Logik

Prof. Dr. Madlener

TU Kaiserslautern

SS 2005

Logik

Studiengang „Informatik“, „Technoinformatik“und „ WiWi/Inf“
SS'05
Prof. Dr. Madlener TU - Kaiserslautern

Vorlesung:
Mi 11.45-13.15 52/207 **Achtung:** Mi 4.5 46/215

- Informationen
  http://www-madlener.informatik.uni-kl.de/teaching/ss2005/logik/logik.html
- Grundlage der Vorlesung: Skript
  Einführung in die Logik und Korrektheit von Programmen.

- Bewertungsverfahren:
  Übungen:
  Aufsichtsarbeit: max. 30 Punkte,
  Abschlussklausur: max. 70 Punkte.
- Übungen: Hörsaalübung Montag 10:00–11:30 1/106
  Sprechzeiten siehe Homepage

Grundlagen der Aussagenlogik

- Syntax
- Semantik
- Deduktiver Aufbau der Aussagenlogik
- Natürliche Kalküle

Algorithmischer Aufbau der Aussagenlogik

- Semantische Tableaux
- Normalformen
- Davis-Putman-Algorithmen
- Resolutions-Verfahren

Grundlagen der Prädikatenlogik

- Beziehungen zwischen Eigenschaften von Elementen
- Semantik der P-Logik 2-Stufe – Interpretationen, Belegungen, Bewert
- Transformationen von Termen und Formeln

Entscheidbarkeit in der Prädikatenlogik

- Unentscheidbarkeit der Allgemeingültigkeit
- Hauptsätze der Prädikatenlogik erster Stufe
- Theorien erster Stufe

Algorithmen der Prädikatenlogik

- Aufzählungsverfahren für PL-1
- Resolventenmethode – (Allg. Resolutionsverfahren)
- Logisches Programmieren und Prolog

Beziehungen zwischen Eigenschaften von Elementen Funktionen, Prädikate

Funktionen, Prädikate

- Beschreibung von Strukturen (Datentypen, Algebren,...)
- Beschreibung von Integritätsbedingungen (Relationen,...)
- Lösbarkeit von Anfragen (Gleichungen,...)

Beispiel 3.1

Bereich $\mathbb{Z}$, Operationen, Prädikate,...

- Zahlen: Konstanten: $\cdots, -2, -1, 0, 1, 2, \cdots$
- Individuenvariablen (Abzählb.): $x, y, z, \ldots$
- Funktionszeichen (Operatoren): $+, \cdot, -$ (Stelligkeit 2)
- Prädikatszeichen (Relationen): $<, >$ (Stelligkeit 2)
- Logische Zeichen: $=, \land, \lor, \neg, \rightarrow, \ldots$
- Quantoren: $\forall$ (für alle), $\exists$ (es gibt)
- Trennzeichen: $(,), \cdots$

Hilberts-10-Problem: Gibt es ein Verfahren, um zu entscheiden, ob eine diophantische Gleichung eine Lösung hat?

- Hat ein Polynom in mehreren Veränderlichen mit Koeffizienten in $\mathbb{Z}$ eine Lösung in $\mathbb{Z}$?
  $\exists x, y, z.3x^2 - 2xy^3 + 5z^3 + 6 = 0$
- Entscheide effektiv, ob eine Aussage in $\mathbb{Z}$ gilt.
- Beschreibung von Eigenschaften:
  $U(x) \equiv \exists z.x = 2z + 1, \quad G(x) \equiv \exists z.x = 2z$
  $\forall x(U(x) \vee G(x))$

Aufbau von Formeln:
Aus Konstanten, Funktionen, Individuenvariablen können **Terme** definiert werden. Terme dienen als Bezeichner für Elemente.

- Jede Variable ist Term, jede ganze Zahl ist Term
  t_1, t_2 Terme, so auch $(t_1 + t_2), (t_1 - t_2), (t_1 \cdot t_2)$
- **Atomare Formeln**:
- t_1, t_2 Terme: So sind $t_1 = t_2, t_1 < t_2, t_1 > t_2$ Atomare Formeln.
- A, B Formeln: So sind $(A \wedge B), (A \vee B), (A \rightarrow B), (\neg A)$, und für
- x Variable, A Formel: $\forall x.A$, $\exists x.A$ Formeln.

Interpretationen

- **Bedeutung von Termen und Formeln**. Interpretation:
  Hier in $\mathbb{Z}$: Terme klar $+, -, \cdot$ durch die Operatoren auf $\mathbb{Z}$.
  Bedeutung von: $(x+5)=y$, $\forall x(x+5)=0$, $\exists x(x+5)=0$, $\forall x \exists y(x+5)=y$
- **Interpretation:** Bereich, Funktionen, Prädikate.
  Belegung der Individuen-Variablen.

▶ $x \to 3 \qquad y \to 8$, dann $(x+5)=y$ wahr.

- **Allgemeinere Formeln**: Quantifizierung über Funktionen, Prädikaten.
- $A \equiv \exists F((F(a)=b) \wedge \forall x[p(x) \to F(x)=g(x,F(f(x)))]$
  wobei F Funktionsvariable, a, b, p Konstanten und f, g Funktionskonstanten sind.

Interpretationen

1. $D = \mathbb{N} \quad a = 0, b = 1 \quad f(x) = x \dot{-} 1$
   $g(x, y) = x \cdot y \quad p(x) \equiv x > 0$
   - Gibt es eine Funktion $F: \quad \mathbb{N} \to \mathbb{N}$:
     $F(0) = 1 \qquad F(x) = x \cdot F(x \dot{-} 1) \quad (x > 0)$
2. $D = \mathbb{N} \quad a = 0, b = 1 \quad f(x) = x$
   $g(x, y) = y + 1 \quad p(x) \equiv x > 0$
   - Gibt es eine Funktion $F: \quad \mathbb{N} \to \mathbb{N}$:
     $F(0) = 1 \qquad F(x) = F(x) + 1 \quad (x > 0)$

Allgemeine Sprache der Prädikatenlogik zweiter Stufe

Definition 3.2 (Syntax)

(a) Alphabet:

1. **Wahrheitswerte:** W, F (Log-Konstanten)
2. **Logische Symbole:**
   - 2.1 **Junktoren:** $\neg, \rightarrow, \wedge, \vee, \leftrightarrow, \cdots$, **If _ Then _ Else _**
   - 2.2 **Operatoren:** $=$, **if _ then _ else**
   - 2.3 **Quantoren:** $\forall$ (Allquantor), $\exists$ (Existenzquantor)
3. **Variablensymbole:**
   - 3.1 n-stellige **Funktionsvariablen:** $F_j^n (j \geq 1, n \geq 0)$
     $n = 0$ **Individuenvariablen:** Bezeichnung x_j
   - 3.2 n-stellige **Prädikatenvariablen:** $P_j^n (j \geq 1, n \geq 0)$
     $n = 0$ **aussagenlogische Variablen**

4. **Konstantensymbole:**
   4.1 n-stellige **Funktionskonstanten:** $f_j^n (j \geq 1, n \geq 0)$
   f_j^0 **Individuenkonstanten:** Bezeichnung a_j
   4.2 n-stellige **Prädikatskonstanten:** $p_j^n (j \geq 1, n \geq 0)$
   p_j^0 **A-log-Konstanten** Bezeichnung p_j
5. **Hilfssymbole** (Klammern).

! Alle Zeichen verschieden, kein Buchstabe Teilwort eines anderen.

! Entscheidbare Teilalphabete. Stelligkeiten eindeutig festgelegt.

! Spezielle Sprachen werden durch Festlegung der Konstanten (oft nur endlich viele) definiert.

Allgemeine Sprache der Prädikatenlogik zweiter Stufe (Forts.)

(b) Ausdrücke: Terme - Formeln

1. Die Menge **Term** der Terme (Bezeichner):

   i. Jede Individuenvariable x_j und Individuenkonstante a_j $(j \geq 1)$ ist ein (atomarer) Term
   ii. Sind $t_1, \dots, t_n (n \geq 1)$ Terme, so auch $f_j^n(t_1, \dots, t_n)$ und $F_j^n(t_1, \dots, t_n)$ $(j \geq 1)$
   iii. Ist A Formel, t_1, t_2 Terme, so auch $(\textbf{if } A \textbf{ then } t_1 \textbf{ else } t_2)$
   iv. **Term** ist kleinste Menge mit i die Abg. bzg. ii. und iii ist.

2. Die Menge der Formeln **Form**:
   - Atomare Formeln: **AForm**
     i. $W, F \in \mathbf{AForm}$
     ii. $p_j^0, P_j^0 \in \mathbf{AForm}\ \ (j \geq 1)$
     iii. Sind $t_1, \ldots, t_n$ $(n \geq 1)$ Terme, so
     $p_j^n(t_1, \ldots, t_n), P_j^n(t_1, \ldots, t_n) \in \mathbf{AForm}\ \ (j \geq 1)$
     iv. Sind t_1, t_2 Terme, dann ist $(t_1 = t_2) \in \mathbf{AForm}$
   - Formeln: **Form**
     i. $\mathbf{AForm} \subseteq \mathbf{Form}$
     ii. $A, B, C \in \mathbf{Form}$, so auch
     $(\neg A), (A \rightarrow B), (A \vee B), (A \wedge B), (A \leftrightarrow B)$
     $(\mathbf{If}\ A\ \mathbf{Then}\ B\ \mathbf{Else}\ C) \in \mathbf{Form}$
     iii. Ist v Variable, A Formel
     $((\forall v)A), ((\exists v)A) \in \mathbf{Form}$ (gelegentlich mit Einschränkung)

(c) freie (gebundene) Variable. **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.
- Ein **Vorkommen** einer Variablen v in einer Formel heißt gebunden, falls es im Geltungsbereich eines Quantors Qv vorkommt. 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.**

(d) Teilterme und Teilformeln werden wie üblich definiert.
Beachte: Jeder Term, Formel wird eindeutig aus den Teiltermen bzw. Teilformeln aufgebaut.

Bemerkung 3.3

a) **Term**, **Form** *sind rekursiv entscheidbar, zusammengesetzte Terme und Formeln lassen sich eindeutig zerlegen. Freie und gebundene Vorkommen lassen sich effektiv bestimmen.*

b) **Vereinbarungen**: *Stelligkeit aus Kontext*
a, b, c *Individuenkonstanten,* $f, g, h, \dots$ *Funktionskonstanten* $p, q, r \dots$ *Prädikatskonstanten,* $x, y, z \dots$ *Individuenvariablen* $F, G, H, \dots$ *Funktionsvariablen* $P, Q, R, \dots$ *Prädikatsvariablen,* $A, B, C, \dots$ *Formeln,* t, s *Terme.*
Die Mengen $Var(t)$, $Var(A)$ *seien die Variablen, die in* t *bzw. in* A *vorkommen.*

Beispiel

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

Term Term Term Term Term

at-F at.F at.F

Formel

Formel

Formel

i) $A \equiv$ Formel

$Var(A) = \{F, x\}$ F kommt $3\times$ vor, gebunden

x kommt $4\times$ vor, gebunden

A abgeschlossene Formel

ii) $$A \equiv \forall P\{ \underset{\text{geb.}}{\underset{\uparrow}{P}}(a) \wedge \forall x[[\underset{\text{geb.}}{\underset{\uparrow}{x}} \neq a \wedge \underset{\text{geb.}}{\underset{\uparrow}{P}}(f(\underset{\text{geb.}}{\underset{\uparrow}{x}}))] \rightarrow \underset{\text{geb.}}{\underset{\uparrow}{P}}(\underset{\text{geb.}}{\underset{\uparrow}{x}})]] \rightarrow \underset{\text{geb.}}{\underset{\uparrow}{P}}(\underset{\text{frei}}{\underset{\uparrow}{x}})\}$$

A ist nicht abgeschlossen, x hat freies Vorkommen d.h. $FVar(A) = \{x\}$.

Eingeschränkte Teilsprachen

Konstanten, Variablen einschränken

1. Allgemeine Sprache der Prädikatenlogik erster stufe: PL1
   - Nur Individuenvariablen x_i
     (keine Funktion- und Prädikatsvariablen)

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

2. - Sprache der Gleichheitslogik
     Nur Individuenvariablen x_j $(j \geq 1)$, Funktionskonstanten.
     Keine Prädikatskonstanten und Variablen, keine Funktionsvariablen.
     Oft noch eingeschränkt, nur Individuenkonstanten
   - reine Gleichheitslogik
     Term : $x_i, a_i,$ **if** A **then** t_1 **else** t_2
     AForm : $W, F,\ t_1 = t_2$

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

3. Sprache der quantifizierten Aussagenlogik
   Nur aussagenlogische Konstanten $p_j^0 (j \geq 1)$ und aussagenlogische Variablen $P_j^0 (j \geq 1)$ sind zugelassen.
   Keine Terme. Atomare Formeln: aussagenlogische Konstanten und Variablen, W, F, p, P_i.

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

4. Sprache der Aussagenlogik
   Nur aussagenlogische Konstanten $p_j^0 (j \geq 1)$ sind zugelassen.

$$(\textbf{If } p_1 \textbf{ Then } p_2 \textbf{ Else } p_3) \leftrightarrow ((p_1 \rightarrow p_2) \vee (\neg p_1 \rightarrow p_3))$$

5. Sprache der monadischen Logik (2-Stufe)
   Individuenvariablen x_i, keine Funktionssymbole.
   Monadische Prädikatsvariablen + Konstanten: p_j^1, P_j^1

$$\forall P \forall x \forall y ((P(x) \rightarrow p(y)) \rightarrow p(x))$$

Semantik der P-Logik 2-Stufe
Interpretationen, Belegungen, Bewertungen

	$D \neq \emptyset$,	Funktionen	$f : D^n \rightarrow D$ (totale Funktion),
!		Prädikate	$P \subseteq D^n$ (als Relationen)
			oder $P : D^n \rightarrow \{W, F\}$
!	0-stellige	Funktionen (Element aus D),	
		Prädikate (Element aus $\{W, F\}$).	

