\!\!\mid A \wedge F \models\!\!=\!\!\mid F$,
- $A \vee A \models\!\!=\!\!\mid A,\ A \wedge A \models\!\!=\!\!\mid A$,
- $B \wedge QvA \models\!\!=\!\!\mid Qv(B \wedge A),\ B \vee QvA \models\!\!=\!\!\mid Qv(B \vee A)$, falls Q Quantor ist und v in B nicht frei vorkommt,
- $\forall v\ A \models\!\!=\!\!\mid \neg\exists v\ \neg A,\ \exists v\ A \models\!\!=\!\!\mid \neg\forall v\ \neg A$,
- $\forall x\ A(x) \to B \models\!\!=\!\!\mid \exists y\ (A(y) \to B)$, falls y weder in $A(x)$ noch in B frei vorkommt,
- $\forall x\ (A \to B) \models \forall x\ A \to \forall x\ B$

Die beiden folgenden Äquivalenzen erlauben die Umbenennung gebundener Variablen:

- $\forall v\ A \models\!\!=\!\!\mid \forall y\ A_v[y]$ und $\exists v\ A \models\!\!=\!\!\mid \exists y\ A_v[y]$, wobei y in A nicht frei vorkommt und $A_v[y]$ eine erlaubte Substitution ist. Denn für alle $d \in D$ gilt
$$I^{y,d}(A_v[y]) = (I^{y,d})^{v,d}(A) = (I^{v,d})^{y,d}(A) = I^{v,d}(A).$$
- $\forall v\ B \models\!\!=\!\!\mid B$ und $\exists v\ B \models\!\!=\!\!\mid B$, falls v nicht frei in B vorkommt.

2.15 Satz. Sei $\Gamma \subseteq$ Form und $A, B \in$ Form. Dann gelten:

1. **Deduktionstheorem:** $\Gamma, A \models B$ gilt genau dann, wenn $\Gamma \models A \to B$ gilt.

2. **Modus-Ponens-Regel:** Aus $\Gamma \models A$ und $\Gamma \models A \to B$ folgt $\Gamma \models B$

3. **Kontraposition:** $\Gamma, A \models \neg B$ gilt genau dann, wenn $\Gamma, B \models \neg A$ gilt.

4. **Generalisierungstheorem:** Kommt die Variable v in keiner Formel von Γ frei vor, dann gilt $\Gamma \models A$genau dann, wenn $\Gamma \models \forall v\ A$ gilt. Insbesondere gilt $A \models \forall v\ A$ bzw. $\models A \to \forall v\ A$, falls v nicht frei in A vorkommt.

5. **Ersetzungstheorem:** Sei $A' \in$ Form und A Teilformel von A'. Entsteht B' aus A' indem man an einigen Stellen die Teilformel A durch B ersetzt und gilt $A \models\!\!=\!\!\mid B$, so gilt auch $A' \models\!\!=\!\!\mid B'$.

Beweis. Die Beweise für 1.,2. und 3. entsprechen den jeweiligen Beweisen zur Aussagenlogik.

4. "$\Longrightarrow$": Es gelte $\Gamma \models A$ und v komme nicht frei in Γ vor. Sei ferner I eine Interpretation mit $I(v) = d \in D$ und $I \models \Gamma$. Dann gilt $I^{v,d'} \models \Gamma$ für alle $d' \in D$ und somit $I^{v,d'}(A) = 1$ für alle $d' \in D$. Also gilt gilt $\Gamma \models \forall v\ A$.

"$\Longleftarrow$" gilt trivialer weise.

5) klar. ■

2.16 Beispiele.

1.

$$\begin{aligned}
\models \exists x \forall y\; A \to \forall y \exists x\; A &\overset{\text{Ded.theor.}}{\iff} \\
\exists x \forall y\; A \models \forall y \exists x\; A &\overset{\text{Gen.theor.}}{\iff} \\
\exists x \forall y\; A \models \exists x\; A &\iff \\
\neg \forall x\, \neg \forall y\; A \models \neg \forall x\, \neg A &\overset{\text{Kon.theor.}}{\iff} \\
\forall x\, \neg A \models \forall x\, \neg \forall y\; A &\overset{\text{Gen.theor.}}{\iff} \\
\forall x\, \neg A \models \neg \forall y\; A &\iff \quad \{\forall x\, \neg A, \forall y\; A\} \text{ nicht erfüllbar}
\end{aligned}$$

2. Eine Variante des Ersetzungstheorems: Die Formel A' entstehe aus der Formel A durch Substitution einiger Vorkommen von x in A durch y. Dann gilt $\models \forall x \forall y\; (x = y \to (A \leftrightarrow A'))$.

   (z.B. $A \equiv f(x, y) = g(x)$ und $A' \equiv f(y, y) = g(x)$)

2.17 Definition. Eine Formel ist in *PKNF (Pränex Konjunktiver Normalform),* falls sie die Gestalt hat:

$$\underbrace{(\Delta v_1) \ldots (\Delta v_n)}_{\text{Präfix}} \underbrace{\{[A_{11} \vee \ldots \vee A_{1l_1}] \wedge [A_{21} \vee \ldots \vee A_{2l_2}] \wedge \ldots \wedge [A_{m1} \vee \ldots \vee A_{ml_m}]\}}_{\text{Matrix}}$$

mit

1. alle Δ sind $\exists$- oder $\forall$-Quantoren,
2. alle v_j $(1 \leq j \leq n)$ sind Variablen, die in mindestens einem A_{kl} $(1 \leq k \leq m, 1 \leq l \leq l_k)$ vorkommen und paarweise verschieden sind,
3. alle A_{kl} $(1 \leq k \leq m, 1 \leq l \leq l_k)$ sind entweder atomare oder negierte atomare Formeln.

Sind in der Matrix $\vee$ und $\wedge$ vertauscht, so ist die Formel in *PDNF (Pränex Disjunktiver Normalform).*

Beispiel: Die Formel $\forall x \exists Q \forall y\; \{[\neg p \vee x \neq a \vee x = b] \wedge [Q(y) \vee a = b]\}$ ist in PKNF.

2.18 Satz. Jede Formel $A \in$ Form lässt sich effektiv in eine äquivalente Formel in PKNF (PDNF) transformieren.

Beweis. Eine zur Formel A äquivalente Formel in PKNF (PDNF) wird durch folgendes Verfahren berechnet, dessen Schritte nacheinander auf die Formel A angewandt werden:

1. Eliminiere alle überflüssigen Quantoren der Formel, d.h. streiche für eine Variable v alle $\exists v$ und $\forall v$, in deren Geltungsbereich v nicht vorkommt: $\forall v\; B \models\!\!=\!\!\mid B$ und $\exists v\; B \models\!\!=\!\!\mid B$, falls v nicht frei in B vorkommt.

2. Umbenennung von gebundenen Variablen. Gibt es eine Teilformel $(\exists v)\ B$ bzw. $(\forall v)\ B$, wobei die Variable v im Rest der Formel frei oder gebunden vorkommt, so ersetze die Teilformel durch $(\exists v')\ B_v[v']$ bzw. $(\forall v')\ B_v[v']$, wobei v' eine Variable vom gleichen Typ wie v ist und in der Formel noch nicht vorkommt. Nach diesem Schritt sind alle quantifizierten Variablen paarweise verschieden und keine Variable tritt gleichzeitig frei und gebunden auf.

3. Elimination von logischen Verknüpfungen und Operatoren.

   a) "if_then_else": Enthält eine Teilformel B den Term $t \equiv \text{if } C \text{ then } t_1 \text{ else } t_2$, dann ersetze B durch $\text{IF } C \text{ THEN } B_1 \text{ ELSE } B_2$, wobei B_j aus B entsteht, wenn man t durch t_j ersetzt $(j = 1, 2)$.
   b) Ersetze $\rightarrow, \leftrightarrow$ und IF_THEN_ELSE durch äquivalente Formeln in $\vee, \wedge$ und $\neg$.
      Beispiel: Ersetze $B \rightarrow C$ durch $\neg B \vee C$,
      $B \leftrightarrow C$ durch $(\neg B \vee C) \wedge (\neg C \vee B)$ und
      $\text{IF } B \text{ THEN } B_1 \text{ ELSE } B_2$ durch $(B \wedge B_1) \vee (\neg B \wedge B_2)$.

4. Schiebe die Negationen so weit nach innen, bis jedes $\neg$ unmittelbar vor einer atomaren Formel steht:

$$\begin{aligned} \neg(\forall v)\ A &\models\!\!\!=\!\!\!\mid (\exists v)\ \neg A, \\ \neg(\exists v)\ A &\models\!\!\!=\!\!\!\mid (\forall v)\ \neg A, \\ \neg(A \wedge B) &\models\!\!\!=\!\!\!\mid \neg A \vee \neg B, \\ \neg(A \vee B) &\models\!\!\!=\!\!\!\mid \neg A \wedge \neg B, \\ \neg\neg A &\models\!\!\!=\!\!\!\mid A \end{aligned}$$

5. Schiebe die Quantoren nach außen: Für $\star \in \{\wedge, \vee\}$ gilt: $(\exists v)\ A \star B \models\!\!\!=\!\!\!\mid \exists v\ (A \star B)$ und $(\forall v)\ A \star B \models\!\!\!=\!\!\!\mid \forall v\ (A \star B)$, falls v in B nicht frei vorkommt. Dass v in B nicht frei vorkommt, wird durch Schritt 2 gewährleistet.

6. Bringe die Matrix der Formel in KNF (bzw. DNF), d.h. ersetze $(A \wedge B) \vee C$ durch $(A \vee C) \wedge (B \vee C)$ und $C \vee (A \wedge B)$ durch $(C \vee A) \wedge (C \vee B)$.

Da alle Umformungen die Äquivalenz erhalten, ist die nach Schritt 6 entstanden Formel äquivalent zu Formel A und in PKNF (bzw. PDNF). ■

2.19 Beispiel und Bemerkung.

1. Umformung der Formel

$$\forall x\ [(\forall y\ p(x) \vee \forall z\ q(z, y)) \rightarrow \neg \forall y\ r(x, y)]$$

   in eine äquivalente Formel in PKNF:

   Eliminieren überflüssiger Quantoren

$$\forall x\ [(p(x) \vee \forall z\, q(z, y)) \rightarrow \neg \forall y\ r(x, y)]$$

Umbenennen der gebundenen Variablen

$$\forall x\ [(p(x) \vee \forall z\ q(z,y)) \to \neg\forall y'\ r(x,y')]$$

Eliminieren von Verknüpfungen und Operatoren

$$\forall x\ [\neg(p(x) \vee \forall z\ q(z,y)) \vee \neg\forall y'\ r(x,y')]$$

Negation nach innen

$$\forall x\ [(\neg p(x) \wedge \exists z\ \neg q(z,y)) \vee \forall y'\ \neg r(x,y')]$$

Quantoren nach außen

$$\forall x \exists z \exists y'\ [(\neg p(x) \wedge \neg q(z,y)) \vee \neg r(x,y')]$$

Matrix in KNF

$$\forall x \exists z \exists y'\ [(\neg p(x) \vee \neg r(x,y')) \wedge (\neg q(z,y) \vee \neg r(x,y'))]$$

Eine dazu äquivalente Formel in PDNF ist

$$\forall x \exists z \exists y'\ [(\neg p(x) \wedge \neg q(z,y)) \vee \neg r(x,y')].$$

2. Es kann mehrere Formeln in PKNF geben, die äquivalent zur Ausgangsformel sind. Beispielsweise ist die Reihenfolge der Quantoren und Variablen nicht eindeutig bestimmt.

Die nächsten Sätze betreffen die Frage nach der Entscheidbarkeit der Sprachen der Logik: Gegeben eine Sprache L der Logik. Ist die Menge der allgemeingültigen Formeln $\{A \in \mathrm{Form}_L \mid\ \models A\}$ rekursiv entscheidbar?

Wie bereits gesehen ist diese Menge für die Sprachen der Aussagenlogik, der Quantifizierten Aussagenlogik sowie des Gleichheitskalküls entscheidbar. Es zeigt sich, dass das für allgemeine Sprachen erster Stufe und erst recht für Sprachen höherer Stufen nicht der Fall ist.

2.20 Satz. Die Allgemeingültigkeit für die Prädikatenlogik erster Stufe ist im allgemeinen unentscheidbar.

Genauer: Es gibt eine Menge von Formeln in den Konstanten a (0-stellig), f_0, f_1 (1-stellig) und der Prädikatskonstanten p (2-stellig), deren Allgemeingültigkeit unentscheidbar ist.

Beweis. Man beweist die Unentscheidbarkeit, indem man zeigt, dass sich das Post'sche Korrespondenz Problem (PCP) auf das Entscheidungsproblem der Prädikatenlogik erster Stufe reduzieren lässt. Das heißt, es existiert eine Turing-Maschine, die bei Eingabe eines Post'schen Systems S über dem Alphabet $\Sigma = \{0,1\}$ als Ausgabe eine prädikatenlogische Formel W_S erster Stufe liefert, die genau dann allgemeingültig ist, wenn das PCP S eine

Lösung hat. Da das Post'sche Korrespondenz Problem aber nicht entscheidbar ist, ist die Allgemeingültigkeit für die Prädikatenlogik erster Stufe auch nicht entscheidbar.

Zur Konstruktion der Formel W_S werden nur Konstantensymbole benötigt: Eine Individuenkonstante a, zwei einstellige Funktionskonstanten f_0 und f_1 und eine binäre Prädikatskonstante p. Zur Beschreibung der Formeln W_S wird eine Abkürzung eingeführt: Ein Term der Form $f_{\sigma_m}(\ldots(f_{\sigma_2}(f_{\sigma_1}(x)))\ldots)$, $\sigma_i \in \{0,1\}$ $(1 \leq i \leq m)$, wird geschrieben als $f_{\sigma_1\sigma_2\ldots\sigma_m}(x)$. Sei

$$S = \{(\alpha_1, \beta_1), (\alpha_2, \beta_2), \ldots, (\alpha_n, \beta_n)\},$$

$n \geq 1$, ein PCP über $\Sigma = \{0,1\}$, dann lautet die Formel W_S der Prädikatenlogik erster Stufe:

$$\left[\bigwedge_{j=1}^{n} p(f_{\alpha_j}(a), f_{\beta_j}(a)) \wedge \forall x \forall y \left(p(x,y) \rightarrow \bigwedge_{j=1}^{n} p(f_{\alpha_j}(x), f_{\beta_j}(y))\right)\right] \rightarrow \exists z\; p(z,z)$$

Beh.: Das System S hat genau dann eine Lösung, wenn W_S allgemeingültig ist.

"$\Longleftarrow$": Angenommen W_S ist allgemeingültig. Sei I eine Interpretation mit Definitionsbereich $\{0,1\}^\star$, der Menge aller Worte über dem Alphabet $\{0,1\}$, die a als das leere Wort, f_0 als $f_0(x) = x0$ (Konkatenation von x mit Zeichen 0) und f_1 als $f_1(x) = x1$ (Konkatenation von x mit 1) interpretiert. Unter der Interpretation I sei $p(x,y)$ genau dann wahr, wenn gilt $x = \alpha_{j_1}\alpha_{j_2}\ldots\alpha_{j_m}$ und $y = \beta_{j_1}\beta_{j_2}\ldots\beta_{j_m}$ für eine nichtleere Folge $j_1, j_2, \ldots, j_m$ $(1 \leq j_i \leq n)$. Da W_S allgemeingültig ist, muss W_S wahr sein unter I. Da aber die Voraussetzung in der Formel W_S (d.h. die Teilformel vor dem Implikationszeichen) wahr ist, muss auch die Konsequenz $\exists z\; p(z,z)$ wahr sein unter I. Das heißt, es gibt eine nichtleere Folge $j_1, j_2, \ldots, j_m$ $(1 \leq j_i \leq n)$ so, dass $\alpha_{j_1}\alpha_{j_2}\ldots\alpha_{j_m} = \beta_{j_1}\beta_{j_2}\ldots\beta_{j_m}$. Mit anderen Worten, das PCP S hat eine Lösung.

"$\Longrightarrow$": Angenommen das PCP S hat die Lösung $j_1, j_2, \ldots, j_m$ (d.h. $\alpha_{j_1}\alpha_{j_2}\ldots\alpha_{j_m} = \beta_{j_1}\beta_{j_2}\ldots\beta_{j_m}$). Es ist zu zeigen, dass dann jede Interpretation I, die die Voraussetzung in der Formel W_S wahr macht, auch die Konsequenz wahr macht. Dann ist W_S allgemeingültig. Sei daher I eine beliebige Interpretation, unter der

$$\bigwedge_{j=1}^{n} p(f_{\alpha_j}(a), f_{\beta_j}(a)) \tag{1}$$

und

$$\forall x \forall y \left[p(x,y) \rightarrow \bigwedge_{j=1}^{n} p(f_{\alpha_j}(x), f_{\beta_j}(y))\right] \tag{2}$$

wahr ist. Aus (1) und wiederholtem Anwenden von (2) lässt sich folgern, dass $p(f_{\alpha_{j_1}\alpha_{j_2}\ldots\alpha_{j_m}}(a), f_{\beta_{j_1}\beta_{j_2}\ldots\beta_{j_m}}(a))$ wahr ist unter I. Wegen der Voraussetzung $\alpha_{j_1}\alpha_{j_2}\ldots\alpha_{j_m} = \beta_{j_1}\beta_{j_2}\ldots\beta_{j_m}$ impliziert das aber, dass die Konsequenz von W_S, nämlich $\exists z\; p(z,z)$ wahr ist unter I.

■

Beispiel. $S = ((0,000),(0100,01),(001,1))$ hat die Lösung $j_1 = 1, j_2 = 3$ $(0001 = 0001)$. Die entsprechende Formel W_S lautet:

$$\Big[p(f_0(a), f_{000}(a)) \wedge p(f_{0100}(a), f_{01}(a)) \wedge p(f_{001}(a), f_1(a))$$

$$\wedge \bigwedge_{1 \leq j \leq 3} \Big(p(x,y) \to p(f_{\alpha_j}(x), f_{\beta_j}(x))\Big)\Big] \to \exists z\, p(z,z)$$

2.21 Hauptsätze der Prädikatenlogik erster Stufe. Sei L eine prädikatenlogische Sprache erster Stufe.

1. Die Menge der allgemeingültigen Formeln $\{A \mid \models A\}$ über L ist *rekursiv aufzählbar.* Es gibt ein deduktives System für L, dessen Theoreme genau die allgemeingültigen Formeln von L sind.

2. *Kompaktheitssatz* für die Prädikatenlogik erster Stufe. Sei $\Sigma \subseteq \mathrm{Form}_L$. Σ ist genau dann erfüllbar, wenn jede endliche Teilmenge von Σ erfüllbar ist.

3. Seien $\Sigma \subseteq \mathrm{Form}_L$ und $A \in \mathrm{Form}_L$. Genau dann gilt $\Sigma \models A$, wenn es eine endliche Teilmenge Σ_0 von Σ mit $\Sigma_0 \models A$ gibt.

4. *Satz von Löwenheim-Skolem.* $\Sigma \subseteq \mathrm{Form}_L$ ist genau dann erfüllbar, wenn es eine Interpretation $I = (D, I_c, I_v)$, mit abzählbaren oder endlichen D gibt, die Σ erfüllt.

Beweis. Zu 1., 2. und 3. gebe man ein deduktives System für die Prädikatenlogik erster Stufe an (später).
4. Informell: Σ enthält nur abzählbar viele Funktions- und Prädikatskonstanten. O.B.d.A. seien alle Formeln abgeschlossen (notfalls existentiell). Sei $I = (D, I_v)$ eine Interpretation. D muss für die u.u. unterschiedlichen Interpretationen aller möglichen Terme Werte bereithalten. Idee: Wähle als D die abzählbare Menge Term der Terme ($\approx$ "Herbrand Universum"). ■

2.22 Satz (Satz von Gödel)**.** Satz 2.21 gilt nicht für Sprachen der Prädikatenlogik zweiter Stufe. Insbesondere gilt:

1. Die Menge der allgemeingültigen Formeln für Sprachen der Prädikatenlogik zweiter Stufe ist nicht rekursiv aufzählbar.

2. Es gibt kein deduktives System, dessen Theoreme die Menge der allgemeingültigen Formeln zweiter Stufe sind.

3. Es gibt erfüllbare Mengen, die keine abzählbaren Modelle haben.

Beweis. s. Yasuhara [1971]

2.3 Das deduktive System $\mathcal{F}$

Wir betrachten nun Sprachen der ersten Stufe mit Formeln in $\neg, \rightarrow, \forall, =$. Die übrigen Junktoren und der Existenzquantor dienen nur als "Abkürzungen" für äquivalente Formeln.

2.23 Definition. *Deduktives System* für eine Sprache der Prädikatenlogik erster Stufe. Sei L eine Sprache erster Stufe mit Formeln in $\neg, \rightarrow, \forall, =$. Das deduktive System $\mathcal{F} = (\mathrm{Ax}, R)$ ist bestimmt durch die Menge Ax von Axiomen und R von Regeln.

Ax enthält alle Generalisierungen (Eine Formel A ist Generalisierung der Formel B, falls $A \equiv \forall x_1 \ldots \forall x_n\ B$, wobei x_j eine Variable ist ($0 \leq j \leq n$).) folgender Axiome:

Ax1: $A \rightarrow (B \rightarrow A)$

Ax2: $(A \rightarrow (B \rightarrow C)) \rightarrow ((A \rightarrow B) \rightarrow (A \rightarrow C))$

Ax3: $(\neg B \rightarrow \neg A) \rightarrow (A \rightarrow B)$

Ax4: $\forall x\ A \rightarrow A_x[t]$, falls $A_x[t]$ erlaubt

Ax5: $\forall x\ (A \rightarrow B) \rightarrow (\forall x\ A \rightarrow \forall x\ B)$

Ax6: $A \rightarrow \forall x\ A$, falls x nicht frei in A vorkommt

Ax7: $x = x$

Ax8: $x = y \rightarrow (A \rightarrow A')$, wobei A' aus A entsteht, indem man x an einigen Stellen in A durch y ersetzt

Bem.: Bei den Axiomen Ax1, Ax2 und Ax3 handelt es sich um Tautologien. Kommt x in A nicht frei vor, kann Ax5 geschrieben werden $\forall x\ (A \rightarrow B) \rightarrow (A \rightarrow \forall x\ B)$.

R enthält alle Regeln, die durch das folgende Regelschema gegeben sind:

$$\frac{A, A \rightarrow B}{B} \quad \text{(Modus Ponens)}$$

Man schreibt $\vdash_{\mathcal{F}} A$ (oder kurz $\vdash A$), falls A in $\mathcal{F}$ herleitbar ist bzw. $\Sigma \vdash_{\mathcal{F}} A$, (oder kurz $\Sigma \vdash A$, falls A in $\mathcal{F}$ aus Σ herleitbar ist.

Beachte: Alle Axiome sind allgemeingültige Formeln, d.h. die Axiome sind korrekt. Sind die Formeln A und $A \rightarrow B$ allgemeingültig, so ist auch B allgemeingültig. Ist die Formel A allgemeingültig, dann ist auch $\forall x\ A$ (eine Generalisierung) allgemeingültig. Deshalb folgt aus $\vdash A$ stets $\models A$ und aus $\Sigma \vdash A$ folgt $\Sigma \models A$. Das heißt $\mathcal{F}$ ist korrekt!

Man kann andere Axiomatisierungen angeben. Betrachte das deduktive System $\mathcal{F}' = (\mathrm{Ax}', R')$, wobei Ax' alle Generalisierungen der Axiome Ax1,..,Ax8 umfasst. Die Regelmenge R' enthalte zusätzlich zum Modus-Ponens noch die *Generalisierungsregel*:

$$\frac{A}{\forall x\ A}$$

Dann gilt für $A \in$ Form und $\Sigma \subseteq$ Form :

$\vdash_{\mathcal{F}} A$ genau dann, wenn $\vdash_{\mathcal{F}'} A$ gilt, also genau dann, wenn $\models A$ gilt und aus $\Sigma \vdash_{\mathcal{F}} A$ folgt $\Sigma \vdash_{\mathcal{F}'} A$ Gilt zusätzlich, dass Σ nur abgeschlossene Formeln enthält, dann gilt auch $\Sigma \vdash_{\mathcal{F}'} A$ genau dann, wenn $\Sigma \models A$ gilt.

Für eine Menge $\Gamma \subseteq$ Form folgt wegen der Generalisierungsregel jedoch nicht immer aus $\Gamma \vdash_{\mathcal{F}'} A$ auch $\Gamma \models A$.