Definition 3.4 (**Interpretationen, Belegungen**)

Sei $\mathcal{L}$ Sprache der P-Logik 2-Stufe
(festgelegt durch die Menge der Konstantensymbole i. R. endlich).

a) Eine **Interpretation** I für $\mathcal{L}$ ist ein Tripel $I = (D, I_c, I_v)$ mit
   - $D \neq \emptyset$ Individuenbereich (Definitionsbereich).
   - I_c ist eine Interpretation (Belegung) der Konstanten
     $f^n \in \mathcal{L}$, so $I_c(f^n) : D^n \to D$
     $p^m \in \mathcal{L}$, so $I_c(p^m) \subseteq D^m$(oder $: D^m \to \{W, F\}$)
   - I_v ist eine Belegung der Variablen:
     F^n Funktionsvariablen $I_v(F^n) : D^n \to D$ $(n \geq 0)$
     P^m Prädikatsvariablen $I_v(P^m) \subseteq D^m(D^m \to \{W, F\})$

   (D, I_c) heißt auch **Relationalstruktur.**
   Kommen keine Prädikatskonstanten vor, so **Algebra.**

Interpretationen, Belegungen (Forts.)

b) Fortsetzung von I auf **Term**, bzw. **Form**:
I : **Term** $\to D$ $\quad$ I : **Form** $\to \mathbb{B}$

i) Bewertung der Terme:

- $I(a_i) = I_c(a_i) \qquad I(x_i) = I_v(x_i)$
- $I(f(t_1, \ldots, t_n)) = I_c(f)(I(t_1), \ldots, I(t_n))$
- $I(F(t_1, \ldots, t_n)) = I_v(F)(I(t_1), \ldots, I(t_n))$
- $I(\textbf{if } A \textbf{ then } t_1 \textbf{ else } t_2) = \begin{cases} I(t_1) & \text{falls } I(A) = 1 \ \ (W) \\ I(t_2) & \text{falls } I(A) = 0 \ \ (F) \end{cases}$

ii) Bewertung der Formeln:

- $I(W) = 1 \qquad I(F) = 0 \qquad I(p^0) = I_c(p^0) \qquad I(P^0) = I_v(P^0)$
- $I(p(t_1, \ldots, t_n)) = I_c(p)(I(t_1), \ldots, I(t_n))$
- $I(P(t_1, \ldots, t_n)) = I_v(P)(I(t_1), \ldots, I(t_n))$
- $I(t_1 = t_2) = \begin{cases} 1 & \text{falls } I(t_1) =_D I(t_2) \\ 0 & \text{sonst} \end{cases}$

Interpretationen, Belegungen (Forts.)

b) ii)

- $I(\neg A)$, $I(A \wedge B)$, $I(A \vee B)$, $I(A \rightarrow B)$, $I(A \leftrightarrow B)$, $I(\textbf{If } A \textbf{ Then } B \textbf{ Else } C)$
  Wie in A-Logik.
- $I((\forall x)A) = \begin{cases} 1 & \text{falls für alle } d \in D \text{ gilt } I^{x,d}(A) = 1 \\ 0 & \text{sonst} \end{cases}$
- $I((\exists x)A) = \begin{cases} 1 & \text{falls es } d \in D \text{ gibt mit } I^{x,d}(A) = 1 \\ 0 & \text{sonst} \end{cases}$

  $I^{x,d} = (D, I_c, I'_v)$, $\qquad I'_v(y) = \begin{cases} d & y \equiv x \\ I_v(y) & \text{sonst} \end{cases}$
- Entsprechend für Quantifizierungen mit den anderen Funktions- und Prädikatsvariablen.

Interpretationen, Belegungen (Forts.)

Jede Interpretation $I = (D, I_c, I_v)$ induziert durch 3 (i) und ii)) eine Bewertung aller Terme und Formeln, die die bewerteten Konstanten und Variablen als freie Variablen enthalten.

Umgekehrt wird jede Bewertung I, die i) und ii) genügt eindeutig durch eine solche Interpretation induziert.

Beachte: 3-Parameter: D, I_c, I_v.

Interpretationen, Belegungen (Forts.)

c) Gilt $I(A) = 1$, so ist A wahr in der Interpretation I oder I **erfüllt** A.

Schreibweise $\models_I A$ oder $I \models A$

(Beachte A ist hier eine beliebige Formel, kann also freie Variablen enthalten).

Folgerungen

Bemerkung 3.5

- *Um die Bewertung einer Formel A zu bestimmen, genügt es die Bewertung der in ihr vorkommenden Konstanten und frei vorkommenden Variablen zu kennen!!*
  $I = (\underline{D}, \underline{I_c}, I_v)$
- *Insbesondere: Ist A abgeschlossen, so genügt es Interpretationen der Form (D, I_c), d. h. Definitionsbereich und Belegung der Konstanten zu betrachten.*
- *Seien I_1, I_2 Interpretationen mit $D_1 = D_2$ und A eine Formel. Stimmen I_1 und I_2 auf allen Konstanten und freien Variablen, die in A vorkommen, überein, so gilt $I_1(A) = I_2(A)$.*

Beispiel

Beispiel 3.6

i) $\exists x \forall y (p(y) \rightarrow x = y)$
Stimmt in allen Interpretationen für die $I(p)$ höchstens ein Element enthält. „Es gibt höchstens ein x, so dass $p(x)$ wahr ist“.

ii) „Es gibt genau ein x, so dass $p(x)$ wahr ist“.

$$\exists x[p(x) \land \forall y[p(y) \rightarrow x = y]]$$

iii) $\forall z \exists u \exists v((z = u \lor z = v) \land u \neq v)$
$\land \forall x \forall y \forall P[x \neq y \lor P(x,x) \lor \neg P(y,y)]$

- Wahr in jeder Interpretation mit $|D| \geq 2$
- Falsch in jeder Interpretation mit $|D| = 1$

Beispiel (Fort.)

iv) Eigenschaften von Relationen: Reflex., Sym., Tra.

- $\forall x \; p(x, x)$
- $\forall x \forall y (p(x, y) \rightarrow p(y, x))$
- $\forall x \forall y \forall z [(p(x, y) \wedge p(y, z)) \rightarrow p(x, z)]$

v) Relationen, Funktionen

- $A \equiv \forall x \; p(x, f(x))$, $A_1 \equiv p(x, f(x))$
  $I = (\mathbb{N}, I_c, I_v)$, $I_c(p) \equiv \; \leq$ -Prädikat, $I_c(f) : n \rightarrow n^2$,

  $I^{x,n}(A_1)(\equiv n \leq n^2) = 1$

Definition

Definition 3.7

Sei $\mathcal{L}$ Sprache der P-Logik 2-Stufe

a) $A \in$ **Form** heißt **allgemeingültig,** falls $I(A) = 1$ für jede Interpretation I für $\mathcal{L}$.
Schreibweise: $\models A$

b) $A \in$ **Form** heißt **erfüllbar,** falls es eine Interpretation I für $\mathcal{L}$ gibt mit $I(A) = 1$. I heißt auch Modell für A (nur für abg. A). Gibt es keine solche Interpretation, so heißt A **unerfüllbar**.

c) $\Sigma \subseteq$ **Form** heißt **erfüllbar,** falls es eine Interpretation I für $\mathcal{L}$ gibt, die alle Formeln $A \in \Sigma$ erfüllt.

Einfache Folgerungen

Bemerkung 3.8

A allgemeingültig gdw $\neg A$ *unerfüllbar.*
Es genügt Interpretationen zu betrachten, die die Konstanten und freien Variablen der Formel A belegen.

- unendlich viele, da $D \neq \emptyset$ *beliebig.*

Bemerkung 3.9

Es gibt allgemeingültige Formeln: **Tautologie-Theorem:**
Sei $A(p_1, \ldots, p_n)$ *eine Formel der Aussagenlogik in A-Variablen* $p_1, \ldots, p_n$. A' *entstehe aus A durch simultane Ersetzung von* p_i *durch* $B_i \in$ **Form**. *Dann* $A' \in$ **Form**.
Ist A Tautologie, so ist A' *allgemeingültig.*
(z. B. $A_1 \vee \neg A_1, A_1 \rightarrow (A_2 \rightarrow A_1) \ldots$*)*

Einfache Folgerungen

Bemerkung 3.10

i) $\forall x \forall y \forall P(x \neq y \vee (P(x,x) \vee \neg P(y,y)))$ *ist allgemeingültig.*
Es genügt Interpretationen mit $I = (D)$ $D \neq \emptyset$ *zu betrachten.*
$|D| = 1$ *ok,* $|D| > 1$
$I_v(x, y, P)$, $x \to d_1$, $y \to d_2$
$d_1 \neq d_2$ *ok,* $d_1 = d_2 \rightsquigarrow I_v(P)(d_1, d_2) = \begin{cases} 1 \\ 0 \end{cases}$

ii) $A \equiv \exists P \forall x \exists y (P(x,x) \wedge \neg P(x,y))$
Weder allgemeingültig noch unerfüllbar.
$|D| = 1 \rightsquigarrow I(A) = 0$, $|D| \geq 2 \rightsquigarrow I(A) = 1$

iii) *Allgemeingültig sind:*
$t = t$, $\forall x A \leftrightarrow \neg \exists x (\neg A)$, $\exists x A \leftrightarrow \neg \forall x (\neg A)$

Ordnungsrelationen

Bemerkung 3.11 (Eigenschaften von Ordnungsrelationen:)

p 2-stellige P-Konstante

$A_1 \equiv \forall x \forall y \forall z((p(x,y) \wedge p(y,z)) \rightarrow p(x,z))$ *Tra.*

$A_2 \equiv \forall x \forall y(p(x,y) \vee p(y,x) \vee x = y)$ *Trichot.*

$A_3 \equiv \forall x \neg p(x,x)$ *Antireflex.*

$A_4 \equiv \forall x \forall y(p(x,y) \rightarrow \exists z(p(x,z) \wedge p(z,y)))$ *dicht*

$A_5 \equiv \forall x \exists y \; p(x,y)$ *ohne letztes Elem.*

$A_6 \equiv \forall x \exists y \; p(y,x)$ *ohne erstes Elem.*

Keine der Formeln ist allgemeingültig! (Überzeugen Sie sich)

Sie sind erfüllbar: $I_1 = (\{0,1,2\}, <)$

$I_2 = (\mathbb{N}, <)$

$I_3 = ([0,1], <)$

$I_4 = (\mathbb{Q}, <)$

$<$	0	1	2
0	*F*	*W*	*W*
1	*F*	*F*	*W*
2	*F*	*F*	*F*

Einige typische Formeln

Beispiele

Allgemeingültige Formeln	Nicht-Allgemeingültige Formeln
$\forall x p(x) \rightarrow \exists x p(x)$	$\exists x p(x) \rightarrow \forall x p(x)$
$p(x) \rightarrow p(x)$	$p(x) \rightarrow p(a)$
$\forall x q(x) \rightarrow q(a)$	$q(a) \rightarrow \forall x q(x)$
$p(a) \rightarrow \exists x p(x)$	$\exists x p(x) \rightarrow p(a)$
$\forall x \forall y (p_1(x,y) \rightarrow p_1(x,y))$	$\forall x \forall y (p_1(x,y) \rightarrow p_1(y,x))$
$\exists y \forall x p_1(x,y) \rightarrow \forall x \exists y p_1(x,y)$	$\forall x \exists y p_1(x,y) \rightarrow \exists y \forall x p_1(x,y)$
$(\forall x p(x) \vee \forall x q(x)) \rightarrow$ $\forall x (p(x) \vee q(x))$	$\forall x (p(x) \vee q(x)) \rightarrow$ $\forall x p(x) \vee \forall x q(x)$
Zeige $\neg A$ unerfüllbar	Zeige $\neg A$ erfüllbar $I = (\mathbb{Z}, p(x) \Leftrightarrow x > 0,$ $q(x) : x \leq 0, p_1(x,y) : x > y$ $a \leftarrow 0, x \leftarrow 1)$

Arithmetik

Beispiel 3.12

Die Sprache der Arithmetik $\mathbb{N}$

- *Konstanten:* $0, S, +, \cdot,$ $\quad I = (\mathbb{N}, 0, ', +, \cdot)(n' = n + 1)$
- *Stelligkeiten* $0, 1, 2, 2$
  1. $\forall x \forall y (S(x) = S(y) \rightarrow x = y)$
  2. $\forall x \quad S(x) \neq 0$
  3. $\forall x \quad x + 0 = x$
  4. $\forall x \forall y (x + S(y)) = S(x + y)$
  5. $\forall x \quad x \cdot 0 = 0$
  6. $\forall x \forall y \quad x \cdot S(y) = (x \cdot y) + x$
  7. $\forall P[(P(0) \wedge \forall x(P(x) \rightarrow P(S(x)))) \rightarrow \forall x P(x)]$
- *Sind gültig in* I.
  Man beachte 7. ist Induktionsprinzip für Teilmengen von $\mathbb{N}$.
  Es ist eine Formel der P-Logik 2-Stufe.