Denn beispielsweise gilt für alle einstelligen Prädikatskonstanten p zwar $p(x) \vdash_{\mathcal{F}'} \forall x\, p(x)$, aber $p(x) \not\models \forall x\, p(x)$.

2.24 Bemerkung. Alle Formeln, die sich aus Ax1, Ax2, Ax3 und Modus-Ponens herleiten lassen, sind allgemeingültig (genauer: Tautologien). Ist andererseits A eine Tautologie der Aussagenlogik und entsteht A' aus A durch Substitution der aussagenlogischen Konstanten durch Formeln erster Stufe, so gilt $\vdash_{\mathcal{F}} A'$.

2.25 Beispiele.

1. $\vdash_{\mathcal{F}'} \forall x\, (p(x) \to \exists y\, p(y))$

   Beweis:

   $$\begin{array}{ll}
   B_1 \equiv \forall x\, [(\forall y\, \neg p(y) \to \neg p(x)) \to (p(x) \to \neg\forall y\, \neg p(y))] & \text{(Ax3,Gen)} \\
   B_2 \equiv \forall x\, ((\forall y\, \neg p(y) \to \neg p(x)) \to (p(x) \to \neg\forall y\, \neg p(y))) \to & \\
   \qquad [\forall x\, (\forall y\, \neg p(y) \to \neg p(x)) \to \forall x\, (p(x) \to \neg\forall y\, \neg p(y))] & \text{(Ax5)} \\
   B_3 \equiv \forall x\, (\forall y\, \neg p(y) \to \neg p(x)) \to \forall x\, (p(x) \to \neg\forall y\, \neg p(y)) & \text{(MP)} \\
   B_4 \equiv \forall x\, (\forall y\, \neg p(y) \to \neg p(x)) & \text{(Ax4,Gen)} \\
   B_5 \equiv \forall x\, (p(x) \to \exists y\, p(y)) & \text{(MP)}
   \end{array}$$

2. $\vdash_{\mathcal{F}} \forall x\, A \to \exists x\, A$.

   Beweis:

   $$\begin{array}{ll}
   \vdash \forall x\, \neg A \to \neg A & \text{(Ax4)} \\
   \vdash (\forall x\, \neg A \to \neg A) \to (A \to \neg\forall x\, \neg A) & \text{(Taut)} \\
   \vdash A \to \neg\forall x\, \neg A & \text{(MP)} \\
   \vdash \forall x\, A \to A & \text{(Ax4)} \\
   \vdash (\forall x\, A \to A) \to & \\
   \quad ((A \to \neg\forall x\, \neg A) \to (\forall x\, A \to \neg\forall x\, \neg A)) & \text{(Taut)} \\
   \vdash (A \to \neg\forall x\, \neg A) \to (\forall x\, A \to \neg\forall x\, \neg A) & \text{(MP)} \\
   \vdash \forall x\, A \to \exists x\, A & \text{(MP)}
   \end{array}$$

3. $\vdash_{\mathcal{F}'} \forall x\, A \to \forall z\, A_x[z]$, falls z nicht in A vor kommt.

   Beweis:

   $$\begin{array}{ll}
   \vdash \forall x\, A \to A_x[z] & \\
   \vdash \forall z\, (A \to A_x[z]) \to (A \to \forall z\, A_x[z]) & \text{(Ax5, Bem)} \\
   \vdash \forall z\, (\forall x\, A \to A_x[z]) & \text{(Gen)} \\
   \vdash \forall x\, A \to \forall z\, A_x[z] & \text{(MP)}
   \end{array}$$

Es soll nun der Frage nachgegangen werden, ob sich die Sätze aus 2.15 auch auf die deduktiven Systeme $\mathcal{F}$ und $\mathcal{F}'$ anwenden lassen.

2.26 Satz. Seien $\Gamma \subseteq \text{Form}, A, B \in \text{Form}$.

1. *Deduktionstheorem*:
   a) Genau dann gilt $\Gamma \vdash_{\mathcal{F}} A \to B$, wenn $\Gamma, A \vdash_{\mathcal{F}} B$ gilt.
   b) Genau dann gilt $\Gamma \vdash_{\mathcal{F}'} A \to B$, wenn $\Gamma, A \vdash_{\mathcal{F}'} B$ gilt und die Generalisierung nicht auf eine in A frei vorkommende Variable angewandt wurde.
   c) Genau dann gilt $\Gamma, A \vdash_{\mathcal{F}'} B$, wenn $\Gamma \vdash_{\mathcal{F}'} \tilde{A} \to B$ gilt, wobei $\tilde{A}$ ein universeller Abschluss von A ist.
2. *Generalisierungstheorem:*
   a) Falls $\Gamma \vdash_{\mathcal{F}} A$ gilt und x eine Variable ist, die in Γ nicht frei vorkommt, dann gilt $\Gamma \vdash_{\mathcal{F}} \forall x\, A$.
   b) Genau dann gilt $\Gamma \vdash_{\mathcal{F}'} A$, wenn $\Gamma \vdash_{\mathcal{F}'} \forall x\, A$ gilt.
3. *Kontrapositionstheorem:* Genau dann gilt $\Gamma, A \vdash_{\mathcal{F}} \neg B$, wenn $\Gamma, B \vdash_{\mathcal{F}} \neg A$ gilt.

Beweis. 1.(a).
"$\Longrightarrow$": Falls $\Gamma \vdash_{\mathcal{F}} A \to B$, dann gibt es dafür einen Beweis $C_1 C_2 \dots C_n$ der Form

$$\begin{array}{ll} C_1 & \Gamma \\ \vdots & \vdots \\ C_n & A \to B \end{array}$$

Dann gibt es aber auch einen Beweis $D_1 D_2 \dots D_{n+1}$für $\Gamma, A \vdash_{\mathcal{F}} A \to B$, nämlich

$$\begin{array}{ll} D_1 & A \\ D_2 & \Gamma \\ \vdots & \vdots \\ D_{n+1} & A \to B \end{array}$$

mit $D_{j+1} \equiv C_j$ für $1 \leq j \leq n$.
Somit:

$$\begin{array}{lll} B_1 & \Gamma \vdash A \to B & \text{Voraussetzung} \\ B_2 & \Gamma, A \vdash A \to B & \text{s.o.} \\ B_3 & \Gamma, A \vdash A & A \text{ ist aus } A \text{ herleitbar} \\ B_4 & \Gamma, A \vdash B & \text{MP}(B_2, B_3) \end{array}$$

"$\Longleftarrow$": Induktion über die Zahl n der Beweisschritte für $\Gamma, A \vdash B$:
$n = 1$: B ist selbst Axiom oder Hypothese. Falls B ein Axiom ist oder $B \in \Gamma$, dann

$$\begin{array}{lll} B_1 & \Gamma \vdash B & \text{Axiom oder Hypothese} \\ B_2 & \Gamma \vdash B \to (A \to B) & \text{Ax1} \\ B_3 & \Gamma \vdash A \to B & \text{MP}(B_1, B_2) \end{array}$$

Ist $B \equiv A$, dann (da $\vdash_{\mathcal{F}} A \to A$ (zeigen!))

$$B_1 \qquad \Gamma \vdash A \to A$$

$n \to n+1$: Für den Fall, dass B Axiom oder Hypothese ist, wurde der Beweis bereits angegeben. Ansonsten resultiert B im Beweis für $\Gamma, A \vdash B$ aus vorangegangenen Schritten durch Anwendung der Modus-Ponens-Regel. Es stellt sich also folgende Situation dar:

$$\begin{array}{rl} \vdots & \vdots \\ \text{Schritt } j: & \Gamma, A \vdash C \\ \vdots & \vdots \\ \text{Schritt } k: & \Gamma, A \vdash C \to B \\ \vdots & \vdots \\ \text{Schritt } n+1: & \Gamma, A \vdash B \end{array}$$

mit $j, k \leq n$. Da für den j-ten und k-ten Schritt die Induktionsvoraussetzung bereits gilt, erhält man den folgenden Beweis für $\Gamma \vdash A \to B$:

B_1	$\Gamma \vdash A \to C$	nach Induktionsvoraussetzung
B_2	$\Gamma \vdash A \to (C \to B)$	nach Induktionsvoraussetzung
B_3	$\Gamma \vdash (A \to (C \to B)) \to ((A \to C) \to (A \to B))$	Ax2
B_4	$\Gamma \vdash A \to B$	2mal MP

1.(b).

Der Beweis lässt sich analog zu dem Beweis aus 1.(a). führen. Es muss im Beweis "$\Longleftarrow$" beim Induktionsschritt jedoch zusätzlich der Fall betrachtet werden, dass der $n+1$-te (letzte) Schritt im Beweis von $\Gamma, A \vdash B$ durch die Anwendung der Generalisierungsregel erfolgte, d.h.

$$\begin{array}{rl} \vdots & \vdots \\ \text{Schritt } j: & \Gamma, A \vdash C \\ \vdots & \vdots \\ \text{Schritt } n+1: & \Gamma, A \vdash \forall x\, C \end{array}$$

mit $B \equiv \forall x\, C$, $j \leq n$ und x kommt in A nicht frei vor. Man erhält dann als Beweis für $\Gamma \vdash A \to B$:

B_1	$\Gamma \vdash A \to C$	nach Induktionsvoraussetzung
B_2	$\Gamma \vdash \forall x\, (A \to C)$	Gen(B_1)
Da x in A nicht frei vorkommt:		
B_3	$\Gamma \vdash \forall x\, (A \to C) \to (A \to \forall x\, C)$	Ax5
B_4	$\Gamma \vdash A \to \forall x\, C$	MP(B_2, B_3)

1.(c).

Der Beweis für die Variante $\Gamma, A \vdash_{\mathcal{F}'} B$ genau dann, wenn $\Gamma \vdash_{\mathcal{F}'} \tilde{A} \to B$, wobei $\tilde{A}$ universeller Abschluss von A ist :

"$\Longleftarrow$":

B_1	$\Gamma \vdash \tilde{A} \to B$	Voraussetzung
B_2	$\Gamma, A \vdash \tilde{A} \to B$	vgl. 1.(a)$\Rightarrow$
B_3	$\Gamma, A \vdash A$	A ist aus A herleitbar
B_4	$\Gamma, A \vdash \forall x_1\, A$	Anwendung der
B_5	$\Gamma, A \vdash \forall x_2 \forall x_1\, A$	Generalisierungsregel bis alle freien
	$\vdots$	Vorkommen von Variablen x_j in A
B_{3+k}	$\Gamma, A \vdash \tilde{A}$	$(1 \leq j \leq k)$ gebunden sind
B_{3+k+1}	$\Gamma, A \vdash B$	MP(B_2, B_{3+k})

"$\Longrightarrow$": Induktion über die Zahl n der Beweisschritte für $\Gamma, A \vdash B$:

$n = 1$: B ist selbst Axiom oder Hypothese. Falls B ein Axiom ist oder $B \in \Gamma$, dann

B_1	$\Gamma \vdash B$	Axiom oder Hypothese
B_2	$\Gamma \vdash B \to (\tilde{A} \to B)$	Ax1
B_3	$\Gamma \vdash \tilde{A} \to B$	MP(B_1, B_2)

Ist $B \equiv A$, dann (da $\vdash \tilde{A} \to A$ (zeigen!)):

$$B_1 \quad \Gamma \vdash \tilde{A} \to A$$

$n \to n+1$: Für den Fall, dass B Axiom oder Hypothese ist, wurde die Aussage bereits gezeigt. Ansonsten resultiert B aus vorangegangenen Schritten im Beweis für $\Gamma, A \vdash B$ durch Anwendung der Modus-Ponens-Regel oder der Generalisierungsregel. Es stellt sich also eine der beiden folgenden Situationen ein:

$$(\star) \qquad \begin{array}{rl} \vdots & \vdots \\ \text{Schritt } j: & \Gamma, A \vdash C \\ \vdots & \vdots \\ \text{Schritt } k: & \Gamma, A \vdash C \to B \\ \vdots & \vdots \\ \text{Schritt } n+1: & \Gamma, A \vdash B \end{array}$$

mit $j, k \leq n$ oder

$$(\star\star) \qquad \begin{array}{rl} \vdots & \vdots \\ \text{Schritt } j: & \Gamma, A \vdash C \\ \vdots & \vdots \\ \text{Schritt } n+1: & \Gamma, A \vdash \forall x\, C \end{array}$$

mit $B \equiv \forall x\, C,\ j \leq n$.
Als Beweis für $\Gamma \vdash \tilde{A} \to B$ erhält man im Fall ($\star$):

B_1	$\Gamma \vdash \tilde{A} \to C$	nach Induktionsvoraussetzung
B_2	$\Gamma \vdash \tilde{A} \to (C \to B)$	nach Induktionsvoraussetzung
B_3	$\Gamma \vdash (\tilde{A} \to (C \to B)) \to ((\tilde{A} \to C) \to (\tilde{A} \to B))$	Ax2
B_4	$\Gamma \vdash \tilde{A} \to B$	2mal MP

Im Fall ($\star\star$) ergibt sich folgender Beweis:

B_1	$\Gamma \vdash \tilde{A} \to C$	nach Induktionsvoraussetzung
B_2	$\Gamma \vdash \forall x\, (\tilde{A} \to C)$	Gen(B_1)

Da in $\tilde{A}$ keine Variable frei vorkommt (also auch x nicht) :

B_3	$\Gamma \vdash \forall x\, (\tilde{A} \to C) \to (\tilde{A} \to \forall x\, C)$	Ax5
B_4	$\Gamma \vdash \tilde{A} \to \forall x\, C$	MP(B_2, B_3)

2.(a).
Sei $B_0, \ldots, B_m$ ein Beweis für $\Gamma \vdash_{\mathcal{F}} A$. Zeige: $\Gamma \vdash \forall x\, B_j$ ($j = 0, \ldots, m$).
Induktion nach j:

- $B_j \in$ Ax $\surd$
- $B_j \in \Gamma$

B_j	Voraussetzung
$B_j \to \forall x\, B_j$	Ax6, da x nicht frei in Γ
$\forall x\, B_j$	MP

- B_j entsteht durch MP aus B_h und B_k, d.h. $B_k \equiv B_h \to B_j$, dann lässt sich folgender Beweis formulieren:

C_1	$\Gamma \vdash \forall x\, B_h$	Induktionsvoraussetzung
C_2	$\Gamma \vdash \forall x\, (B_h \to B_j)$	Induktionsvoraussetzung
C_3	$\Gamma \vdash \forall x (B_h \to B_j) \to (\forall x\, B_h \to \forall x\, B_j)$	Ax5
C_4	$\Gamma \vdash \forall x\, B_h \to \forall x\, B_j$	MP(C_2, C_3)
C_5	$\Gamma \vdash \forall x\, B_j$	MP(C_1, C_4)

2b.
"$\Longrightarrow$":

B_1	$\Gamma \vdash A$	
B_2	$\Gamma \vdash \forall x\, A$	Gen(B_1)

"$\Longleftarrow$":

B_1	$\Gamma \vdash \forall x\, A$	
B_2	$\Gamma \vdash \forall x\, A \to A_x[x]$	Ax4, Substitution erlaubt
B_3	$\Gamma \vdash A$	MP(B_1, B_2)

3.
Mit der Voraussetzung $\vdash_{\mathcal{F}} (D \to \neg E) \to (E \to \neg D)$ (Variante von Ax3 (zeigen!)):

"$\Longrightarrow$": Es gelte $\Gamma, A \vdash_{\mathcal{F}} \neg B$. Dann folgt aus dem Deduktionstheorem $\Gamma \vdash_{\mathcal{F}} A \to \neg B$. Ein Beweis für $\Gamma \vdash_{\mathcal{F}} B \to \neg A$ ist dann:

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

Die Behauptung $\Gamma, B \vdash_{\mathcal{F}} \neg A$ folgt nun mit dem Deduktionstheorem.

"$\Longleftarrow$": symmetrisch zu "$\Longrightarrow$" ■

2.27 Definition. Sei $\Gamma \subseteq$ Form. Γ heißt *konsistent*, falls es kein $A \in$ Form gibt, mit $\Gamma \vdash_{\mathcal{F}} A$ und $\Gamma \vdash_{\mathcal{F}} \neg A$.

Beachte: Γ ist genau dann konsistent, wenn jede endliche Teilmenge von Γ konsistent ist. Ist Γ *inkonsistent* (d.h. nicht konsistent), dann gilt $\Gamma \vdash_{\mathcal{F}} A$ für jede Formel A.

Genau dann ist $\Gamma \cup \{\neg A\}$ inkonsistent, wenn $\Gamma \vdash_{\mathcal{F}} A$ gilt; genau dann ist $\Gamma \cup \{A\}$ inkonsistent, wenn $\Gamma \vdash_{\mathcal{F}} \neg A$ gilt.

Ist Γ inkonsistent, so ist Γ nicht erfüllbar: Sei nämlich A eine Formel mit $\Gamma \vdash_{\mathcal{F}} A$ und $\Gamma \vdash_{\mathcal{F}} \neg A$ und sei I eine Interpretation, die Γ erfüllt, dann folgt wegen $\Gamma \models A$ und $\Gamma \models \neg A$, dass I sowohl A als auch $\neg A$ erfüllt. Widerspruch.

Die Menge der allgemeingültigen Formeln ist konsistent.

2.28 Satz (Gödel)**.** *Vollständigkeit der Axiomatisierung:* Seien $A \in$ Form, $\Sigma \subseteq$ Form, dann gilt:

1. äquivalent sind
   - $\models A$,
   - $\vdash_{\mathcal{F}} A$ und
   - $\vdash_{\mathcal{F}'} A$.
2. Σ ist genau dann konsistent, wenn Σ erfüllbar ist.
3. Genau dann gilt $\Sigma \vdash_{\mathcal{F}} A$, wenn $\Sigma \models A$ gilt.

Beweis. s. Yasuhara [1971] bzw. Enderton [1972]

2.29 Definition. Sei L eine Sprache erster Stufe. $\Gamma \subseteq \mathrm{Form}_L$ heißt *logische Theorie erster Stufe*, falls Γ abgeschlossen ist gegenüber logischer Folgerung, d.h. für alle $A \in$ Form gilt $\Gamma \models A$ genau dann, wenn $A \in \Gamma$.

2.30 Bemerkungen. Sei L Sprache erster Stufe.

1. $T = \{A \mid A \text{ allgemeingültig}\}$ ist eine Theorie. Sie ist in jeder Theorie über L enthalten.

2. $T_\Sigma = \{A \mid \Sigma \models A\}$ ist eine Theorie ($\Sigma \subseteq \text{Form}$).

   Angenommen $T_\Sigma \models A$ und I sei eine Interpretation, die Σ erfüllt, dann ist $I(T_\Sigma) = 1$ und $I(A) = 1$, d.h. $\Sigma \models A$ und $A \in T_\Sigma$. T_Σ ist die von Σ erzeugte Theorie erster Stufe.

3. Ist T eine Theorie, dann gilt $T \vdash A$ genau dann, wenn $A \in T$.

   Insbesondere: Ist T inkonsistent, dann $T = \text{Form}$.

4. Sei R eine Struktur in L (d.h. es genügt Interpretationen $I = (D, I_c)$ zu betrachten), dann ist

   $$T_R := \{A \in \text{Form} \mid R \models \tilde{A},\ \text{wobei } \tilde{A} \text{ universeller Abschluss von } A \text{ ist}\}$$

   eine Theorie.

5. $T \subseteq \text{Form}$ ist genau dann eine Theorie, wenn $\{A \mid\, \models A\} \subseteq T$ und T abgeschlossen gegenüber Modus Ponens ist.

2.31 Definition. Sei T eine Theorie.

1. T heißt *vollständig*, falls für jede abgeschlossene Formel A gilt: $A \in T$ oder $\neg A \in T$.

2. T heißt (endlich) *rekursiv axiomatisierbar*, falls es eine (endliche) rekursive Teilmenge $\Sigma \subseteq \text{Form}$ gibt, mit $T_\Sigma = \{A \mid \Sigma \models A\} = T$.

3. T heißt *entscheidbar*, falls T eine entscheidbare Teilmenge von Form ist.

2.32 Bemerkungen.

1. T_R ist vollständig für jede Struktur R. Außerdem ist T_R konsistent.

2. T ist genau dann erfüllbar, wenn T konsistent ist.

3. Ist T rekursiv axiomatisierbar, dann ist T auch rekursiv aufzählbar.

4. Ist T vollständig, konsistent und rekursiv axiomatisierbar, dann ist T entscheidbar.

   Falls für jede abgeschlossene Formel A genau dann $A \in T$ gilt, wenn $\neg A \notin T$ gilt, so ist "$A \in T$" entscheidbar für abgeschlossene Formeln, da T rekursiv aufzählbar ist.

5. Ist T vollständig und konsistent, dann gilt $T = T_R$ für eine Struktur R.

2.33 Beispiele.

1. Th($\mathbb{N}$) ist das Relationalsystem $< \mathbb{N}; 0, S, +, \star; = >$, das die “natürliche” Interpretation der Sprache der Arithmetik über den Konstanten $0, S, +$ und $\star$ bezeichnet. (Ist $n \in \mathbb{N}$, dann hat das entsprechende Element $\tilde{n} \in L$ die Form $\tilde{n} = S(S(\ldots S(0)\ldots))$. Es gilt (Gödel):

   a) Th($\mathbb{N}$) ist nicht rekursiv!

   b) Th($\mathbb{N}$) ist nicht rekursiv axiomatisierbar!

   Eine Konsequenz ist, dass die Peano-Axiome $P_1 \ldots P_6$ keine Axiomatisierung von Th($\mathbb{N}$) bilden.

   P_1 $\forall x \forall y\, S(x) = S(y) \to x = y$
   P_2 $\forall x\, S(x) \neq 0$
   P_3 $\forall x\, x + 0 = x$
   P_4 $\forall x \forall y\, x + S(y) = S(x + y)$
   P_5 $\forall x\, x \star 0 = 0$
   P_6 $\forall x \forall y\, x \star S(y) = x \star y + x$
   P_7 $A_x[0] \to (\forall x\, (A \to A_x[S(x)]) \to \forall x\, A)$,
   falls x die einzige in Formel A frei vorkommende Variable ist.

2. Die Peano-Arithmetik (auch Arithmetik erster Stufe genannt), ist die Menge aller Formeln (erster Stufe) über der Basis $B = (\{0, 1, +, \star\}, \{<\})$, die in der Interpretation $I = (\mathbb{N}, I_c)$ allgemeingültig sind. Dabei soll I_c den Funktionssymbolen und dem Prädikatssymbol die übliche Bedeutung geben. Die Preßburger-Arithmetik entspricht der Peano-Arithmetik, es entfällt jedoch das Multiplikationssymbol.

   Es gilt: Die Peano-Arithmetik und die Preßburger-Arithmetik sind vollständige Theorien. Die Preßburger-Arithmetik ist entscheidbar und axiomatisierbar. Die Peano-Arithmetik ist nicht entscheidbar, nicht rekursiv aufzählbar und nicht axiomatisierbar.

3. Für die Theorie Th($\mathbb{R}$) ($\mathbb{R}$ bezeichnet die reellen Zahlen) über einer Sprache in den Konstanten $0, 1, +, \star$ und $<$ mit ihrer “natürlichen” Interpretation gilt:

   Th($\mathbb{R}$) ist rekursiv entscheidbar!

   Th($\mathbb{R}$) ist rekursiv axiomatisierbar!

   Eine Axiomatisierung für Th($\mathbb{R}$) besteht aus den Axiomen für Körper (+ Nullstellen für Polynome) und zusätzlich:

$$
\begin{array}{l}
\neg(x < x) \\
x < y \wedge y < z \to x < z \\
x < y \vee y = x \vee y < x \\
x < y \to x + z < y + z \\
0 < x \vee 0 < y \to 0 < x \star y \\
0 < x \to \exists y\, (y \star y = x)
\end{array}
$$

4. Sei L eine Sprache erster Stufe mit Gleichheit und dem zweistelligen Prädikatssymbol $<$. L habe keine Konstanten und Funktionssymbole. Sei DLO die zu L gehörende Theorie mit folgenden Axiomen:

$$\begin{array}{l} \forall x \forall y \forall z\, (x < y \wedge y < z \rightarrow x < z) \\ \forall x \forall y\, (x < y \vee x = y \vee y < x) \\ \forall x\, \neg(x < x) \\ \forall x \forall y \exists z\, (x < y \rightarrow (x < z \wedge z < y)) \\ \forall x \exists y \exists z\, (y < x \wedge x < z) \end{array}$$

   Interpretationen, die DLO erfüllen, sind dichte, linear geordnete Systeme ohne Endpunkte. Beispiele dafür sind die reellen Zahlen und die rationalen Zahlen, sowie offene Intervalle davon. DLO ist eine vollständige und entscheidbare Theorie [Yasuhara, 1971].