Arithmetik Beispiel (Forts.)

Frage nach der **Axiomatisierbarkeit** der Arithmetik:

- Ist die Allgemeingültigkeit für Formeln einer Sprache $\mathcal{L}$ entscheidbar?
- Rekursiv aufzählbar?
- Welche effektiven Methoden gibt es?

Allgemeingültigkeit: Entscheidbare Fälle

Lemma 3.13

Die Allgemeingültigkeit für Formeln der quantifizierten A-Logik ist entscheidbar.

Beweis:

Methode der **Quantorenelimination**: Finde zu Formel der Q-A-Logik eine logisch äquivalente der A-Logik.
(Problemreduktion!)

$(\forall P_i^0)B \leftrightarrow B_{P_i^0}[W] \wedge B_{P_i^0}[F] \qquad (P_i^0 \leftarrow W,\ P_i^0 \leftarrow F)$

$(\exists P_i^0)B \leftrightarrow B_{P_i^0}[W] \vee B_{P_i^0}[F]$

$I(\qquad) = I(\qquad)$

- Nach Transformation bleibt eine Formel der A-Logik: Entscheide, ob diese eine Tautologie ist.

Allgemeingültigkeit: Entscheidbare Fälle (Forts.)

Beispiel 3.14

$\forall P \exists Q ((P \leftrightarrow \neg Q) \vee (p \rightarrow Q))$ ist Allgemeingültig

$\rightsquigarrow$

$\exists Q((W \leftrightarrow \neg Q) \vee (p \rightarrow Q)) \wedge \exists Q((F \leftrightarrow \neg Q) \vee (p \rightarrow Q))$

$\rightsquigarrow$

$[((W \leftrightarrow \neg W) \vee (p \rightarrow W)) \vee ((W \leftrightarrow \neg F) \vee$
$(p \rightarrow F))] \wedge [(F \leftrightarrow \neg W) \vee (p \rightarrow W) \vee ((F \leftrightarrow \neg F) \vee (p \rightarrow F))]$

$\rightsquigarrow W \wedge W \qquad \rightsquigarrow W$

Allgemeingültigkeit: Entscheidbare Fälle (Forts.)

Ausblick

Andere:

- Gleicheitslogik
- Monadische P-Logik 1-Stufe mit $=$
- Monadische P-Logik 2-Stufe mit $=$
- Pressburgerarithmetik: Gültige PL1-Formeln in $(\mathbb{N}, 0,', +, <)$
- Syntaktisch eingeschränkte Formelnklassen
  $\forall(\quad), \forall\exists, \exists\forall, \ldots$?

Transformationen von Termen und Formeln

Einschränkung auf PL1

Definition 3.15 (Substitution)

$A \in \textbf{Form}, t, \hat{t} \in \textbf{Term}, x$ Individuen-Variable.

- **Substitution** von x durch t in A bzw. (in $\hat{t}$):
  - $A_x[t]$ $(\hat{t}_x[t])$ ist die Formel (der Term), die aus A (bzw. $\hat{t}$) entsteht, wenn man jedes freie Vorkommen von x in A (bzw. $\hat{t}$) durch t ersetzt.
- Analog **simultane Substitutionen**
  - $A_{x_1,\dots,x_n}[t_1, \dots, t_n]$ bzw. $t_{x_1,\dots,x_n}[t'_1, \dots, t'_n]$

Transformationen von Termen und Formeln (Forts.)

- Allgemeiner ist eine **Substitution** durch $\sigma : \{x_i : i \in \mathbb{N}\} \to$ **Term** gegeben.
  $\sigma(A)$, $\sigma(t)$ können entsprechend durch Induktion über den Aufbau der Formeln bzw. Terme definiert werden.
- Die Substitution heißt **erlaubt**, falls kein Vorkommen einer Variablen in t bzw. t_i nach der Substitution in A gebunden vorkommt.
- Dies ist der Fall z. B. wenn die Variablen von t nicht in A vorkommen.
  (Kann durch Umbenennung der gebundenen Variablen erreicht werden).

Beispiel 3.16 (Substitutionen)

$A \equiv \exists y \;\; x = 2 \cdot y \qquad t \equiv y + 1$

- $A_x[t] \equiv \exists y \quad y + 1 = 2 \cdot y$ — keine erlaubte Substitution
- $A_y[t] \equiv \exists y \quad x \equiv 2 \cdot y$ — da keine freie Vorkommen
- $t_y[t] \equiv (y + 1) + 1$

- Wie Ändert sich die Bedeutung einer Formel bei Substitutionen?
  - $I = (\mathbb{N}, +, \cdot) \quad x \leftarrow 3 \quad y \leftarrow 2 \quad t \leftarrow 3 \quad I(x) = I(t)$
- Ersetzt man x durch t, sollte sich die Bedeutung einer Formel nicht verändern.
  - $I(A) = I(\exists y \; x = 2 \cdot y) = 0$
  - $I(A_x[t]) = I(\exists y \; y + 1 = 2y) = 1$ (keine erlaubte Substitution)

  Betrachte die Formel:
  - $B \equiv \forall x(P(x, y) \rightarrow Q(x)) \quad t \equiv f(y, z)$

  $\hookrightarrow$ $B_y[t] \equiv \forall x(P(x, f(y, z)) \rightarrow Q(x))$ erlaubte Substitution.

Lemma 3.17 (Substitutionslemma)

*Sei A Term oder Formel, x Individuenvariable, $t \in$ **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: Sind I, I' Interpretationen, die sich höchstens für x unterscheiden und gilt $I'(x) = I(t)$, so gilt $I'(A) = I(A_x[t])$.

- **Beweis:**
  Induktion über Aufbau von Formeln bzw. Terme. ■

Folgerungen aus Substitutionslemma

Folgerung 3.18

Sei $A_x[t]$ erlaubt.

a) *Ist A allgemeingültig, so auch $A_x[t]$.*
b) $\forall xA \rightarrow A_x[t]$ *ist allgemeingültig.*
c) *Spezialfälle: allgemeingültig sind:*
   $\forall xA \rightarrow A, \qquad A \rightarrow \exists xA$
d) *Falls Substitution nicht erlaubt, so gilt das Lemma nicht:*
   $A \equiv \exists y(S(x) = y),\ A_x[y] \equiv \exists y(S(y) = y)$
   $I = (\mathbb{N}, \cdots) \quad I(A) = 1 \quad I(A_x[y]) = 0$
   $\forall x \exists y(S(x) = y)$ *allgemeingültig.*
   $\forall x \exists y(S(x) = y) \rightarrow \exists y(S(y) = y)$ *nicht allgemeingültig.*

Universeller und Existentieller Abschluss

- **Beachte**: Sei $A(x_1, \ldots, x_n)$ Formel in der $x_1, \ldots, x_n$ frei vorkommen, dann gilt
  - A allgemeingültig gdw $\forall x_1 \cdots \forall x_n A$ allgemeingültig (**universeller Abschluss**)
  - A erfüllbar gdw $\exists x_1 \cdots \exists x_n A$ erfüllbar (**existentieller Abschluss**)
  - Sind $A, A \to B$ allgemeingültig, so ist auch B allgemeingültig.

Semantischer Folgerungsbefriff

Definition 3.19

Sei $\mathcal{L}$ eine (Teil-)Sprache der PL2
$\Gamma \subseteq \textbf{Form}, A, B \in \textbf{Form}$

a) A ist **logische Folgerung** aus Γ : $\Gamma \models A$. Wenn jede Interpretation, die Γ erfüllt auch A erfüllt
$(I(\Gamma) = 1 \rightsquigarrow I(A) = 1)$.
   - Sei $\mathrm{Folg}(\Gamma) = \{A \in \textbf{Form} : \Gamma \models A\}$.

b) A und B sind **logisch äquivalent** $A \models\!\!\!\Dashv B$, falls
$A \models B$ und $B \models A$.
(Lässt sich auf Mengen verallgemeinern!)

Bemerkung 3.20

1. $\Gamma \models A$ gdw $\{\Gamma, \neg A\}$ *nicht erfüllbar.*
2. $\emptyset \models A$ gdw $\models A$ *(d.h. A ist allgemeingültig).*
3. Γ *nicht erfüllbar* gdw $\Gamma \models A$ *für alle* $A \in$ **Form**.
4. $\Gamma \subset \Sigma, \Gamma \models A$, *so* $\Sigma \models A$. *(Monotonieeigenschaft)*
5. $\Gamma \models\!\!\!\!= \!\!\dashv \Sigma$, *d. h.* $\Gamma \models B$ *für* $B \in \Sigma$ *und* $\Sigma \models C$ *für* $C \in \Gamma$.
   Insbesondere Γ *erfüllbar* gdw Σ *erfüllbar und*
   $\Gamma \models A$ gdw $\Sigma \models A$. *Also* $\mathrm{Folg}(\Gamma) = \mathrm{Folg}(\Sigma)$.
6. $A \models\!\dashv B$ gdw $\models A \leftrightarrow B$ gdw $I(A) = I(B)$ *für jede Interpretation* I.
   $A \models\!\dashv B$ *dann* $\Gamma \models A$ gdw $\Gamma \models B$.

Beispiel 3.21

i) $\forall x Q(x) \models Q(y)$
(Spezialfall von $\forall x A \rightarrow A_x[t]$ allgemeingültig)

ii) $A(y) \not\models \forall y A(y)$ (y kommt frei in A vor)
$A(y) \equiv p(y)$, $I = (\{0,1\}, I(p)(x) \equiv (x=0), y \leftarrow 0)$
$I(A(y)) = 1$, jedoch $I(\forall y A(y)) = 0 \quad I(p)(1) = 0$

iii) $\models \exists x(p(x) \rightarrow \forall x p(x))$
Sei $\mathcal{I} = (D, I(p))$ eine Interpretation, d. h. $I(p) \subseteq D$
$I(\exists x(p(x) \rightarrow \forall x p(x)) = 1 \text{ gdw}$ es gibt $d \in D$, so dass
$I' = (D, I(p), x \leftarrow d)$, $I'(p(x) \rightarrow \forall x p(x)) = 1$
$\text{gdw } d \notin I(p)$ oder $I'(\forall x p(x)) = 1$ für ein $d \in D$
gdw es gibt ein $d \in D$ mit $d \notin I(p)$ oder $I(p) = D$

iv) Beispiele für äquivalente Formeln

- $\neg\neg A \models\!\!\!\!= A$
- $W \vee A \models\!\!\!\!= A \vee W \models\!\!\!\!= W$
  $F \wedge A \models\!\!\!\!= A \wedge F \models\!\!\!\!= F$
  $A \vee A \models\!\!\!\!= A \qquad A \wedge A \models\!\!\!\!= A$
- $B \circ QvA \models\!\!\!\!= Qv(B \circ A)$ $\qquad$ Q Quantor, v nicht frei in B
- Z.B. $B \vee QvA \models\!\!\!\!= Qv(B \vee A)$
- $\forall vA \models\!\!\!\!= \neg\exists v(\neg A)$, $\exists vA \models\!\!\!\!= \neg\forall v(\neg A)$
- $\forall xA(x) \rightarrow B \models\!\!\!\!= \exists y(A(y) \rightarrow B)$
  falls y weder in $A(x)$ noch in B frei vorkommt
- $\forall x(A \rightarrow B) \models \forall xA \rightarrow \forall xB$
- $\forall vA \models\!\!\!\!= \forall yA_v[y]$ $\exists vA \models\!\!\!\!= \exists yA_v[y]$
  Falls Sub. erlaubt und y nicht frei in A
- **Beachte**: $\forall vB \models\!\!\!\!= B \qquad \exists vB \models\!\!\!\!= B$
  Falls v nicht frei in B vorkommt.

Satz 3.22 (Wichtige Sätze)

Sei $\Gamma \subseteq$ **Form**, $A, B \in$ **Form**.

1. **Deduktionstheorem** • $\Gamma, A \models B$ gdw $\Gamma \models A \rightarrow B$
2. **Modus-Ponens-Regel** • $\Gamma \models A, \Gamma \models A \rightarrow B$ *so* $\Gamma \models B$
3. **Kontrapositionsregel** • $\Gamma, A \models \neg B$ gdw $\Gamma, B \models \neg A$
4. **Generalisierungs-Theorem**
   Kommt $v \in Var$ *in keiner Formel von* Γ *frei vor, so*
   - $\Gamma \models A$ gdw $\Gamma \models \forall v\ A$

   Insbesondere: $A \models \forall v\ A$ *bzw.* $\models A \rightarrow \forall v\ A$,
   falls v *nicht frei in* A *vorkommt.*
5. **Ersetzungstheorem**
   Sei $A' \in$ **Form**, A *Teilformel von* A'. *Entsteht* B' *aus* A' *indem man einige Vorkommen der Teilformel* A *durch* B *ersetzt und gilt* $A \models\!\!\!\dashv B$, *so auch* $A' \models\!\!\!\dashv B'$.