2.4 Semantische Tableaux

Im Rest dieses Kapitels werden zwei Aufzählungsverfahren für die Prädikatenlogik erster Stufe vorgestellt:

Frage: Gilt $\Sigma \models A$ für eine Formelmenge Σ und eine Formel A? Sei $\Sigma \subseteq \mathrm{Form}$. Dann ist $T_\Sigma = \{A \mid \ \Sigma \models A\}$ rekursiv aufzählbar. Gesucht wird also ein "einfaches" (effektives) Aufzählungsverfahren für die Formeln aus T_Σ. Es gibt dafür verschiedene Möglichkeiten: Zum einen legt die Möglichkeit der Axiomatisierung die Idee nahe, alle möglichen Beweise ihrer Länge nach aufzuzählen. Diese Möglichkeit ist aber normalerweise umständlich. Eine andere Möglichkeit beruht auf der Äquivalenz von $\Sigma \models A$ und "$\{\Sigma, \neg A\}$ nicht erfüllbar". Man versucht aus der Voraussetzung, dass $\{\Sigma, \neg A\}$ ein Modell hat, so schnell und so systematisch wie möglich einen Widerspruch herzuleiten, um $\Sigma \models A$ zu beweisen. Die Tableaux-Methode für die Prädikatenlogik erster Stufe beruht auf dieser Strategie.

2.34 Definition. *Tableaux* für die Prädikatenlogik erster Stufe.

Sei L eine prädikatenlogische Sprache erster Stufe in $\wedge, \vee, \neg, \rightarrow, \exists$ und $\forall$. Wie für die Aussagenlogik (vgl. 1.29) werden die Formeln der Sprache in Klassen unterteilt:

1. Atomare und negierte atomare Formeln
2. α-Formeln (analog der Aussagenlogik: $A \wedge B, \neg(A \vee B), \neg(A \rightarrow B)$ und $\neg\neg A$)
3. β-Formeln (analog zur Aussagenlogik: $\neg(A \wedge B), (A \vee B)$ und $(A \rightarrow B)$)

Mit der Einteilung in atomare, α- und β-Formeln sind jedoch nicht alle möglichen Formeln einer prädikatenlogischen Sprache erster Stufe behandelt. Für Formeln mit Quantoren werden zusätzlich die folgenden Klassen definiert:

4. γ-Formeln: Formeln der Struktur $\forall x\ A$ und $\neg\exists x\ A$ $(A \in \mathrm{Form})$
5. δ-Formeln: Formeln der Struktur $\exists x\ A$ und $\neg\forall x\ A$ $(A \in \mathrm{Form})$

Als Regeln werden neben den Regeln für α- und β-Formeln aus Definition 1.29 für γ- und δ-Formeln zusätzliche Regeln eingeführt:

<u>γ-Regeln:</u>

γ	$\forall x\ A$	$\neg\exists x\ A$
$\gamma[t]$	$A_x[t]$	$\neg A_x[t]$

dabei ist t ein beliebiger Term, der so gewählt ist, dass die Substitution erlaubt ist.

<u>δ-Regeln:</u>

δ	$\exists x\ A$	$\neg\forall x\ A$
$\delta[y]$	$A_x[y]$	$\neg A_x[y]$

dabei ist y eine Variable, mit der Einschränkung, dass y "neu" ist, d.h. y kommt noch nicht im Ast zu $A_x[y]$ (bzw. $\neg A_x[y]$) vor, und die Substitution ist erlaubt. Statt y kann auch eine neue Konstante "a" verwendet werden.

γ-Regeln und δ-Regeln werden bei der systematischen Tableau-Konstruktion wie früher α- bzw. β-Regeln angewendet. Es ist bei jeder Anwendung jedoch darauf zu achten, dass $A_x[t]$ eine erlaubte Substitution ist bzw. dass y "neu" ist.

δ-Formeln brauchen nur einmal berücksichtigt zu werden, und können dann abgeharkt werden. Dagegen können γ-Formeln nicht abgeharkt werden.

2.35 Lemma. Sei Γ eine Menge von Formeln mit Universum $\mathcal{U}$ (von Termen), für die folgendes gilt:

1. Für keine Formel A gilt $A \in \Gamma$ und $\neg A \in \Gamma$.
2. Für jede α-Formel in Γ gilt $\alpha_1, \alpha_2 \in \Gamma$.
3. Für jede β-Formel in Γ gilt $\beta_1 \in \Gamma$ oder $\beta_2 \in \Gamma$.
4. Für jede γ-Formel in Γ gilt $\gamma[t] \in \Gamma$ für alle $t \in \mathcal{U}$.
5. Für jede δ-Formel in Γ gibt es ein $t \in \mathcal{U}$ mit $\delta[t] \in \Gamma$.

Dann ist Γ erfüllbar!

Beweis. Der Beweis kann analog des Beweises zu Lemma 1.33 geführt werden. Dazu wird eine Interpretation $I = (\mathcal{U}, \ldots)$ betrachtet, mit

$$\begin{aligned} I(P(t_1, \ldots, t_n)) &= 0, \text{ falls } \neg P(t_1, \ldots t_n) \in \Gamma \text{ und} \\ I(P(t_1, \ldots, t_n)) &= 1 \text{ sonst,} \end{aligned}$$

für eine Formel P, die aus den Termen $t_1, .., t_n \in \mathcal{U}$ und Prädikatskonstanten zusammengesetzt ist.

Behauptung: $I(A) = 1$, für alle $A \in \Gamma$.

Induktion über den Aufbau von A :

Ist A atomar, eine α-Formel oder eine β-Formel, dann ist $I(A) = 1$, wenn $A \in \Gamma$ analog zum Beweis von Lemma 1.33.

Ist $(\Delta x\ A) \in \Gamma$ eine γ-Formel, $\Delta \in \{\forall, \neg\exists\}$, dann gilt für jedes $k \in \mathcal{U}$, dass $A_x[k] \in \Gamma$ und somit nach gilt Induktionsvoraussetzung $I(A_x[k]) = 1$, also ist auch $I(\Delta x\ A) = 1$.

Ist $(\Delta x\ A) \in \Gamma$ eine δ-Formel, $\Delta \in \{\exists, \neg\forall\}$, dann gibt es ein $k \in \mathcal{U}$, so dass $A_x[k] \in \Gamma$ und somit gilt nach Induktionsvoraussetzung $I(A_x[k]) = 1$. Dann ist aber auch nach Definition $I(\Delta x\ A) = 1$. ■

2.36 Bemerkung. In Sprachen, die die Gleichheit "=" enthalten, wird zusätzlich eine ϵ-Regel eingeführt:

$$\frac{t_1 = t_2}{A[t_1 \leftarrow t_2]},$$

d.h. in einer Formel A kann ein Term t_1 durch einen "gleichen" Term t_2 ersetzt werden, wenn im Ast zu A der Knoten mit der Markierung $t_1 = t_2$vorkommt.

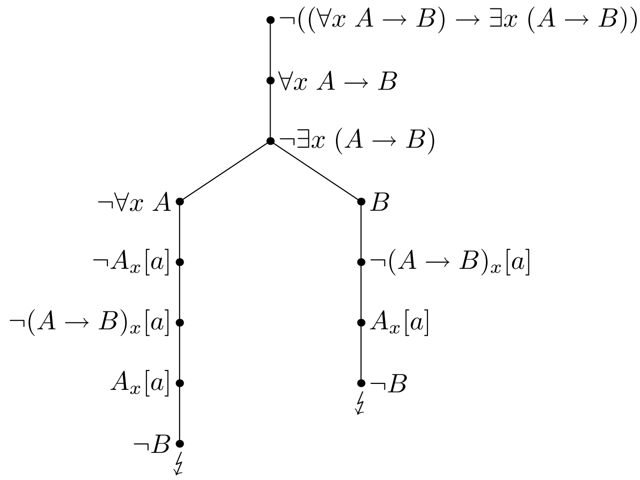
2.37 Satz. Sei L eine Sprache erster Stufe und sei $A \in \mathrm{Form}_L, \Sigma \subseteq \mathrm{Form}_L$.

1. $\models A$ gilt genau dann, wenn es gibt abgeschlossenes Tableau für $\neg A$ gibt.
2. $\Sigma \models A$ gilt genau dann, wenn es ein abgeschlossenes Tableau für $\Sigma \cup \{\neg A\}$ gibt.

Beweis. s. Smullyan [1968]

2.38 Beispiele.

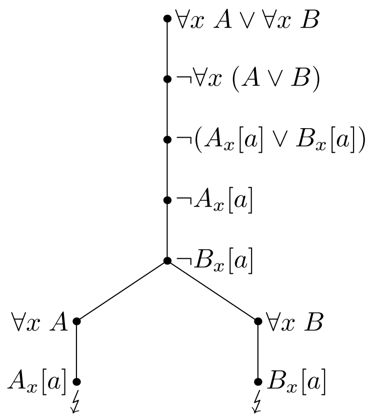
1. $\models (\forall x\ A \to B) \to \exists x\ (A \to B)$, x soll in B nicht frei vorkommen



Beachte dabei, dass $\neg B_x[a] \equiv \neg B$ ist, da x in B nicht frei vorkommt.

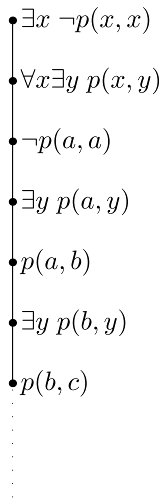
2. Sei $\Gamma = \{\forall x\ A \vee \forall x\ B\},\ F = \forall x\ (A \vee B)$.

   Gilt $\Gamma \models F$?



   γ-Formeln können nicht abgeharkt werden. Stattdessen können immer wieder neue Terme t eingeführt werden, für die $\gamma[t]$ erfüllt werden muss.

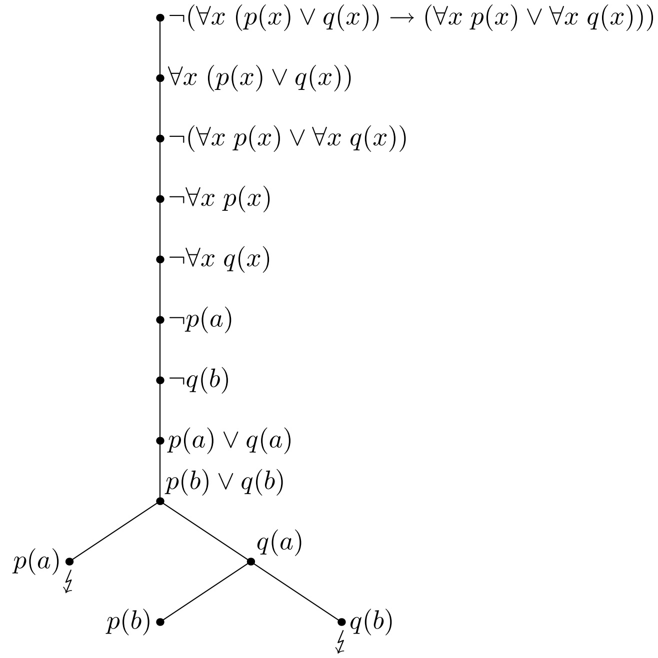
3. $\exists x\ \neg p(x,x) \not\models \neg \forall x \exists y\ p(x,y)$ Gesucht ist eine Interpretation, die Modell für $\Sigma = \{\exists x\ \neg p(x,x), \forall x \exists y\ p(x,y)\}$ ist.



   In diesem Fall terminiert das Tableau nicht. Aus dem Tableau lässt sich jedoch eine Interpretation I über dem unendlichen Definitionsbereich $\{a,b,c,\ldots\}$ ablesen mit $I(p)(a,a) = 0, I(p)(a,b) = 1, I(p)(b,c) = 1, I(p)(c,d) = 1, \ldots$, die Σ erfüllt.