Beispiel 3.23 (Anwendung der Sätze)

a) $\models \exists x \forall y A \rightarrow \forall y \exists x A$

gdw $\underbrace{\exists x \forall y A}$ $\models \forall y \exists x A$	Deduktionstheorem
gdw $\exists x \forall y A \models \exists x A$	Generalisierungstheorem
gdw $\neg \forall x \neg \forall y A \models \neg \forall x \neg A$	Ersetzungstheorem
gdw $\forall x \neg A \models \forall x \neg \forall y A$	Kontrapositionsregel
gdw $\forall x \neg A \models \neg \forall y A$	Generalisierungstheorem
gdw $\{\forall x \neg A, \forall y A\}$	nicht erfüllbar

b) Variante des Ersetzungstheorems
A' entstehe aus A durch Substitution (erlaubte) einiger Vorkommen von x in A durch y. Dann gilt
$\models \forall x \forall y (x = y \rightarrow (A \leftrightarrow A'))$
(z. B. $A \equiv f(x, y) = g(x) \qquad A' \equiv f(y, y) = g(x)$)

Normalformen

Definition 3.24 (Normalformen (Präfix Normalformen))

Eine Formel ist in **PKNF** (Pränex Konjunktiver NF), falls sie die Gestalt

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

$\Delta \in \{\exists,\forall\}$, v_j Variablen, die in mindestens einem A_{kl} vorkommen und paarweise verschieden sind.

A_{kl} Literale, d. h. atomare oder negierte atomare Formeln.

Beispiel 3.25

$\forall x \exists Q \forall y \{[\neg p \vee x \neq a \vee x = b] \wedge [Q(y) \vee y = b]\}$

Normalformen

Satz 3.26 (Verfahren PKNF)

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

(Beachte die Länge der Formel kann exponentiell in der Länge der ursprünglichen Formel wachsen. Dies kann man vermeiden, wenn man nur erfüllungsäquivalente Formeln benötigt!).

Verfahren PKNF, PNDF

Schritte:

1. Eliminiere überflüssige Quantoren.
2. Umbenennung gebundener Variablen.
3. Eliminiere logische Verknüpfungen und Operatoren.
   $\rightarrow, \leftrightarrow,$ **If** $\ldots,$ **if** $\ldots$
4. NNF: Negation vor Atome.
5. Quantoren nach außen.
6. Matrix in KNF (DNF).

Verfahren PKNF, PDNF - Beispiele

Beispiel 3.27

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

1. $\forall x[(p(x) \vee \forall z q(z, y)) \rightarrow \neg \forall y r(x, y)]$
2. $\forall x[(p(x) \vee \forall z q(z, y)) \rightarrow \neg \forall y' r(x, y')]$
3. $\forall x[\neg(p(x) \vee \forall z q(z, y)) \vee \neg \forall y' r(x, y')]$
4. $\forall x[(\neg p(x) \wedge \exists z \neg q(z, y)) \vee \exists y' \neg r(x, y')]$
5. $\forall x \exists z \exists y'[(\neg p(x) \wedge \neg q(z, y)) \vee \neg r(x, y')]$
6. $\forall x \exists z \exists y'[(\neg p(x) \vee \neg r(x, y')) \wedge (\neg q(z, y) \vee \neg r(x, y'))]$ (Ist PKNF)
7. PDNF $\forall x \exists z \exists y'[(\neg p(x) \wedge \neg q(z, y)) \vee \neg r(x, y')]$

Unentscheidbarkeit der Allgemeingültigkeit

Satz 4.1

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

- *(In den Formeln kommt keine „$=$“ vor).*

Beweis:

Reduziere effektiv unentscheidbares Problem (z.B. PCP) auf die Entscheidung der Allgemeingültigkeit bestimmter Formeln.

Unentscheidbarkeit der Allgemeingültigkeit (Forts.)

- **PCP**: $\Sigma = \{0,1\} \quad S = \{(\alpha_1, \beta_1), \ldots, (\alpha_n, \beta_n)\} \quad n \geq 1$
  mit $\alpha_i, \beta_i \in \Sigma^*$
  S hat eine **Lösung** gdw es gibt $j_1, \ldots, j_l \in [1 \ldots n], l > 0$ mit

$$\alpha_{j1}\alpha_{j2} \cdots \alpha_{jl} \equiv \beta_{j1}\beta_{j2} \cdots \beta_{jl}$$

- z. B.
  $S = ((0, 000), (0100, 01), (001, 1))$ hat die Lösung
  $j_1 = 1, j_2 = 3$, $0001 \equiv 0001$.
- **Problem PCP**: Eingabe S. Entscheide, ob S Lösung hat.
  Ist nicht rekursiv entscheidbar (aber rekursiv aufzählbar).

$\hookrightarrow$ **Reduktion**: Zu S berechne eine Formel A_S mit
S hat Lösung gdw A_S ist allgemeingültig.

Unentscheidbarkeit der Allgemeingültigkeit (Forts.)

Sprache PL1 mit:

- a F-Konstante, f_0, f_1 1-st. F-Konstanten, p 2-stellige P-Konstante.
- $\sigma_i \in \{0,1\}$ $(1 \leq i \leq m)$, so $f_{\sigma_m}(\cdots(f_{\sigma_2}(f_{\sigma_1}(x)))\cdots)$ Term, als Kodierung von Wörter: $f_{\underbrace{\sigma_1\sigma_2\ldots\sigma_m}_{\in\Sigma^*}}(x)$. Beachte die Reihenfolge!
- $S = \{(\alpha_1,\beta_1),(\alpha_2,\beta_2),\cdots,(\alpha_n,\beta_n)\}$ $n \geq 1$ $\Sigma = \{0,1\}$

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

Beispielreduktion

Beispiel 4.2

- Die zugeordnete Formel A_S zu $S = ((0, 000), (0100, 01), (001, 1))$ ist:

  $$[p(f_0(a), f_{000}(a)) \wedge p(f_{0100}(a), f_{01}(a)) \wedge p(f_{001}(a), f_1(a)) \wedge \forall x \forall y [\bigwedge_{1 \le j \le 3} p(x, y) \to p(f_{\alpha_j}(x), f_{\beta_j}(x))]] \to \exists z p(z, z)$$

- **Behauptung**:
  S hat Lösung gdw A_S allgemeingültig.

Beweis der Behauptung

:

„$\curvearrowleft$“: Angenommen A_S allgemeingültig, I Interpretation mit
$D = \{0,1\}^*$, $a \leftarrow \varepsilon$, $f_0 : x \to x0$, $f_1 : x \to x1$
$p(x,y)$ gdw $x \equiv \alpha_{j_1} \cdots \alpha_{j_m}$, $y \equiv \beta_{j_1} \ldots \beta_{j_m}$ für $j_1, \ldots, j_m$
$(j_i \in [1 \ldots n] \quad m > 0)$.

„$\curvearrowright$“: Angenommen PCP S habe Lösung $j_1 \ldots j_m$, d. h.
$\alpha_{j_1}\alpha_{j_2} \cdots \alpha_{j_m} \equiv \beta_{j_1}\beta_{j_2} \ldots \beta_{j_m}$
$\rightsquigarrow p(f_{\alpha_{j_1}\alpha_{j_2}\cdots\alpha_{j_m}}(a), f_{\beta_{j_1}\beta_{j_2}\ldots\beta_{j_m}}(a))$ wahr,
also ist A_S allgemeingültig.

! Es gibt weitere Unentscheidbarkeitsresultate die wir später noch behandeln werden. Es sei jedoch erwähnt, dass die Grenzen zwischen den entscheidbaren und unentscheidbaren Fälle der Allgemeingültigkeit sehr genau bekannt sind. (Siehe etwa Börger).

Hauptsätze der Prädikatenlogik erster Stufe

Sei $\mathcal{L}$ Sprache der PL1

Satz 4.3

a) *Die Menge der allgemeingültigen Formeln* $\{A \in \textbf{Form}(\mathcal{L}) : \ \models A\}$ *ist* **rekursiv aufzählbar** *(i. Allg. nicht rekursiv entscheidbar).*

Es gibt ein rekursives deduktives System $\mathcal{F}$ für $\mathcal{L}$ mit

$$\vdash_{\mathcal{F}} A \ \text{ gdw } \ \models A \quad (A \in \textbf{Form}(\mathcal{L}))$$

Hauptsätze der Prädikatenlogik erster Stufe (Forts.)

b) **Kompaktheitssatz für PL1**: *Sei* $\Sigma \subseteq \mathbf{Form}(\mathcal{L})$.
Σ *erfüllbar* gdw *jede endliche Teilmenge von* Σ *ist erfüllbar*.

c) Insbesondere: $\Sigma \subseteq \mathbf{Form}(\mathcal{L}), A \in \mathbf{Form}(\mathcal{L})$:

$$\Sigma \models A \text{ gdw es gibt } \Sigma_0 \subseteq \Sigma, \Sigma_0 \text{ endlich} : \Sigma_0 \models A$$

Hauptsätze der Prädikatenlogik erster Stufe (Forts.)

d) **Satz von Löwenheim-Skolem**:
$\Sigma \subseteq \textbf{Form}(\mathcal{L})$ *erfüllbar* gdw *es gibt eine Interpretation* $I = (D, I_c, I_v)$, *wobei* D *abzählbar oder endlich ist, die* Σ *erfüllt*.

(D kann als eine Termmenge über die konstanten Symbole gewählt werden).

Beachte jedoch: nicht gültig für PL2

Negative Ergebnisse für die Prädikatenlogik zweiter Stufe

Satz 4.4 (von Gödel)

Für Sprachen von PL2- und PL2 Formeln:

a) *Die Menge der allgemeingültigen Formeln 2-Stufe für PL2-Sprachen ist nicht rekursiv aufzählbar.*
b) *Es gibt kein rekursives „deduktives System“, dessen Theoreme die Menge der allgemeingültigen Formeln zweiter Stufe sind.*
c) *Es gibt erfüllbare Mengen von Formeln 2-Stufe, die keine abzählbaren Modelle haben.*

Deduktive Systeme für PL1

Definition 4.5 (Deduktive Systeme für PL1)

Sei $\mathcal{L}$ Sprache 1-Stufe mit Formeln in $\neg, \rightarrow, \forall, =$. $\mathcal{F} = (\mathbf{Ax}, \mathbf{R})$ bestimmt durch Axiomenmenge Ax (Axiomenschema) und Menge R von Regeln (Regelschema).

- Ax enthält **alle Generalisierungen** von folgenden durch Schemata beschriebenen Formelmengen:

  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

Deduktive Systeme für PL1 (Forts.)

Ax5: $\forall x\,(A \rightarrow B) \rightarrow (\forall x\, A \rightarrow \forall x\, B)$
(Ax5': $\forall x(A \rightarrow B) \rightarrow (A \rightarrow \forall x B)$ x nicht frei in A)
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 durch Ersetzen einiger freier Vorkommen von x durch y (erlaubt)

- R enthält alle Regeln, die vom **Regelschema Modus Ponens**

MP $\dfrac{A, A \rightarrow B}{B}$ Modus Ponens beschrieben werden.

Deduktive Systeme für PL1 (Forts.)

- **Alternatives** Deduktionssystem $\mathcal{F}' = (\mathbf{Ax}, \mathbf{R}')$
- ► R' enthält MP-Regel und **Generalisierungsregel.**

  GR $\dfrac{A}{\forall x A}$ Generalisierung (ohne Einschränkungen)

! **Beachte**: Ax enthält nur allgemeingültige Formeln. MP und GR-Regel führen nicht aus der Menge der allgemeingültigen Formeln hinaus.

Ziel

- $\underset{\mathcal{F}}{\vdash} A$ gdw $\underset{\mathcal{F}'}{\vdash} A$ gdw $\models A$
  $\rightsquigarrow$ $\rightsquigarrow$ Korrektheit
- $\Sigma \underset{\mathcal{F}}{\vdash} A$ gdw $\Sigma \models A$
  $\rightsquigarrow$ Korrektheit
- $\Sigma \underset{\mathcal{F}}{\vdash} A$, so $\Sigma \underset{\mathcal{F}'}{\vdash} A$ Umkehrung i. Allg. nicht
- $\Gamma \underset{\mathcal{F}'}{\vdash} A$ $\not\rightsquigarrow$ $\Gamma \underset{\mathcal{F}}{\vdash} A$ (nur für Γ Abg. Formeln).

  z. B. $p(x) \underset{\mathcal{F}'}{\vdash} \forall x p(x)$ aber $p(x) \not\models \forall x p(x)$

- **Bemerkung**: Alle Tautologien (taut. Theorem) sind herleitbar in $\mathcal{F}$, d. h. Theoreme von $\mathcal{F}$.

Beispiele

Beispiel 4.6

1. $\underset{\mathcal{F}}{\vdash} \forall x\, (p(x) \rightarrow \exists y\, p(y))$

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

Beispiele (Forts.)

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

Beweis:

$$
\begin{array}{ll}
\vdash \forall x\, \neg A \to \neg A & \text{(Ax4)} \\
\vdash (\forall x\, \neg A \to \neg A) \to (A \to \neg\forall x\, \neg A) & \text{(Ax3)} \\
\vdash A \to \neg\forall x\, \neg A & \text{(MP)} \\
\vdash \forall x\, A \to A & \text{(Ax4)} \\
\vdash (\forall x\, A \to A) \to & \\
\quad ((A \to \neg\forall x\, \neg A) \to (\forall x\, A \to \neg\forall x\, \neg A)) & \text{(Taut)} \\
\vdash (A \to \neg\forall x\, \neg A) \to (\forall x\, A \to \neg\forall x\, \neg A) & \text{(MP)} \\
\vdash \forall x\, A \to \exists x\, A & \text{(MP)}
\end{array}
$$

■

Beispiele (Forts.)