4. Sei $A \equiv \forall x\ (p(x) \vee q(x)) \rightarrow (\forall x\ p(x)\ \vee\ \forall x\ q(x))$.

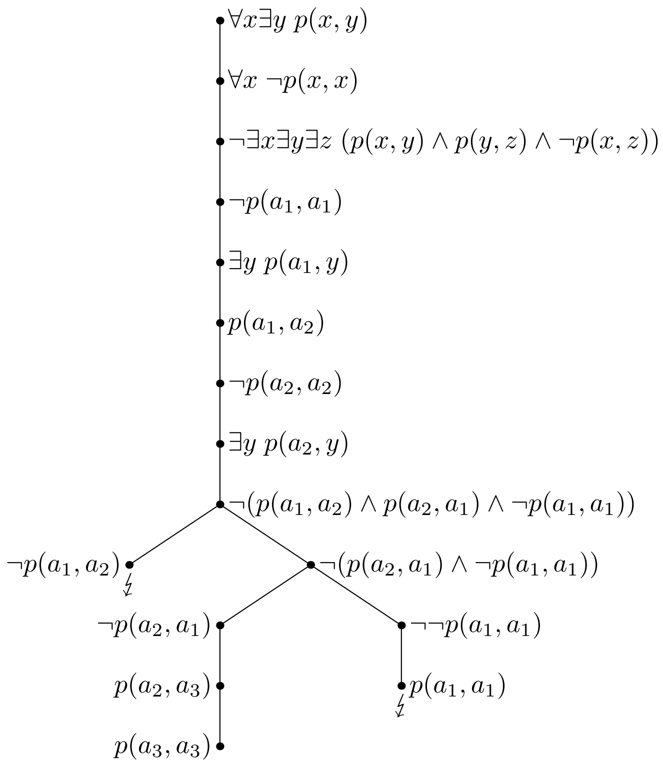
Frage: Ist A allgemeingültig ($\models A$)?



Offene vollständige Äste liefern Interpretationen, die $\neg A$ wahr machen. Dieses Tableau liefert die Interpretation $I = (\{a, b\}, \ldots\}$ mit $I(p)(a) = 0, I(p)(a) = 1, I(q)(a) = 1$ und $I(q)(b) = 0$. Also ist $I(A) = 0$ und es gilt <u>nicht</u> $\models A$.

5. Seien $A_1 \equiv \forall x \exists y\ p(x, y)$, $A_2 \equiv \forall x\ \neg p(x, x)$ und $B \equiv \exists x \exists y \exists z\ (p(x, y) \wedge p(y, z) \wedge \neg p(x, z)$.

Frage: Gilt $\{A_1, A_2\} \models B$?



Aus dem Tableau kann die Interpretation I über $\{a_1, a_2, a_3\}$ abgelesen werden, mit $I(p)(a_1, a_1) = 0, I(p)(a_1, a_2) = 1, I(p)(a_2, a_2) = 0, I(p)(a_2, a_1) = 0, I(p)(a_2, a_3) = 1, I(p)(a_3, a_3) = 0$. Somit ist I ein Gegenbeispiel zu $\{A_1, A_2\} \models B$, d.h. $\{A_1, A_2\} \not\models B$,

Die Allgemeingültigkeit für beliebige Formeln über Sprachen der Prädikatenlogik erster Stufe ist unentscheidbar. Wie sieht das aber für spezielle Formelklassen aus? In folgenden Formeln sei A eine quantorenfreie Formel ohne Funktionskonstanten. $(n, m \in \mathbb{N})$

1. $\forall x_1 \ldots \forall x_m \exists y_1 \ldots \exists y_n \; A$,
2. $\forall x_1 \ldots \forall x_m \exists y \forall z_1 \ldots \forall z_n \; A$,
3. $\forall x_1 \ldots \forall x_m \exists y_1 \exists y_2 \forall z_1 \ldots \forall z_n \; A$,
4. $\exists y_1 \ldots \exists y_n \forall x_1 \ldots \forall x_m \; A$, $(m, n \geq 1)$ und
5. $\forall x_1 \ldots \forall x_m \exists y_1 \exists y_2 \exists y_3 \forall z_1 \ldots \forall z_n \; A$, $(m, n \geq 0)$.

Die Allgemeingültigkeit für Formeln der Art 1.,2. und 3. ist entscheidbar, für Formeln der Art 4. und 5. ist die Allgemeingültigkeit nicht entscheidbar [Manna].

2.5 Die Resolventen-Methode

Die Resolventen-Methode (auch Resolutionsmethode genannt) ist ein partielles Entscheidungsverfahren für die Unerfüllbarkeit einer gegebenen Formel einer Sprache der Prädikatenlogik erster Stufe. Das bedeutet, dass das Verfahren bei Eingabe einer nicht erfüllbaren Formel B die Unerfüllbarkeit von B entdeckt und hält, bei Eingabe einer erfüllbaren Formel jedoch "unendlich" weiter laufen kann. Um zu testen, ob eine gegebene Formel A allgemeingültig ist, wird man also die Resolventen-Methode auf die Formel $\neg A$ anwenden. Ist $\neg A$ unerfüllbar (d.h. A ist allgemeingültig), dann wird das Resolventenverfahren das feststellen und halten. Die Resolventenmethode ist Grundlage für automatische Beweisverfahren und wird zur Interpretation von "Logik programmen" eingesetzt.

Die Resolventen-Methode wird angewandt auf Formeln in einer speziellen *Klauselform*. In diesem Abschnitt wird zunächst ein effektives Verfahren eingeführt, mit dem man eine beliebige Formel A der Prädikatenlogik erster Stufe in eine Formel A' in Klauselform transformieren kann, ohne dass dabei die Eigenschaft von A erfüllbar zu sein oder nicht verloren geht. Danach wird auf den Satz von Herbrand eingegangen, der einen grundlegenden Einfluss auf die Entscheidungsprozeduren hat, die auf der Resolventen-Methode beruhen. Schließlich werden zwei solche Entscheidungsprozeduren vorgestellt, nämlich die Methode von Davis und Putnam und die Grundresolventen-Methode.

2.39 Definition. 1. Ein *Literal* ist eine atomare Formel oder eine negierte atomare Formel.

2. Eine *Klausel* ist die Disjunktion einer oder mehrerer Literale.

3. Eine Formel der Prädikatenlogik erster Stufe ist in *Klauselform* (KLF), wenn sie die Form hat
$$\forall x_1 \forall x_2 \ldots \forall x_n \, [C_1 \wedge C_2 \wedge \ldots \wedge C_k]$$
Dabei bezeichnen die C_j Klauseln, und die $x_1, x_2, \ldots, x_n$ sind alle Variablen, die in den C_j vorkommen ($1 \leq j \leq k$). $\forall x_1 \forall x_2 \ldots \forall x_n$ heißt *Präfix* und $[C_1 \wedge C_2 \wedge \ldots \wedge C_k]$ die *Mantisse* der Formel.

Beachte: Eine Formel A ist genau dann in KLF, wenn A abgeschlossen und in PKNF ist und der Präfix kein $\exists$-Quantor enthält.

2.40 Lemma (Skolem)**.** Jede Formel D der Prädikatenlogik erster Stufe kann in eine Formel D' in Klauselform transformiert werden, so dass D genau dann erfüllbar ist, wenn D' erfüllbar ist.

Beweis. Die Transformation geschieht durch die Konstruktion einer endlichen Folge von Formeln $D_1, \ldots, D_n$, so dass $D_1 \equiv D$ und $D_n \equiv D'$ und für jedes j, mit $1 \leq j < n$, gilt: D_j ist genau dann erfüllbar, wenn D_{j+1} erfüllbar ist. Das Transformationsverfahren besteht aus 9 Schritten. Bis auf Schritt 7 ist die Invarianz im Bezug auf die Erfüllbarkeit für alle Schritte offensichtlich. Eine Rechtfertigung für Schritt 7 wird im Anschluss gegeben.

Verfahren: Gegeben eine Formel D.

Schritt 1: Bilde den existentiellen Abschluss von D.

Schritt 2: Eliminiere in D alle überflüssigen Quantoren.

Schritt 3: Benenne gebundene Variablen, die mehrfach in D quantifiziert werden um.

Schritt 4: Eliminiere durch logische Umformungen in D alle Operatoren, die von $\wedge, \vee, \neg$ verschieden sind.

Beispiel: Ersetze $(A \to B)$ durch $(\neg A \vee B)$ und $(\mathrm{IF}\, A\, \mathrm{THEN}\, B\, \mathrm{ELSE}\, C)$ durch $(\neg A \vee B) \wedge (A \vee C)$.

Schritt 5: Schiebe alle $\neg$ so weit nach innen, bis sie vor einem (positiven) Literal stehen.

Beispiel: Ersetze $(\neg \forall x\ A)$ durch $(\exists x\ \neg A)$, $\neg(A \vee B)$ durch $(\neg A \wedge \neg B)$ und $\neg\neg A$ durch A.

Schritt 6: Ziehe die Quantoren nach rechts.

Beispiel: Ersetze $\Delta x\ (A \star B)$ durch $(A \star \Delta x\ B)$, falls x nicht frei in A vorkommt, und durch $(\Delta x\ A \star B)$, falls x in B nicht frei vorkommt $(\Delta \in \{\exists, \forall\}, \star \in \{\vee, \wedge\})$.

Schritt 7 : Eliminiere existentielle Quantoren durch Einführung von *Skolem-Funktionen* (eventuell auch nullstelligen): Wähle die erste Teilformel (von links) der Form $\exists y\ B(y)$ und ersetze sie durch $B_y[f(x_1, \ldots, x_n)]$, $(n \geq 0)$, wobei

1. $x_1, \ldots, x_n$ alle unterschiedlichen freien Variablen in $\exists y\ B(y)$ sind, die links von $\exists y\ B(y)$ universell quantifiziert sind, und
2. f eine n-stellige Funktionskonstante ist, die noch nicht vorkommt.

Schritt 8 : Schiebe die $\forall$-Quantoren nach links.

Schritt 9 : Bringe die Matrix (durch Anwendung der Distributivgesetze) in Konjunktive Normalform.

Für die durch einmalige Anwendung der Ersetzung aus Schritt 7 aus einer Formel $D_j = \exists y\ B(y)$ entstehende Formel $D_{j+1} = B_y[f(x_1, \ldots, x_n)]$ gilt: D ist genau dann erfüllbar, wenn D_{j+1} erfüllbar ist. Denn: Ist D_j erfüllbar, dann muss es dafür ein Modell I geben, so dass es für jede mögliche Belegung für die Variablen $x_1, \ldots, x_n$ ein oder mehrere Werte für y gibt, mit denen D_j wahr wird. Betrachte nun eine Interpretation I', die zu I identisch ist, die aber zusätzlich eine n-stellige Funktionskonstante f so interpretiert, dass $f(x_1, \ldots, x_n)$ für jede Belegung von $x_1, \ldots, x_n$ gerade das Funktionsergebnis liefert, das y annehmen kann um D_j wahr zu machen. Dann ist I' ein Modell für D_{j+1} (d.h. D_{j+1} ist erfüllbar). Mit einem ähnlichen Argument (ausgehend von einem Modell für D_{j+1}) kann die Umkehrrichtung gezeigt werden. ■

2.41 Beispiel. Transformieren einer Formel in eine Formel in KLF. Betrachte die Formel

$$A \equiv \forall x\ \{p(x) \to \exists z\ \{\neg \forall y\ [q(x,y) \to p(f(x_1))] \wedge \forall y\ [q(x,y) \to p(x)]\}\}.$$

1. Existentieller Abschluss und Elimination von Quantoren:

$$\exists x_1 \forall x\ \{p(x) \to \{\neg \forall y\ [q(x,y) \to p(f(x_1))] \wedge \forall y\ [q(x,y) \to p(x)]\}\}$$

2. Umbenennung von y:

$$\exists x_1 \forall x \, \{p(x) \rightarrow \{\neg \forall y \, [q(x,y) \rightarrow p(f(x_1))] \wedge \forall z \, [q(x,z) \rightarrow p(x)]\}\}$$

3. Elimination von $\rightarrow$:

$$\exists x_1 \forall x \, \{\neg p(x) \vee \{\neg \forall y \, [\neg q(x,y) \vee p(f(x_1))] \wedge \forall z \, [\neg q(x,z) \vee p(x)]\}\}$$

4. $\neg$ nach innen:

$$\exists x_1 \forall x \, \{\neg p(x) \vee \{\exists y \, [q(x,y) \wedge \neg p(f(x_1))] \wedge \forall z \, [\neg q(x,z) \vee p(x)]\}\}$$

5. Schiebe Quantoren nach rechts:

$$\exists x_1 \forall x \, \{\neg p(x) \vee \{[\exists y \, q(x,y) \wedge \neg p(f(x_1))] \wedge [\forall z \, \neg q(x,z) \vee p(x)]\}\}$$

6. Eliminieren der Ex.-Quantoren $\exists x_1$ und $\exists y$:

$$\forall x \, \{\neg p(x) \vee \{[q(x,g(x)) \wedge \neg p(f(a))] \wedge [\forall z \, \neg q(x,z) \vee p(x)]\}\}$$

7. Quantoren nach links:

$$\forall x \forall z \, \{\neg p(x) \vee \{[q(x,g(x)) \wedge \neg p(f(a))] \wedge [\neg q(x,z) \vee p(x)]\}\}$$

8. Distributivgesetze anwenden:

$$\forall x \forall z \, \{[\neg p(x) \vee q(x,g(x))] \wedge [\neg p(x) \vee \neg p(f(a))] \wedge [\neg p(x) \vee \neg q(x,z) \vee p(x)]\}$$

Schließlich erhält man nach Vereinfachungen die Formel in Klauselform:

$$\forall x \, \{[\neg p(x) \vee q(x,g(x))] \wedge \neg p(f(a))\}.$$

Nach Definition ist eine Formel A unerfüllbar genau dann, wenn $I(A) = 0$ für jede Interpretation I, also insbesondere für jeden Definitionsbereich für I. Beim Beweis für die Unerfüllbarkeit von A wäre es jedoch von großem Nutzen, wenn man irgendeinen ("kleineren") festen Definitionsbereich D so bestimmen könnte, dass A in D genau dann unerfüllbar ist, wenn A überhaupt unerfüllbar ist. Tatsächlich gibt es einen solchen Definitionsbereich.