3. $\vdash t = t \qquad \vdash (x = y) \rightarrow (y = x)$
   $\vdash ((x = y) \rightarrow ((y = z) \rightarrow (x = z))$
   Folgen aus Ax, $\forall x(x = x)$, $(x = y) \rightarrow A(x) \leftrightarrow A(y))$ für A (erlaubt).
   $\vdash ((t_1 = t_1') \wedge \cdots \wedge (t_n = t_n')) \rightarrow (A(t_1, \ldots, t_n) \leftrightarrow A(t_1', \ldots, t_n'))$, wobei $A(x_1, \ldots, x_n)$ eine Formel mit mindestens n-freien Variablen und Substitutionen erlaubt ($x_i \leftarrow t_i$ bzw. $x_i \leftarrow t_i'$).
   Spezialfall:
   $\vdash ((t_1 = t_1') \wedge \cdots \wedge (t_n = t_n')) \rightarrow (f(t_1, \ldots, t_n) = f(t_1', \ldots, t_n'))$

Beweisidee

- $\forall x[(x = y) \to (A(x) \leftrightarrow A(y))]$
- $\forall x[(x = y) \to (A(x) \leftrightarrow A(y))] \to ((t = y) \to (A(t) \leftrightarrow A(y))$
- $(t = y) \to (A(t) \leftrightarrow A(y))$
- $\forall y(t = y) \to (A(t) \leftrightarrow A(y))$
- $(t = t') \to (A(t) \leftrightarrow A(t'))$

Deduktionstheorem - Generalisierungstheorem

Satz 4.7 (Hauptsätze für $\mathcal{F}$ und $\mathcal{F}'$)

Seien $\Gamma \subseteq \textbf{Form}, A, B \in \textbf{Form}$.

a) **Deduktionstheorem**
   1) $\Gamma \underset{\mathcal{F}}{\vdash} A \rightarrow B \;\text{ gdw }\; \Gamma, A \underset{\mathcal{F}}{\vdash} B$
   2) $\Gamma \underset{\mathcal{F}'}{\vdash} A \rightarrow B \;\text{ gdw }\; \Gamma, A \underset{\mathcal{F}'}{\vdash} B$, *falls die Generalisierung nicht auf eine in A frei vorkommende Variable angewandt wurde.*
   3) $\Gamma, A \underset{\mathcal{F}'}{\vdash} B \;\text{ gdw }\; \Gamma \underset{\mathcal{F}'}{\vdash} \tilde{A} \rightarrow B$,
   wobei $\tilde{A}$ ein universeller Abschluss von A ist.

b) **Generalisierungstheorem:**
   1) *Falls* $\Gamma \underset{\mathcal{F}}{\vdash} A$ *und x nicht frei in Γ vorkommt, so* $\Gamma \underset{\mathcal{F}}{\vdash} \forall x\, A$
   2) $\Gamma \underset{\mathcal{F}'}{\vdash} A \text{ gdw } \Gamma \underset{\mathcal{F}'}{\vdash} \forall x\, A$

c) **Kontrapositionstheorem:** *Für $\mathcal{F}$ und $\mathcal{F}'$*
   - $\Gamma, A \vdash \neg B \text{ gdw } \Gamma, B \vdash \neg A$

Deduktionstheorem - Generalisierungstheorem (Forts.)

$\hookrightarrow$ Es gelten somit für die hier vorgestellten prädikatenlogischen Systeme die für das deduktive System der Aussagenlogik entsprechenden Sätze. Vorsicht muss man beim System $\mathcal{F}'$ mit dem Deduktionstheorem haben, da wir dafür eine allgemeinere Generalisierungsregel zugelassen haben die semantisch nicht immer korrekt ist.

- Hinzu kommt das Generalisierungstheorem in den zwei Varianten.

Beweis Deduktionstheorem

1. „$\curvearrowright$“ Aus $\Gamma \vdash A \rightarrow B$ folgt auch $\Gamma, A \vdash A \rightarrow B$. Da auch $\Gamma, A \vdash A$ gilt, folgt $\Gamma, A \vdash B$, wegen MP ($\mathcal{F}$ und $\mathcal{F}'$).
   „$\curvearrowleft$“ Ang. $\Gamma, A \underset{\mathcal{F}}{\vdash} B$. Behauptung: $\Gamma \underset{\mathcal{F}}{\vdash} A \rightarrow B$
   - Induktion über Beweislänge: Axiom oder Hypothese

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

   - Schritt ist MP-Schritt

     $$\begin{array}{rll} \text{Schritt } j: & \Gamma, A \vdash C & \text{IV:} \Gamma \vdash A \rightarrow C \\ \vdots & \vdots & \\ \text{Schritt } k: & \Gamma, A \vdash C \rightarrow B & \text{IV}: \Gamma \vdash A \rightarrow (C \rightarrow B) \\ \vdots & \vdots & \text{Ax2+MP} \\ \text{Schritt } n+1: & \Gamma, A \vdash B & \Gamma \vdash A \rightarrow B \end{array}$$

Beweis Generalisierungstheorem

2. In $\mathcal{F}'$ „$\curvearrowleft$“ Generalisierungsregel

$$\begin{array}{l} \Gamma, A \vdash C \\ \quad\vdots \\ \Gamma, A \vdash \forall x C \quad x \text{ nicht frei in } A \end{array}$$

Dann IV:

$$\begin{array}{ll} \Gamma \vdash A \to C & \\ \Gamma \vdash \forall x(A \to C) & \text{(Gen)} \\ \Gamma \vdash \forall x(A \to C) \to (A \to \forall x C) & \text{(Ax5)} \\ & x \text{ nicht frei in } A \\ \Gamma \vdash A \to \forall x C & \end{array}$$

Definition 4.8

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

Konsistenz im deduktiven System

Bemerkung 4.9

- Γ *ist konsistent* gdw *jede endliche Teilmenge von* Γ *ist konsistent.*
- *Ist* Γ *inkonsistent, dann gilt* $\Gamma \underset{\mathcal{F}}{\vdash} A$ *für jede Formel* A.
- $\Gamma \cup \{\neg A\}$ *inkonsistent* gdw $\Gamma \vdash A$.
- $\Gamma \cup \{A\}$ *inkonsistent* gdw $\Gamma \vdash \neg A$.
- *Ist* Γ *inkonsistent, so ist* Γ *nicht erfüllbar: Sei nämlich* A *mit* $\Gamma \vdash A$ *und* $\Gamma \vdash \neg A$. I *Interpretation, die* Γ *erfüllt (wegen* $\Gamma \models A$ *und* $\Gamma \models \neg A$*) folgt aber* I *erfüllt* $\{A, \neg A\}$ $\lightning$
- *Die Menge der allgemeingültigen Formeln ist konsistent.*
- *Die Menge der Theoreme von* $\mathcal{F}$ $(\mathcal{F}')$ *ist konsistent.*

Vollständigkeit der Axiomatisierung

Satz 4.10 (Gödel)

Vollständigkeit der Axiomatisierung
Seien $A \in \textbf{Form}, \Sigma \subseteq \textbf{Form}$, *dann gilt:*

a) $\models A \ \text{ gdw } \ \underset{\mathcal{F}}{\vdash} A \ \text{ gdw } \ \underset{\mathcal{F}'}{\vdash} A.$

b) Σ *konsistent* gdw Σ *erfüllbar.*

c) $\Sigma \underset{\mathcal{F}}{\vdash} A \ \text{ gdw } \ \Sigma \models A.$

Beweis:

Siehe Yashuhara oder Enderton. ■

Theorien erster Stufe

Definition 4.11

Sei $\mathcal{L}$ eine Sprache erster Stufe (fixiert durch die Funktions- und -Prädikatskonstanten). $\Gamma \subseteq \mathbf{Form}(\mathcal{L})$ heißt **logische Theorie erster Stufe**, falls Γ abgeschlossen ist gegenüber logischer Folgerung, d. h. $A \in \mathbf{Form}$ $\quad \Gamma \models A$, so $A \in \Gamma$.

- **Beachte**: Alternative Definitionen in Literatur:
  - Γ Theorie, falls Γ abgeschlossen gegen MP und Generalisierung.
  - Γ Menge **abgeschlossener Formeln**, abgeschlossen gegen logische Folgerung.
- T als generische Bezeichnung für Theorien.

Theorien erster Stufe (Forts.)

Bemerkung 4.12

Sei $\mathcal{L}$ Sprache 1-Stufe.

a) $T_{\mathcal{L}} = \{A \mid A \in \mathbf{Form}(\mathcal{L}),\ \textit{allgemeingültig}\}$ *ist Theorie. Sie ist in jeder Theorie über $\mathcal{L}$ enthalten.*

b) $T_{\Sigma} = \{A \mid A \in \mathbf{Form}(\mathcal{L}),\quad \Sigma \models A\}$ *für* $\Sigma \subseteq \mathbf{Form}(\mathcal{L})$ *ist eine Theorie,* **die von Σ erzeugte Theorie** *oder durch die Axiome Σ definierte Theorie.*

c) *Ist T eine Theorie, so* $T \underset{\mathcal{F}}{\vdash} A$ gdw $A \in T$.
T inkonsistent gdw *es gibt A mit $A, \neg A \in T$, d. h.* $T = \mathbf{Form}$.

Theorien erster Stufe (Forts.)

d) *Sei $\mathcal{R}$ Relationalsystem (Struktur) für $\mathcal{L}$ $I = (D, I_c)$. Dann ist $T_{\mathcal{R}} = \{A \mid A \in \textbf{Form}(\mathcal{L}),\ \ \mathcal{R} \models A\}$ eine Theorie:*
Die Theorie von $\mathcal{R}$ *Schreibe auch: $Th(\mathcal{R})$.*

- *Verwende hier $\mathcal{R} \models A$, falls für jede Interpretation der Variablen gilt $\mathcal{R}, I_v \models A$. Insbesondere $\mathcal{R} \models A$ gdw $\mathcal{R} \models \hat{A}$, wobei $\hat{A}$ ein universeller Abschluss von A ist.*
  ($T_{\mathcal{R}} \models A$ zeige $\mathcal{R} \models A$ klar, da $\mathcal{R} \models T_{\mathcal{R}}$)

$\hookrightarrow$ *Insbesondere:*
- *$T_{\mathcal{R}}$ ist konsistent für jede Struktur $\mathcal{R}$.*

e) *$T \subseteq \textbf{Form}(\mathcal{L})$ ist Theorie* gdw
$\{A \mid A \in \textbf{Form}(\mathcal{L}), \models A\} \subseteq T$ und T ist abgeschlossen gegenüber MP.

Theorien erster Stufe (Forts.)

Definition 4.13

Sei T eine Theorie erster Stufe über $\mathcal{L}$

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

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

c) T heißt **entscheidbar**, falls T eine rekursiv entscheidbare Teilmenge von **Form** ist.

Fragen: Finde (endliche) Axiomatisierungen wichtiger Theorien. Insbesondere wann gilt $T_\mathcal{R} = T_\Sigma$ für Σ rekursiv.

Folgerungen

Bemerkung 4.14

a) *$T_{\mathcal{R}}$ ist vollständig für jede Struktur $\mathcal{R}$.*
$T_{\mathcal{R}}$ ist somit konsistent und vollständig.

b) *T erfüllbar (T hat eine Modell)* gdw *T konsistent.*

c) *T ist rekursiv axiomatisierbar, so T rekursiv aufzählbar.*

d) *Ist T vollständig, konsistent und rekursiv axiomatisierbar.*
Dann ist die Menge der Aussagen von T rekursiv entscheidbar.

•A abgeschlossen, so A oder $\neg A$ in T.
Da T rekursiv aufzählbar, findet man A oder $\neg A$ in dieser Aufzählung effektiv.

Folgerungen (Forts.)

e) *Ist T vollständig und konsistent, dann gilt $T = T_{\mathcal{R}}$ für eine Struktur $\mathcal{R}$.*
$\bullet \mathcal{R} \models T$ existiert, da T erfüllbar, d. h. $T \subseteq T_{\mathcal{R}}$.
Angenommen $T \subsetneq T_{\mathcal{R}}$. Dann gibt es abgeschlossene Formel $A \in T_{\mathcal{R}}$ mit $A \notin T$. Da T vollständig ist, muss $\neg A \in T$ gelten, d. h. $A, \neg A \in \Gamma_{\mathcal{R}}$ $\lightning$

Frage: Wann ist T_Σ vollständig für rekursive Σ?
$\rightsquigarrow$ Entscheidbarkeit!

Beispiele

Beispiel 4.15

$Th(\mathbb{N})$ Theorie der natürlichen Zahlen.
$\mathcal{R} = \langle \mathbb{N}; 0, S, +, *, = \rangle$ natürliche Interpretation der Sprache der Arithmetik Konst. $0, S, +, *$ Funktionskonstante $n \in \mathbb{N}$,
$\tilde{n} \equiv S(S \cdots (S(0) \cdots)$ Schreibe auch: $(S^n 0)$.

Die Sätze von Gödel:

a) $Th(\mathbb{N})$ ist nicht rekursiv entscheidbar.
b) $Th(\mathbb{N})$ ist nicht rekursiv axiomatisierbar.
   Oder
c) jede rekursive Axiomenmenge ist nicht vollständig für $Th(\mathbb{N})$.

Beispiele (Forts.)

$\hookrightarrow$ Insbesondere: **Peano Axiome**

$$
\begin{array}{ll}
P_1 & \forall x \forall y\, (S(x) = S(y) \rightarrow x = y) \\
P_2 & \forall x\, S(x) \neq 0 \\
P_3 & \forall x\, x + 0 = x \\
P_4 & \forall x \forall y\, x + S(y) = S(x + y) \\
P_5 & \forall x\, x * 0 = 0 \\
P_6 & \forall x \forall y\, x * S(y) = x * y + x \\
P_7 & A_x[0] \rightarrow (\forall x\, (A \rightarrow A_x[S(x)]) \rightarrow \forall x\, A), \\
 & (A \text{ Formel mit } x \text{ als einzige frei vorkommende Variable in } A)
\end{array}
$$

Sind **keine** Axiomatisierung von $Th(\mathbb{N})$.

Beispiele (Forts.)

Beispiel 4.16 (Elementare Arithmetik (Arith. 1-Stufe))

Basis der Sprache: $(\{0, 1, +, *\}, \{<\})$ Interpretation: $I_A = (\mathbb{N}, I_c)$
natürliche Interpretation I_c

$\hookrightarrow$ $Th(I_A)$ vollständig nicht rekursiv axiomatisierbar.

Beispiele (Forts.)

Beispiel 4.17

Pressburger Arithmetik: Sprache $(\{0, 1, +\}, \{<\})$
Interpretation $I_{PA} = (\mathbb{N}, I_c)$, I_c wie gehabt.

- $Th(I_{PA})$ ist vollständig, entscheidbar und endlich axiomatisierbar.

$$Ax < \begin{cases} \forall x \neg (x < 0) \\ \forall x \forall y \; x < Sy \leftrightarrow x < y \vee x = y \\ \forall x \forall y \; x < y \vee x = y \vee y < x \end{cases}$$

Beispiele (Forts.)

Beispiel 4.18

$\mathbb{R}$ reelle Zahlen, Sprache $(\{0, 1, +, *, \cdots\}, \{<\})$
Sprache der Körpertheorie.

$Th(\mathbb{R})$ ist rekursiv entscheidbar (Quantorenelimination).
$Th(\mathbb{R})$ ist rekursiv axiomatisierbar.

Beispiele (Forts.)

Axiomatisierung:

- Axiome für Körper
- Nullstellen für Polynome mit ungeradem Grad + Ordnungsaxiome:

$$
\begin{array}{ll}
\forall x & \neg(x < x) \\
\forall x \forall y & \forall z((x < y \land y < z) \rightarrow x < z) \\
\forall x \forall y & x < y \lor y = x \lor y < x \\
\forall x \forall y \forall z & x < y \rightarrow x + z < y + z \\
\forall x \forall y & 0 < x \land 0 < y \rightarrow 0 < x * y \\
\forall x & 0 < x \rightarrow \exists y (y * y = x)
\end{array}
$$

Beispiele (Forts.)

Reell abgeschlossene Körper
$\mathbb{R}$ ist „Beispiel“ dafür, Tarski

$\hookrightarrow$ Wichtige Folgerungen: Entscheidbarkeit der Ebenen euklidische Geometrie!

Beispiel 4.19

Theorie der Ordnung mit Gleichheit $(<, =)$ ist weiteres Beispiel einer entscheidbaren Theorie.

Aufzählungsverfahren für PL-1

- Σ rekursiv, $\Sigma \subseteq$ **Form**, $T_\Sigma = \{A \mid \Sigma \models A\}$ r.a.
- $\mathcal{R}$ Struktur, $T_\mathcal{R} = \{A \mid \mathcal{R} \models A\}$ i. Allg. nicht r.a. (vollst.)

1. Deduktive Beweismethoden (T_Σ)
2. Induktive Beweismethoden ($T_\mathcal{R}$)

! Hier nur 1. Grundlage: $\Sigma \models A \text{ gdw } \{\Sigma, \neg A\}$ nicht erfüllbar.

Tableaux-Methode für Pl-1

Definition 5.1

Sprache: $\neg, \wedge, \vee, \rightarrow, \forall, \exists$ (zunächst ohne $=$)
Formeln der Sprache in Klassen einteilen:

- Atomare und negierte atomare Formeln.
- α-Formeln (wie in A-Logik)
  $A \wedge B, \neg(A \vee B), \neg(A \rightarrow B), \neg\neg A, \alpha_1, \alpha_2$ wie gehabt.
- β-Formeln (wie A-Logik) $\quad \neg(A \wedge B), (A \vee B), (A \rightarrow B)$
- γ-Formeln $\quad \forall x A, \neg\exists x A \quad (A \in \textbf{Form})$
- δ-Formeln $\quad \exists x A, \neg\forall x A \quad (A \in \textbf{Form})$

Tableaux-Methode für Pl-1 (Forts.)

- Regeln zur Tableau-Konstruktion:
  α, β Regeln zur Verarbeitung α, β Formeln.

- γ-Regeln:

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

  t beliebiger Term, Substitution erlaubt.

- δ-Regeln:

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

! y Variable (oder Konstante 0-stellig) „y neu". y kommt noch nicht in Formeln im Ast zu $A_x[y]$ frei vor und die Substitution ist erlaubt.

Tableaux-Methode für Pl-1 (Forts.)

- **Systematische Konstruktion eines Modells**:
  Falls die Sprache keine F-Symbole enthält reichen die „Bezeichner“ für Elemente des Definitionsbereichs aus. Sonst wähle als Definitionsbereich für die Interpretation eine geeignete abgeschlossene Termmenge.
- Führe soviele „Terme“ ein wie notwendig.
  Existentielle Formeln benötigen „Lösungen“, solche dürfen nicht eingeschränkt sein (anderswie verwendet worden sein).
- δ-Formeln brauchen nur einmal „erfüllt“ zu werden.
  γ-Formeln hingegen müssen für alle Objekte die eingeführt wurden garantiert werden, müssen somit immer weiterhin beachtet werden.

Tableaux-Methode für Pl-1 (Forts.)

Lemma 5.2

Sei Γ eine Menge von Formeln über einer abgeschlossenen Termmenge (Universum) $\mathcal{U}$ mit

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 $\delta[t] \in \Gamma$.*

$\hookrightarrow$ *Dann ist Γ erfüllbar.*

Tableaux-Methode für Pl-1 (Forts.)

! **Problem:** $\mathcal{U}$ kann unendlich sein. Erst recht wenn F-Symbole vorhanden!

- Sprachen mit „=“ dann Funktionskonstanten und Terme.

- ϵ-Regel

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

$\hookrightarrow$ 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. $\forall x \;\; x = x$ als Wurzel der $=$ Tableaux zulassen. ◀

- Problematische Regel, da sie zu einer Formelexplosion führt.

Tableaux-Methode für Pl-1 (Forts.)

Satz 5.3

Sei $A \in \mathbf{Form}(\mathcal{L})$, $\Sigma \subseteq \mathbf{Form}(\mathcal{L})$

a) $\models A$ gdw *es gibt ein abgeschlossenes Tableau für* $\neg A$.

b) $\Sigma \models A$ gdw *es gibt ein abgeschlossenes Tableau für* $\Sigma \cup \{\neg A\}$.

- **Systematische Tableaux-Konstruktion**:

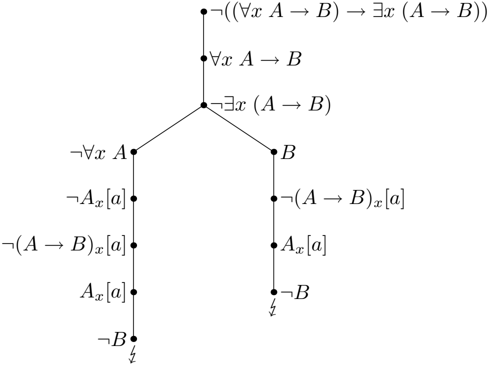
  Reihenfolge der Regelanwendungen: $\Sigma, \quad \alpha, \ \delta, \ \gamma, \ \beta$

  ! Beachte: $t \ A_x[t]$ muss erlaubt sein.
  y neu bei δ-Regel beachten (frei).

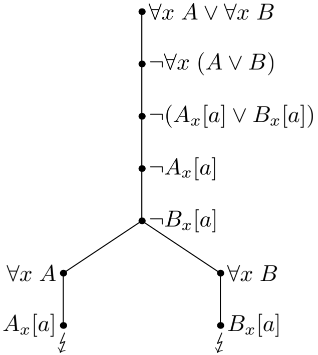
- Falls keine Funktionskonstanten in $\mathcal{L}$ Interpretation mit $\{c_1, c_2, \dots\}$ als Konstanten.

Beispiele

Beispiel 5.4 ($\models (\forall x A \rightarrow B) \rightarrow \exists x(A \rightarrow B)$, x nicht frei in B)



Beispiel 5.5 ($\Gamma = \{\forall x A \vee \forall x B\}$, $A \equiv \forall x (A \vee B)$. Gilt $\Gamma \models A$?)



! γ-Formeln dürfen nicht abgehakt werden, da man immer wieder neue Terme einführen kann und $\gamma[t]$ erfüllt werden muss!

Beispiele (Forts.)

Beispiel 5.6 ($\exists x\ \neg p(x,x) \models \neg\forall x \exists y p(x,y)$)

Gibt es Modelle für $\{\exists x\ \neg p(x,x), \forall x \exists y\ p(x,y)\}$

$$\exists x\ \neg p(x,x)$$
$$\forall x \exists y\ p(x,y)$$
$$\neg p(a,a)$$
$$\exists y\ p(a,y)$$
$$p(a,b)$$
$$\exists y\ p(b,y)$$
$$p(b,c)$$
$$\vdots$$

$\hookrightarrow$ Interpretation:

$p(x,y)$	a	b	c	d	e	$\cdots$
a	0	1				
b			1			
c				1		
d					1	
e						1
						$\ddots$

- Es gibt Interpretation mit nur 2 Elementen $\neg p(a,a), p(a,b), p(b,a), p(b,b)$, die Modell sind.

$\hookrightarrow$ Vorschrift „neu“ kann aufgeweicht werden.

! Erst alle Konstanten verwenden, falls diese Wahl zu $\lightning$ führt müssen neue Konstanten einführt werden, sonst Modell gefunden.

Beispiele (Forts.)

$\exists x \; \neg p(x,x)$

$\forall x \exists y \; p(x,y)$

$\neg p(a,a)$

$\exists y \; p(a,y)$

$p(a,b)$

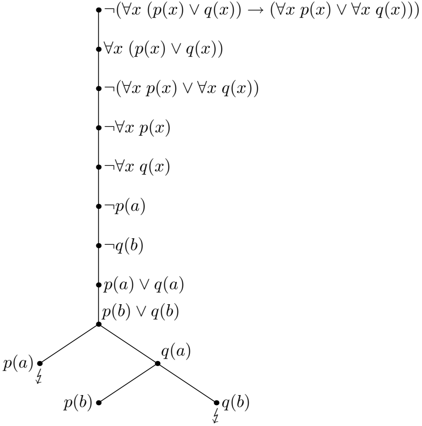
$\exists y \; p(b,y)$

$p(b,a)$

Interpretation z.B.:

	a	b
a	0	1
b	1	0

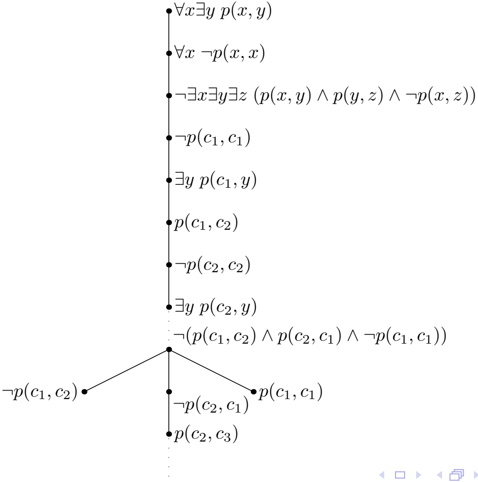
Beispiel 5.7 (Gilt $\models \forall x(p(x) \lor q(x)) \to (\forall x p(x) \lor \forall x q(x))$?)



$$I = (\{a, b\} : \quad I(p)(a) = I(q)(b) = F, \quad I(p)(b) = I(q)(a) = W)$$

Beispiel 5.8

$$\forall x \exists y\ p(x,y), \forall x\ \neg p(x,x) \models \exists x \exists y \exists z\ (p(x,y) \wedge p(y,z) \wedge \neg p(x,z))$$



Beispiele (Forts.)