2.42 Definition. Sei $A \in$ Form. Das *Herbrand-Universum* (H_A) für eine Formel A ist die Menge aller Terme, die aus den Individuenkonstanten und den Funktionskonstanten, die in A vorkommen, gebildet werden können. Falls aus den Individuen- und den Funktionskonstanten, die in A vorkommen, keine Terme gebildet werden können, ist $H_A := \{a\}$ mit einer Individuenkonstanten a.

2.43 Beispiel. Sei $A \equiv \forall x \, \{[\neg p(x) \vee q(x,g(x))] \wedge \neg p(f(a))\}$, dann ist $H_A = \{a, f(a), g(a), g(g(a)), g(f(a)), f($

2.44 Definition. Eine *Herbrandinterpretation* für eine (geschlossene) Formel A ist eine Interpretation $I = (H_A, \ldots)$, die

1. jeder Individuenkonstante a in A den Term a aus H_A zuordnet,
2. jeder n-stelligen ($n \geq 1$) Funktionskonstanten f, die in A vorkommt, eine Funktion $I(f) : {H_A}^n \to H_A$ zuordnet, die die Terme $t_1, t_2, \ldots, t_n \in H_A$ auf den Term (als Zeichenkette) $f(t_1, \ldots, t_n) \in H_A$ abbildet.

Herbrandinterpretationen unterscheiden sich also nur auf den Prädikatskonstanten.

2.45 Satz. Eine Formel A in KLF ist genau dann unerfüllbar, wenn A in allen Herbrandinterpretationen unerfüllbar ist [Manna, 1974].

Sei A eine Formel in KLF der Form $A \equiv \forall x_1 \ldots \forall x_n \ [C_1 \wedge \ldots \wedge C_k]$. Die *Grundinstanz* einer Klausel C_j $(1 \leq j \leq k)$ von A ist die Klausel, die man erhält, wenn man alle Individuenvariablen in C_j durch Terme aus H_A ersetzt. Solche Klauseln, die keine Individuenvariablen enthalten heißen *Grundklauseln*.

2.46 Satz (Herbrand)**.** Eine Formel A in Klauselform ist genau dann unerfüllbar, wenn es eine endliche Konjunktion von Grundinstanzen ihrer Klauseln gibt, die unerfüllbar ist.

Dieser Satz legt ein Verfahren nahe die Unerfüllbarkeit einer Formel in Klauselform zu testen. Das Verfahren ist unter dem Namen "Herbrand-Prozedur" bekannt.

2.47 Herbrand-Prozedur. Sei A eine Formel in KLF mit n Klauseln. Erzeuge der Reihe nach alle Grundklauseln G_j aus den Klauseln C_j $(1 \leq j \leq n)$ von A und prüfe, ob ihre Konjunktion unerfüllbar ist.

Bei Durchführung der Herbrand-Prozedur tauchen zwei Probleme auf. Zum einen stellt sich die Frage, wie man die Grundklauseln systematisch erzeugen kann, und zum anderen, wie man testen kann, ob eine Konjunktion von Grundklauseln unerfüllbar ist.

Es gibt zwei bekannte Verfahren, die Unerfüllbarkeit einer Konjunktion von Klauseln zu testen: die Methode von Davis und Putnam und die Grundresolventen-Methode.

Hier soll die *Grundresolventen-Methode* vorgestellt werden. Sie beruht auf der (Grund-) Resolventenregel.

2.48 Definition. Die *(Grund-) Resolventenregel* kann wie folgt beschrieben werden:

Für je zwei Grundklauseln G und H, von denen eine ein Literal l und die andere dessen Komplement $\neg l$ enthält, konstruiere eine neue Klausel K als Disjunktion von G' und H'. Dabei bezeichne G' bzw. H' die Formeln, die man aus G bzw. H erhält, wenn man alle Vorkommen von l bzw. $\neg l$ in G und H streicht. Doppelte Vorkommen von Literalen in K werden gestrichen. K heißt *Resolvent(e)* von G und H. Im Spezialfall, dass die betrachteten Klauseln jeweils nur aus dem Literal l bzw. $\neg l$ bestehen, ist die entstehende Klausel K die *leere Klausel*, notiert "$\Box$".

Das folgende Lemma beschreibt das Hauptresultat des Verfahrens. Es beruht auf der Beobachtung, dass $S = G_1 \wedge G_2 \wedge \ldots \wedge G_n$ genau dann unerfüllbar ist, wenn $S \wedge G_{n+1}$ unerfüllbar ist, wobei G_{n+1} eine Resolvente aus zwei Klauseln von S ist.

2.49 Lemma. Eine endliche Konjunktion von Grundklauseln $S = G_1 \wedge G_2 \wedge \ldots \wedge G_n$ ist genau dann unerfüllbar, wenn durch wiederholte Anwendung der Grundresolventenregel die leere Klausel $\square$ aus S hergeleitet werden kann.

Verfahren:

Starte für eine Klauselform $S = G_1 \wedge G_2 \wedge \ldots \wedge G_n$ mit

$$\begin{array}{c} G_1 \\ G_2 \\ \vdots \\ G_n \end{array}$$

und bilde alle möglichen Resolventen. Wird $\square$ hergeleitet, dann ist S unerfüllbar, sonst ist S erfüllbar. Das Verfahren stoppt, da mit endlich vielen Literalen nur endlich viele Klauseln gebildet werden können.

2.50 Beispiele.

1.

$$\begin{aligned} S \;\equiv\; & (p_1(a) \vee p_2(b)) \wedge \\ & (p_1(a) \vee \neg p_3(a)) \wedge \\ & (\neg p_1(a) \vee p_3(a)) \wedge \\ & (\neg p_1(a) \vee \neg p_2(b)) \wedge \\ & (p_3(a) \vee \neg p_2(b)) \wedge \\ & (\neg p_3(a) \vee p_2(b)) \end{aligned}$$

$$\begin{array}{rll} 1. & p_1(a) \vee p_2(b) & \\ 2. & p_1(a) \vee \neg p_3(a) & \\ 3. & \neg p_1(a) \vee p_3(a) & \\ 4. & \neg p_1(a) \vee \neg p_2(b) & \\ 5. & p_3(a) \vee \neg p_2(b) & \\ 6. & \neg p_3(a) \vee p_2(b) & \\ 7. & p_2(b) \vee p_3(a) & R(1,3) \\ 8. & p_2(b) & R(6,7) \\ 9. & \neg p_3(a) \vee \neg p_2(b) & R(2,4) \\ 10. & \neg p_2(b) & R(5,9) \\ 11. & \square & R(8,10) \end{array}$$

2. $S \equiv (p_1(a) \vee p_2(b)) \wedge \neg p_2(b) \wedge (\neg p_1(a) \vee p_2(b) \vee p_3(a))$

1.	$p_1(a) \vee p_2(b)$	
2.	$\neg p_2(b)$	
3.	$\neg p_1(a) \vee p_2(b) \vee p_3(a)$	
4.	$p_1(a)$	$R(1,2)$
5.	$p_2(b) \vee p_3(a)$	$R(1,3)$
6.	$\neg p_1(a) \vee p_3(a)$	$R(2,3)$
7.	$p_3(a)$	$R(2,5)$

Es können keine weiteren Klauseln erzeugt werden. S ist also erfüllbar in Interpretationen I, mit $I(p_1)(a) = 1, I(p_3)(a) = 0$ und $I(p_2)(b) = 0$.

Zwar liegt mit der Grundresolventenregel ein Verfahren vor, das Unerfüllbarkeitsproblem für eine endliche Konjunktion von Grundklauseln zu entscheiden. Das zweite wichtige Problem bei der Implementierung des Herbrandverfahrens ist damit jedoch noch nicht gelöst: Die systematische Erzeugung der Grundinstanzen zu einer Klausel. Für die meisten prädikatenlogischen Formeln müssen sehr viele Grundinstanzen erzeugt werden, bevor eine nichterfüllbare Konjunktion darunter ist. Um das Erzeugen von Grundinstanzen zu umgehen, führte Robinson die Resolutions-Regel ein. Sie ist eine Generalisierung der Grundresolventenregel: Man kann sie direkt auf eine Konjunktion von Klauseln (also nicht unbedingt auf Grundklauseln) anwenden, um deren Unerfüllbarkeit zu zeigen.

2.51 Beispiel. Frage: Ist

$$A \equiv [\forall x \exists y\ p(x,y) \wedge \forall y \exists z\ q(y,z)] \rightarrow \forall x \exists y \exists z\ (p(x,y) \wedge q(y,z))$$

allgemeingültig? Zuerst wird $\neg A$ in eine Formel B in Klauselform transformiert:

$$B \equiv \forall x \forall y \forall y_1 \forall z\ [p(x, f(x)) \wedge q(y, g(y)) \wedge (\neg p(a, y_1) \vee \neg q(y_1, z))].$$

Das Herbranduniversum für B lautet $H_B = \{a, f(a), g(a), f(f(a)), f(g(a)), \ldots\}$ Um mit der Grundresolventenmethode zu zeigen, dass B unerfüllbar ist, generiert man nacheinander die Grundinstanzen für die Klauseln aus B und testet sukzessive, ob deren Konjunktion unerfüllbar ist: $[p(a, f(a)) \wedge q(a, g(a)) \wedge [\neg p(a,a) \vee \neg q(a,a)] \ldots \wedge [p(a, f(a)) \wedge q(f(a), g(f(a))) \wedge [\neg p(a, f(a)) \vee \neg q(f(a), g(f(a)))]]$.