Interpretation:

$p(c_1, c_1) = F \quad p(c_1, c_2) = W \quad p(c_2, c_2) = F$
$p(c_2, c_1) = F \quad p(c_2, c_3) = W \quad p(c_3, c_3) = F \ldots$

	c_1	c_2	c_3	c_4	c_5
c_1	0	1			
c_2	0	0	1		
c_3			0	1	
c_4				0	1
$\vdots$					

Beispiele Formeln mit $=$: Monoide und Gruppen

Beispiel 5.9

$\{\forall x \forall y \forall z\ (x \cdot y) \cdot z = x \cdot (y \cdot z)$
$\forall x\ x \cdot 1 = x$
$\forall x\ x \cdot \bar{x} = 1\}$

Gruppenaxiome

Behauptung: Es gilt
$\models \forall x\ 1 \cdot x = x$ und $\models \forall x\ \bar{x} \cdot x = 1$

- $\forall x\ x = x$
- Axiome
- $\neg\forall x\ 1 \circ x = x$
- $\neg\forall x\ \overline{x} \circ x = 1$
- $\neg\overline{b} \circ b = 1$
- $b \circ \overline{b} = 1$
- $(\overline{b} \circ b) \circ 1 = \overline{b} \circ b)$
- $\overline{b} \circ \overline{\overline{b}} = 1$
- $(\overline{b} \circ b) \circ (\overline{b} \circ \overline{\overline{b}}) = \overline{b} \circ b$
- $(\overline{b} \circ b) \circ (\overline{b} \circ \overline{\overline{b}}) = ((\overline{b} \circ b) \circ \overline{b}) \circ \overline{\overline{b}}$
- $(\overline{b} \circ b) \circ \overline{b} = \overline{b} \circ (b \circ \overline{b})$
- $(\overline{b} \circ b) \circ \overline{b} = \overline{b} \circ 1$
- $\overline{b} \circ 1 = \overline{b}$
- $\overline{b} \circ (b \circ \overline{b}) = \overline{b}$
- $(\overline{b} \circ b) \circ \overline{b} = \overline{b}$
- $(\overline{b} \circ b) \circ (\overline{b} \circ \overline{\overline{b}}) = \overline{b} \circ \overline{\overline{b}}$
- $(\overline{b} \circ b) \circ (\overline{b} \circ \overline{\overline{b}}) = 1$
- $\overline{b} \circ b = 1$

Entscheidbare Fälle (Präfixe) Allgemeingültigkeit $\mathcal{L}$ Sprache ohne Funktionskonstanten abgeschlossene Formeln

1. $\forall x_1 \cdots \forall x_m \exists y_1 \cdots \exists y_m$ A — A quantorenfrei
2. $\forall x_1 \cdots \forall x_m \exists y \forall z_1 \cdots \forall z_n A$
3. $\forall x_1 \cdots \forall x_m \exists y_1 \exists y_2 \forall z_1 \cdots \forall z_n A$
4. $\exists y_1 \cdots \exists y_n \forall x_1 \cdots \forall x_m \exists z A$ — $n, m \geq 1$ n. entsch.
5. $\forall x_1 \cdots \forall x_m \exists y_1 \exists y_2 \exists y_3 \forall z_1 \cdots \forall z_n A$ — $m, n \geq 0$ n. entsch.

Beispiel 5.10 (Typ 2. Formel)

- $\neg\forall x_1 \forall x_2 \exists y \forall z\ A$
- $\neg\forall x_2 \exists y \forall z\ A_{x_1}[a]$
- $\neg\exists y \forall z\ A_{x_1,x_2}[a,b]$
- $\neg\forall z\ A_{x_1,x_2,y}[a,b,a]$
- $\neg\forall z\ A_{x_1,x_2,y}[a,b,b]$
- $A_{x_1,x_2,y,z}[a,b,a,a_1]$
- $A_{x_1,x_2,y,z}[a,b,b,a_2]$

Resolventenmethode (PL1 - Resolutionsverfahren)

- **Formeln in Klauselnormalform auf Erfüllbarkeit testen**.

Definition 5.11

Eine Formel $A \in \textbf{Form}(\mathcal{L})$ ist in **Klauselform** (KLF), wenn sie die Gestalt

$$A \equiv \forall x_1 \forall x_2 \cdots \forall x_n [C_1 \wedge C_2 \wedge \cdots \wedge C_k]$$

hat. Dabei sind die C_j Klauseln (d. h. Disjunktionen von Literalen ohne Wiederholungen) und die Variablen $x_1, \ldots, x_n$ sind alle Variablen, die in den C_j vorkommen.

- $\forall x_1, \ldots, \forall x_n$ Präfix $[C_1 \wedge \cdots \wedge C_k]$ Mantisse (Matrix).

$\hookrightarrow$ PKNF und Präfix enthält kein $\exists$ Quantor.

! **Beachte**: In Literalen kommen nun Variablen vor.
L und M sind konjugierte (komplementäre) Literale, wenn
$L = \bar{M}$
(A Atom, $\neg A \equiv \bar{A}$, $\bar{\bar{A}} \equiv A$).

- $p(x)$, $\neg p(y)$ sind nicht konjugiert oder komplementär. Nach Substitution z. B. $x \leftarrow a$, $y \leftarrow a$, $p(a), \neg p(a)$ sind konjugiert. Aber auch $x \leftarrow x$, $y \leftarrow x$, $p(x), \neg p(x)$.

- **Ziel:** Resolventenmethode verallgemeinern auf Formeln mit Variablen und Funktionen (d.h. Terme).

Skolemisierung

Lemma 5.12 (Skolem)

Jede Formel A der P-Logik erster Stufe kann effektiv in eine erfüllbarkeitsäquivalente Formel A' in Klauselform transformiert werden.

- *A ist erfüllbar* gdw *A' ist erfüllbar.*

- **Anwendung**: Nachweis der Allgemeingültigkeit

$$\models B \text{ gdw } \{\neg B\} \text{ unerfüllbar gdw } (\neg B)' \text{ unerfülllbar.}$$

Skolemisierung: Verfahren

Beweis:

A gegeben. Transformiere A wie folgt:

Schritt 1: Bilde den existentiellen Abschluss von A.
Schritt 2: Eliminiere alle überflüssigen Quantoren.
Schritt 3: Umbenennung mehrfach quantifizierter Variablen.
Schritt 4: Ersetze Operatoren, die von $\wedge, \vee, \neg$ verschieden sind. z. B. $\rightarrow$, **If** $\dots$, $\leftrightarrow$ $\dots$
Schritt 5: Schiebe $\neg$ nach innen, bis sie vor den Atomen stehen. Insbesondere $\neg\neg A$ durch A ersetzen.
$\neg\forall x A \rightsquigarrow \exists x \neg A$, $\neg(A \vee B) \rightsquigarrow (\neg A \wedge \neg B)$
$\neg(A \wedge B) \rightsquigarrow (\neg A \wedge \neg B)$

Skolemisierung: Verfahren (Fort.)

Schritt 6: Schränke Quantoren auf ihren Wirkungsbereich ein:

$$\begin{array}{lll} Qx(A * B) & \leadsto A * QxB & x \text{ nicht frei in } A \\ & \leadsto QxA * B & x \text{ nicht frei in } B \end{array}$$

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

a) $x_1 \ldots x_n$ alle unterschiedlichen freien Variablen in $\exists yB(y)$ sind, die links von $\exists yB(y)$ universell quantifiziert sind.

b) f eine „frische“ n-stellige Funktionskonstante.

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

Schritt 9: Bringe Matrix in KNF und simplifiziere.

Skolemisierung

- **Korrektheit** $A_j \equiv \cdots \exists y B(y) \rightsquigarrow A_{j+1} \equiv \cdots B_y[f(x_1, \ldots, x_n)]$
- **Behauptung:** A_j ist erfüllbar gdw A_{j+1} ist erfüllbar.
  Sei I Modell für A_j: Für jede mögliche Belegung der Variablen $x_1, \ldots, x_n$ muss es ein oder mehrere Werte für y geben, mit denen A_j wahr wird.
  I' wie I zusätzlich eine n-stellige Funktionskonstante f : $f(x_1, \ldots, x_n)$ gibt für Wertetupel für $x_1 \ldots x_n$ ein (den) y Wert als Funktionsergebnis.

Beispiele

Beispiel 5.13

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

1. Existentieller Abschluss und Elimination von überf. Quantoren:
   $$\exists x_1 \forall x\ \{p(x) \rightarrow \{\neg\forall y\ [q(x,y) \rightarrow p(f(x_1))] \wedge \forall y\ [q(x,y) \rightarrow p(x)]\}\}$$
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 $\rightarrow$:
   $$\exists x_1 \forall x\ \{\neg p(x) \vee \{\neg\forall y\ [\neg q(x,y) \vee p(f(x_1))] \wedge \forall z\ [\neg q(x,z) \vee p(x)]\}\}$$

Beispiele (Forts.)

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. Quantoren auf Geltungsbereich einschränken:
$\exists x_1 \forall x \, \{\neg p(x) \vee \{[\exists y \; q(x,y) \wedge \neg p(f(x_1))] \wedge [\forall z \, \neg q(x,z) \vee p(x)]\}\}$
6. Eliminieren der Ex.-Quantoren $\exists x_1$ und $\exists y$ (0 - stellige bzw. 1-stellige Funktionseinführung):

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

Beispiele (Forts.)

7. Quantoren nach links:

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

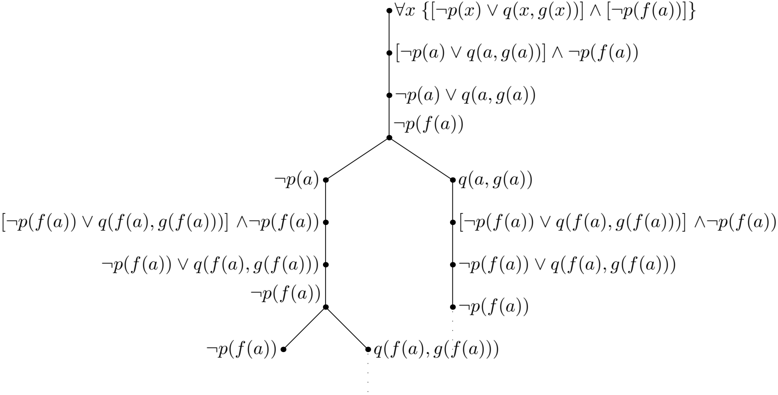
8. KNF + Simplifikation:

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

$$\rightsquigarrow A' \equiv \forall x\ \{[\neg p(x) \vee q(x, g(x))] \wedge \neg p(f(a))\}$$

- $A' \equiv \forall x[[\neg p(x) \vee q(x, g(x))] \wedge [\neg p(f(a))]]$ Erfüllbar?

Beispiele (Forts.) - Tableaux:



Terme: $a, f(a), g(a), g(f(a)), g(g(a)), g(g(f(a))), g(g(g(a)))$.

$$\mathcal{T} = \{a, g^n(a), g^n(f(a)) \quad n \geq 0\}$$ Definitionsbereich

- Erfüllbar:

$$I = (\{a\}, f(a) := a, g(a) := a, p(a) = 0, q(a, a) = 1)$$

! **Bei nicht Erfüllbarkeit**: im endlich abgeschlossenen Tableau kommen nur endlich viele Grundterme (Terme ohne Variablen) vor, d. h. eine endliche Menge von Grundinstanzen der Matrix ist unerfüllbar.

$\rightsquigarrow$ **Herbrand-Interpretationen**.

Herbrand-Universum

Definition 5.14 (Herbrand-Universum)

Sei $A \in$ **Form**. Das **Herbrand-Universum** H_A ist die Menge aller Terme, die aus den Individuenkonstanten (0-stellige Funktionskonstanten) und den Funktionskonstanten, die A vorkommen, gebildet werden können.

Enthält A keine Individuenkonstante, so wird eine hinzugenommen.

Herbrand-Universum (Forts.)

Beispiel 5.15

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

$\hookrightarrow \quad H_A = \{a, f(a), g(a), g(g(a)), g(f(a)), \cdots\}$

Die Menge der Grundterme in a, f 1-stellig, g 1-stellig.

- $B \equiv \forall x \exists y[p(f(x), y, g(x, y))]$

$\hookrightarrow H_B \equiv \{a, f(a), g(a, a), f(g(a, a)), g(a, f(a)), \ldots\}$

Das Herbrand-Universum (Forts.)

Definition 5.16 (Herbrand-Interpretation)

Eine **Herbrand-Interpretation** für eine (abgeschlossene) Formel A ist eine Interpretation $I = (H_A, \ldots)$, die

a) jeder Individuenkonstante a in A den Term $a \in H_A$ zuordnet.

b) jeder n-stelligen Funktionskonstante f, die in A vorkommt, eine Funktion $I(f) : H_A^n \to H_A$ zuordnet, die die Terme $t_1, \ldots, t_n \in H_A$ auf den Term $f(t_1, \ldots, t_n) \in H_A$ abbildet.

Der Satz von Herbrand

Satz 5.17

- *Eine Formel A in Klauselform ist erfüllbar* gdw *A ist in einer Herbrand-Interpretation erfüllbar.*
- *A ist unerfüllbar* gdw *A unerfüllbar in allen Herbrand-Interpretationen.*

▶ Festlegung der Definitionsbereiche der Interpretationen.
Sei $A \equiv \forall x_1 \cdots \forall x_n [C_1 \wedge \cdots \wedge C_k]$.
Eine **Grundinstanz** einer Klausel C_j $(1 \leq j \leq k)$ von A ist eine Klausel, die man erhält, wenn man alle Individuenvariablen in C_j durch Terme aus H_A ersetzt, d. h. eine Grundsubstitution $x_i \leftarrow t_i$ auf C_j anwendet. Solche Klauseln, die keine Individuenvariablen enthalten heißen **Grundklauseln**.

Herbrand-Prozeduren

Satz 5.18 (von Herbrand)

Eine Formel A in Klauselform ist genau dann unerfüllbar, wenn es eine endliche Konjunktion von Grundinstanzen ihrer Klauseln gibt, die unerfüllbar ist.

! Konjunktionen von Grundklauseln sind wie aussagenlogische Formeln in KNF zu behandeln.

$\rightsquigarrow$ Tableaux-Davis-Putnam-(Grund)-Resolution anwendbar.

Herbrand-Prozeduren (Forts.)

↪ **Herbrand-Prozeduren**
A Formel in KLF mit n-Klauseln und Herbrand Universum H_A.

- Erzeuge systematisch alle Grundklauseln G_j aus den Klauseln C_j $(1 \leq j \leq n)$ von A.
- Prüfe, ob die Konjunktion (aller) Grundklauseln unerfüllbar ist.

▶ **Probleme**:

- Systematische Erzeugung der Grundklauseln.
- Auswahl von Grundklauseln, die man auf Erfüllbarkeit testet.
- Welches Verfahren wählt man.