Erst aus dieser Konjunktion kann die leere Klausel hergeleitet werden:

1.	$p(a, f(a))$	
2.	$q(f(a), g(f(a)))$	
3.	$\neg p(a, f(a)) \vee \neg q(f(a), g(f(a)))$	
4.	$\neg q(f(a), g(f(a)))$	$R(1,3)$
5.	$\Box$	$R(2,4)$

Die Idee der allgemeinen Resolutionsmethode ist, die Klauseln (einer Formel wie B) direkt zur Resolution heranzuziehen, ohne den Umweg über ihre Grundinstanzen zu machen. Am Beispiel der Formel B soll hier zunächst illustriert werden, wie das Verfahren arbeitet:

Zunächst die Klauseln von B :

$$\begin{array}{ll} 1. & p(x, f(x)) \\ 2. & q(y, g(y)) \\ 3. & \neg p(a, y_1) \vee \neg q(y_1, z) \end{array}$$

Betrachte die Klauseln 1. und 3. Zwar ist das Literal aus 1. zu keinem Literal der dritten Klausel invers, ersetzt man jedoch die Variable x durch die Konstante a und die Variable y_1 durch den (konstanten) Term $f(a)$, dann erhält man die Klauseln

$$\begin{array}{ll} 1'. & p(a, f(a)) \\ 3'. & \neg p(a, f(a)) \vee \neg q(f(a), z) \end{array}$$

Eine solche Ersetzung heißt *Substitution* und wird notiert als

$$\{\langle x, a\rangle, \langle y_1, f(a)\rangle\}.$$

Die Substitution leistet im obigen Fall etwas besonderes, durch sie werden $p(x, f(x))$ und $p(a, y_1)$ zur selben Formel $p(a, f(a))$. Die Substitution $\{\langle x, a\rangle, \langle y_1, f(a)\rangle\}$ wird deshalb auch *Unifikator* genannt. Man könnte natürlich noch zusätzlich die Variable z durch a oder $f(a)$ usw. substituieren. Die dann entstehende Substitution wäre ebenfalls ein Unifikator. Sie würde aber Informationen enthalten, die zum Unifizieren ("Gleichmachen") von $p(x, f(x))$ und $p(a, y_1)$ nicht notwendig sind. In diesem Sinne ist der Unifikator $\{\langle x, a\rangle, \langle y_1, f(a)\rangle\}$ allgemeiner. Ja, es gibt in diesem Falle keinen Unifikator, der allgemeiner ist. $\{\langle x, a\rangle, \langle y_1, f(a)\rangle\}$ wird deshalb auch *Allgemeinster Unifikator* oder *Most General Unifier* (MGU) genannt.

Resolviert man nun die Klauseln 1′. und 3′. miteinander, dann erhält man

$$4. \quad \neg q(f(a), z)$$

als Resolvente. Durch Anwenden der Substitution $\{\langle y, f(a)\rangle, \langle z, g(f(a))\rangle\}$ auf die Klauseln 2. und 4. werden diese unifiziert und man erhält als Resolvente aus beiden:

$$5. \quad \square$$

2.52 Definition. Eine *Substitution* σ ist eine endliche Menge der Form

$$\{\langle v_1, t_1\rangle, \ldots, \langle v_n, t_n\rangle\},$$

wobei die v_j $(1 \leq j \leq n)$ voneinander verschiedene Variablen sind und die t_j von v_j verschiedene Terme sind. Jedes Element $\langle v_j, t_j\rangle$ heißt *Bindung* für v_j.

2.53 Definition. Sei $\sigma = \{\langle v_1, t_1\rangle, \ldots, \langle v_n, t_n\rangle\}$ eine Substitution und E ein Literal. Dann ist $E\sigma$, die *Instanz* von E, das Literal, das aus E durch gleichzeitiges Ersetzen jedes Vorkommens der Variablen v_j in E durch die Terme t_j $(j = 1, \ldots, n)$ entsteht.

2.54 Definition. Seien $\sigma = \{\langle u_1, s_1\rangle, \ldots, \langle u_n, s_n\rangle\}$ und $\tau = \{\langle v_1, t_1\rangle, \ldots, \langle v_m, t_m\rangle\}$ Substitutionen. Die *Komposition* $\sigma\tau$ von σ und τ ist die Substitution, die man aus der Menge $\{\langle u_1, s_1\tau\rangle, \ldots, \langle u_n, s_n\tau\rangle, \langle v_1, t_1\rangle, \ldots, \langle v_m, t_m\rangle\}$ erhält, wenn man jede Bindung $\langle u_j, s_j\tau\rangle$ mit $u_j = s_j\tau$ $(1 \leq j \leq n)$ und jede Bindung $\langle v_k, t_k\rangle$ mit $v_k \in \{u_1, \ldots, u_n\}$ $(1 \leq k \leq m)$ daraus entfernt.

2.55 Definition. Die durch die leere Menge gegebene Substitution e heißt *Identitätssubstitution*, d.h. $xe = x$ für jede Variable x.

2.56 Definition. Sei $S = A_1 \vee \ldots \vee A_n$ eine Disjunktion atomarer Formeln A_j $(1 \leq j \leq n)$. Eine Substitution σ heißt *Unifikator* für S, falls $A_1\sigma = A_2\sigma = \ldots = A_n\sigma$ gilt. Gibt es für S einen solchen Unifikator, dann heißt S *unifizierbar*. Ein Unifikator heißt *allgemeinster Unifikator* oder *most general unifier* (MGU) für S, wenn es für jeden Unifikator τ von S eine Substitution υ gibt, so dass $\tau = \sigma\upsilon$.

2.57 Definition. Sei S eine endliche Disjunktion von Literalen. Das *"disagreement set"* von S wird wie folgt definiert: Suche die am weitesten links stehende Symbolposition in den Literalen von S, an denen sich die Literale unterscheiden und bestimme für jedes Literal in S den Teilterm, der an der gefundenen Position beginnt. Die Menge all dieser Teilausdrücke ist das "disagreement set".

2.58 Beispiel. Das "disagreement set" für die Disjunktion

$$p(x, g(f(y, z), x), y) \vee p(x, g(a, b), b) \vee p(x, g(g(h(x), a)y, h(x))$$

ist die Menge $\{f(y, z), a, g(h(x), a)\}$.

2.59 Satz (Unifikationstheorem)**.** Gegeben folgender Unifikationsalgorithmus:

Unifikations - Algorithmus

Eingabe: $S \equiv A_1 \vee \ldots \vee A_n$, eine Disjunktion von atomaren Formeln $A_1, \ldots, A_n$.

Ausgabe: MGU σ_k für S, falls dieser existiert, oder eine Meldung, dass S nicht unifizierbar ist sonst.

Verfahren:

1. Setze $k = 0$ und $\sigma_k = e$.
2. Falls σ_k Unifikator für S ist, dann stoppen, σ_k ist gesuchter MGU.

   Ansonsten finde das "disagreement set" D_k von $S\sigma_k \equiv A_1\sigma_k \vee \ldots \vee A_n\sigma_k$.
3. Gibt es v und t in D_k derart, dass v eine Variable ist, die nicht in t vorkommt, dann setze $\sigma_{k+1} := \sigma_k \cup \{\langle v, t\rangle\}$, erhöhe k um 1 und fahre mit Schritt 2 fort.

   Ansonsten melde, dass S nicht unifizierbar ist und stoppe.

Falls S unifizierbar ist, hält der Algorithmus und liefert einen MGU für S als Ergebnis. Ist S nicht unifizierbar, hält der Algorithmus und meldet, dass S nicht unifizierbar ist.

Die nachfolgend vorgestellte Resolutions-Regel basiert auf der Resolventenregel.

2.60 Definition (Resolutions-Regel)**.** Seien C_1 und C_2 Klauseln ohne gemeinsame Variablen (Diese Bedingung ist keine Einschränkung, denn sie kann durch die Transformation einer Klausel C_j in eine äquivalente Klausel C'_j $(j \in \{1,2\})$, in der die Variablen, durch die die Bedingung verletzt wurde umbenannt wurden, stets erfüllt werden.) und ist $p(t_1^{(1)}, \ldots, t_k^{(1)}) \vee \ldots \vee p(t_1^{(n)}, \ldots, t_k^{(n)})$ eine Teildisjunktion von C_1 und $\neg p(s_1^{(1)}, \ldots, s_k^{(1)}) \vee \ldots \vee \neg p(s_1^{(m)}, \ldots, s_k^{(m)})$ eine Teildisjunktion von C_2, so dass es einen MGU β gibt für die Disjunktion $p(t_1^{(1)}, \ldots, t_k^{(1)}) \vee \ldots \vee p(t_1^{(n)}, \ldots, t_k^{(n)}) \vee \neg p(s_1^{(1)}, \ldots, s_k^{(1)}) \vee \ldots \vee \neg p(s_1^{(m)}, \ldots, s_k^{(m)})$ mit dem Faktor $p(r_1, \ldots, r_k)$ der Disjunktion. Dann konstruiere als *Resolvente* von C_1 und C_2 die Klausel C_3 aus der Disjunktion von $C_1\beta$ und $C_2\beta$ nach Streichen von $p(r_1, \ldots, r_k)$ aus $C_1\beta$ und $\neg p(r_1, \ldots, r_k)$ aus $C_2\beta$.

2.61 Satz (Robinson)**.** Eine Formel S in Klauselform ist genau dann unerfüllbar, wenn die leere Klausel aus S mit der Resolutions-Regel hergeleitet werden kann.

2.62 Beispiel. Um zu beweisen, dass die Formel

$$A \equiv \neg\exists y \forall z \, [p(z,y) \leftrightarrow \neg\exists x \, [p(z,x) \wedge p(x,z)]]$$

allgemeingültig ist, wird mit der Resolutionsmethode gezeigt, dass

$$\exists y \forall z \, [p(z,y) \leftrightarrow \neg\exists x \, [p(z,x) \wedge p(x,z)]]$$

unerfüllbar ist. Die $\neg A$ entsprechende Formel in Klauselform lautet

$$\forall z \forall x \, [[\neg p(z,a) \vee \neg p(z,x) \vee \neg p(x,z)] \wedge [p(z,f(z)) \vee p(z,a)] \wedge [p(f(z),z) \vee p(z,a)]]$$

und enthält die Klauseln (nach Umbenennung der Variablen)

1. $\neg p(z_1,a) \vee \neg p(z_1,x_1) \vee \neg p(x_1,z_1)$
2. $p(z_2,f(z_2)) \vee p(z_2,a)$
3. $p(f(z_3),z_3) \vee p(z_3,a)$

Um die Klauseln 1. und 2. zu resolvieren, werden die Teildisjunktionen $\neg p(z_1,a) \vee \neg p(z_1,x_1) \vee \neg p(x_1,z_1)$ aus Klausel 1. und $p(z_2,a)$ aus Klausel 2. betrachtet. Der MGU von $\neg p(z_1,a) \vee \neg p(z_1,x_1) \vee \neg p(x_1,z_1) \vee p(z_2,a)$ lautet

$$\{\langle x_1,a\rangle, \langle z_2,a\rangle, \langle z_1,a\rangle\}.$$

Deshalb erhält man als Resolvente der 1. und 2. Klausel

4. $p(a,f(a))$

Ähnlich erhält man aus den Klauseln 1. und 3. mit den Teildisjunktionen $\neg p(z_1,a) \vee \neg p(z_1,x_1) \vee \neg p(x_1,z_1)$ und $p(z_3,a)$ den MGU $\{\langle x_1,a\rangle, \langle z_1,a\rangle, \langle z_3,a\rangle\}$ und die Resolvente

5. $p(f(a),a)$

Als MGU von $\neg p(x_1, z_1) \vee p(a, f(a))$ erhält man $\{\langle x_1, a\rangle, \langle z_1, f(a)\rangle\}$ und deshalb als Resolvente von Klausel 1. und Klausel 4:

$$6. \quad \neg p(f(a), a)$$

Schließlich erhält man die leere Klausel als Resolvente der Klauseln 5. und 6.

$$7. \quad \square$$

Also ist die Formel $\neg A$ unerfüllbar und A allgemeingültig.

Literatur

Dies war nur ein kleiner Einblick in Methoden der formalen Logik. Mittlerweile sind die Entwicklungen auf diesem Gebiet fortgeschritten und eine nähere Betrachtung ist nicht Ziel dieser Einführung. Der interessierte Leser kann Hinweise auf weiterführende Literatur in den Referenzen finden.

Heinz-Dieter Ebbinghaus, Jörg Flum, and Wolfgang Thomas. *Einfhrung in die mathematische Logik.* Spektrum, Heidelberg, 1998.

H. B. Enderton. *A Mathematical Introduction to Logic.* Academic Press, 1972.

Jean H. Gallier. *Logic for Computer Science.* Harper & Row, New York, 1986.

B. Heinemann and K. Weihrauch. *Logik für Informatiker.* Teubner, Stuttgart, 1991.

H.-J. Kreowski. *Logische Grundlagen der Informatik.* Oldenbourg, München, 1991.

Z. Manna. *Mathematical Theory of Computation.* McGraw-Hill, 1974.

Nimal Nissanke. *Introductory Logic and Sets for Computer Scientists.* Addison-Weslay, Harlow, 1999.

Uwe Schöning. *Logik für Informatiker.* Number 56 in Reihe Informatik. Wissenschaftsverlag, Mannheim, 1989.

R. M. Smullyan. *First-Order Logic.* Springer, 1968.

V. Sperschneider and G. Antoniou. *Logic: a foundation for computer science.* Addison-Wesley, Wokingham, 1991.

A. Yasuhara. *Recursive Function Theory Logic.* Academic Press, 1971.

Index