Grundresolventenmethode

- $A \equiv [\forall x \exists y\ p(x,y) \wedge \forall y \exists z\ q(y,z)] \to \forall x \exists y \exists z$ $(p(x,y) \wedge q(y,z))$
  gilt $\models A$? Transformiere $\neg A$ in KLF
- $B \equiv \forall x \forall y \forall y_1 \forall z [p(x, f(x)) \wedge q(y, g(y)) \wedge (\neg p(a, y_1) \vee \neg q(y_1, z))]$
- $H_B = \{a, f(a), g(a), f(f(a)), f(g(a)), \ldots$
- **Bilde Grundinstanzen**:
  $[p(a, f(a)) \wedge q(a, g(a)) \wedge [\neg p(a,a) \vee \neg q(a,a)]]$
  $\wedge$
  $\vdots$
  $\wedge [p(a, f(a)) \wedge q(f(a), g(f(a))) \wedge [\neg p(a, f(a)) \vee \neg q(f(a), g(f(a)))]$

Grundresolventenmethode (Forts.)

- Teste, ob erfüllbar:
  Grundresolventenmethode

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.	$\neg q(f(a), g(f(a)))$	1R3
5.	$\square$	2R4

? Wie wählt man die Klauseln und Substitutionen geschickt!

Substitutionen

1.	$p(x, f(x))$	$x \leftarrow a \quad p(a, f(a))$
2.	$q(y, g(y))$	Substitutionen
3.	$\neg p(a, y_1) \vee \neg q(y_1, z)$	$y_1 \leftarrow f(a)$
		$\neg p(a, f(a)) \vee \neg q(f(a), z)$
4.	$\neg q(f(a), z)$	1R3 $\quad x \leftarrow a \quad y_1 \leftarrow f(a)$
5.	$\square$	2R4 $\quad y \leftarrow f(a) \quad z \leftarrow g(f(a))$

Definition 5.19 (Erinnerung)

Eine Substitution θ ist eine endliche Menge der Form $\{\langle v_1, t_1\rangle, \dots, \langle v_n, t_n\rangle\}$, v_i ist Individuenvariable $v_i \neq v_j$, t_i Terme ($\mathbf{Term}(\mathbb{V}, \mathbb{F})$, $\mathbb{F}$ Funktionskonstanten $t_i \not\equiv v_i$, $i = 1, \dots, n$.
$\{v_1, \dots, v_n\}$ ist der Definitionsbereich der Substitution $\langle v_i, t_i\rangle$

- „Bindung" für v_i.

- θ induziert Abbildungen $\theta : \mathbf{Term}(\mathbb{V}, \mathbb{F}) \to \mathbf{Term}(\mathbb{V}, \mathbb{F})$
  bzw. $\theta : \mathbf{AForm} \to \mathbf{AForm}$

  Wie üblich:
  - $\theta(v_i) \equiv t_i \quad i = 1, \dots, n$
  - $\theta(v) \equiv v \quad$ sonst
  - $\theta(f(t_1, \dots, t_n)) \equiv f(\theta(t_1), \dots, \theta(t_n))$
  - $\theta(p(t_1, \dots, t_n)) \equiv p(\theta(t_1), \dots, \theta(t_n))$

Substitutionen (Forts.)

- Komposition von Substitutionen: wie üblich.
  $\theta\sigma$ erst θ, dann σ.
  Identität als Substitution erlaubt.
- Betrachte $t_1 \equiv f(g(x), y)$, $t_2 \equiv f(z, g(a))$
  Frage: Gibt es Substitution θ mit $t_1\theta = t_2\theta$ d.h.

$$\theta(f(g(x), y)) \equiv \theta(f(z, g(a)))$$

- θ heißt dann **Unifikator** von t_1 und t_2.
  $\{x \leftarrow a \quad y \leftarrow g(a) \quad z \leftarrow g(a)\}$ sind
  $\{y \leftarrow g(a) \quad z \leftarrow g(x)\}$ Unifikatoren
  $\rightsquigarrow$ **Unifikationsalgorithmen**
  Allgemeinster Unifikator: σ (MGU) (Most General Unifier).
  Eigenschaft: Ist θ Unifikator, so gibt es τ mit $\theta = \sigma\tau$.

Unifikation

Definition 5.20 (Unificator)

Sei $S = A_1 \vee \cdots \vee A_n \quad \{A_1, \ldots, A_n\}$ eine Disjunktion (Menge) atomarer Formeln A_j $(1 \leq j \leq n)$. Eine Substitution θ heißt **Unifikator** für S, falls $A_1\theta \equiv A_2\theta \equiv \cdots \equiv A_n\theta$ (insbesondere müssen die Prädikatskonstanten der A_i alle identisch sein).

- Gibt es für S einen solchen Unifikator, dann heißt S **unifizierbar**.
- Ein Unifikator θ heißt **allgemeinster Unifikator** oder **MGU** für S, wenn es für jeden Unifikator σ von S eine Substitution τ gibt, so dass $\sigma = \theta\tau$.

Unifikationsalgorithmen

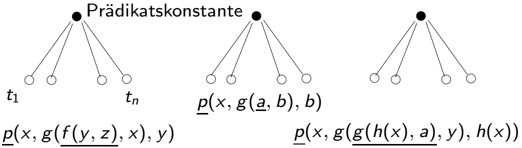
Satz 5.21
Sei $S = \{A_1, \dots, A_n\}$ Menge von atomaren Formeln, dann ist es entscheidbar ob S unifizierbar ist und ein MGU σ lässt sich berechnen.

Beweis:
Unifikationsalgorithmen für atomare Formeln in Präfix Notation.

Idee: Bestimme „Disagreement set“ durch Tiefensuche.

Unifikationsalgorithmen (Forts.)

Darstellung atomarer Formeln



$\hookrightarrow$ DS: $\{\underline{f}(y, z), \underline{a}, \underline{g}(h(x), a)\}$ nicht unifizierbar.

- DS: $\{\ldots, \underline{v}, \ldots, \underline{t}, \ldots\}$, v kommt nicht in t vor.
  Ändere: $\sigma : v \leftarrow t$.
- Berechne DSσ dann weiter bis entweder ein Unifikator bestimmt wurde oder nicht-unifizierbar als Ergebnis vorliegt.

! Es gibt sehr effiziente Unifikationsverfahren (lineare Zeit). Siehe hierfür Literatur.

Resolutionsverfahren

Definition 5.22 (Allgemeine Resolventenregel)

Seien C_1 und C_2 Klauseln ohne gemeinsame Variablen.
Seien $A_1 \vee \cdots \vee A_k$ und $\neg B_1 \vee \neg B_2 \vee \cdots \vee \neg B_l$ Teildisjunktionen von C_1 bzw. C_2, so dass $A_1, \ldots, A_k, B_1, \ldots, B_l$ unifizierbar sind, mit allgemeinsten Unifikator θ und sei

$$A_i\theta \equiv B_j\theta \equiv p(r_1, \ldots, r_n) \text{ [}\textbf{Faktor}\text{]}$$

Die **Resolvente** von C_1 und C_2 ist dann die Klausel $C_1\theta \backslash p(r_1, \ldots, r_n) \cup C_2\theta \backslash \neg p(r_1, \ldots, r_n)$ als Menge von Literalen.

Resolutionsverfahren (Forts.)

Satz 5.23 (Robinson)

Eine Formel A in Klauselform ist genau dann unerfüllbar, wenn die leere Klausel $\Box$ aus A mit der Resolventenregel hergeleitet werden kann.

- [Annahme: Klauseln in A sind variablendisjunkt] Immer erreichbar da
  $\forall\bar{x}[C_1(\bar{x}) \land \cdots \land C_k(\bar{x})] \models\!\!\!\!\!\dashv \forall\bar{x}_1\forall\bar{x}_2\cdots\forall\bar{x}_k$
  $[C_1(\bar{x}_1) \land \cdots \land C_k(\bar{x}_k)]$
- Beachte allgemeine Resolventenregel $k, l \geq 1$
- Viele Varianten: Unit-Resolution, lineare Resolventenregel....

! Ziel ist es, so schnell wie möglich die leere Klausel herzuleiten, d. h. soviel Literale wie möglich zu „faktorisieren“.

Beispiel

Beispiel 5.24

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

- Frage gilt $\models A$?
  $\neg A \equiv \neg\neg\exists y \forall z\, [p(z,y) \leftrightarrow \neg\exists x\, [p(z,x) \wedge p(x,z)]]$

$\hookrightarrow$ KLF($\neg A$)

$$\begin{array}{ll} \forall z \forall x\, [& [\neg p(z,a) \vee \neg p(z,x) \vee \neg p(x,z)] \wedge \\ & [p(z,f(z)) \vee p(z,a)] \wedge \\ & [p(f(z),z) \vee p(z,a)]] \end{array}$$

Beispiel (Forts.)

- Klauseln (nach Umbenennen der Variablen, variablendisjunkt)
  1. $\neg p(z_1, a), \neg p(z_1, x_1), \neg p(x_1, z_1)$
  2. $p(z_2, f(z_2)), p(z_2, a)$
  3. $p(f(z_3), z_3), p(z_3, a)$
- Resolvente von 2. und 1. mit Teildisjunktionen $\neg p(z_1, a), \neg p(z_1, x_1), \neg p(x_1, z_1)$ in 1. und $p(z_2, a)$ in 2. Der Unifikator $\langle x_1, a\rangle, \langle z_2, a\rangle, \langle z_1, a\rangle$ MGU.

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

Beispiel (Forts.)

- Resolvente 3. und 1. mit Teildisjunktionen
  $\neg p(z_1, a), \neg p(z_1, x_1), \neg p(x_1, z_1),\ p(z_3, a)$

- DS:

$$\left.\begin{array}{lll} \{z_1, z_1, x_1, z_3\} & x_1 \leftarrow z_1 & z_3 \leftarrow z_1 \\ \{a, z_1\} & z_1 \leftarrow a & \end{array}\right\} \begin{array}{ll} x_1 \leftarrow a & z_1 \leftarrow a \\ z_3 \leftarrow a & \end{array}$$

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

- Resolvente 4., 1. mit Teildisjunktionen
  $p(a, f(a)), \neg p(x_1, z_1) \qquad x_1 \leftarrow a, z_1 \leftarrow f(a)$

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

Beispiel (Forts.)

- Resolvente 5., 6.
  - 7. $\square$

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

Logisches Programmieren und Prolog

- Sprache der Logik zur Darstellung von Wissen (deklarativ) über Strukturen. Wissensherleitung durch logische Folgerung (deduktiv, Resolution usw.)
- **Prozedurale Sicht von Resolution**

$Z = ((Y \cdot 2) + X) - Y$ Funktion von X, Y

$2 \equiv \mathrm{succ}(\mathrm{succ}(0))$

- 0 $\mathrm{goal}(X, Y, Z) :- \mathrm{mult}(Y, 2, A), \mathrm{add}(A, X, B), \mathrm{subt}(B, Y, Z).$
- 1 $\mathrm{add}(R, \mathrm{succ}(S), \mathrm{succ}(T)) :- \mathrm{add}(R, S, T).$
- 2 $\mathrm{add}(T, 0, T).$
- 3 $\mathrm{subt}(\mathrm{succ}(R), \mathrm{succ}(S), T) :- \mathrm{subt}(R, S, T).$
- 4 $\mathrm{subt}(T, 0, T).$
- 5 $\mathrm{mult}(R, \mathrm{succ}(S), T) :- \mathrm{mult}(R, S, U), \mathrm{add}(R, U, T).$
- 6 $\mathrm{mult}(R, 0, 0).$

Logisches Programmieren und Prolog

- $\mathrm{goal}, \mathrm{add}, \mathrm{subt}, \mathrm{mult}$ sind 3-stellige P-Konstanten zur Darstellung der Funktionen, die den ersten beiden Argumenten als Wert 3-Argumente zuordnen.
- Wie „berechnet“ sich $((Y \cdot 2) + X) - Y$, wenn $X \leftarrow \mathrm{succ}(\mathrm{succ}(0))$ und $Y \leftarrow \mathrm{succ}(0)$
  $? -\mathrm{goal}(\mathrm{succ}(\mathrm{succ}(0)), \mathrm{succ}(0), Z)$

? $\lightning$

. With $Z = \mathrm{succ}(\mathrm{succ}(\mathrm{succ}(0)))$

Horn-Logik

Erinnerung: Klauseln $\{L_1, L_2, \ldots, L_n\}$ Menge von Literalen $= \{A_1, \ldots, A_n\} \cup \{\neg B_1, \ldots, \neg B_m\}$, A_j, B_j Atome.

- **Hornlogik**: Klauseln mit höchstens einem positiven Literal, d. h. $n \leq 1$. Einteilung:

$\{A, \neg B_1, \ldots, \neg B_m\}$ $m > 0$	**Regel Klausel**	◀
$\{A\}$	**Fakt Klausel**	◀
$\{\neg B_1, \ldots, \neg B_m\}$ $m > 0$	**Goal Klausel**	◀
$\varnothing$	**leeres Goal**	◀

Horn-Logik

- Formeln in KLF: Endliche Mengen von Literalen.

Hornklausel	Formel von PL-1
• $\{A, \neg B_1, \ldots, \neg B_m\}$	$\forall\ (A \vee \neg B_1 \vee \cdots \vee \neg B_m)$ bzw. $(\forall\ ((B_1 \wedge \cdots \wedge B_m) \rightarrow A))$
• $\{A\}$	$\forall\ (A)$
• $\{\neg B_1, \ldots, \neg B_m\}$	$\forall\ (\neg B_1 \vee \cdots \vee \neg B_m)$ bzw. $\neg\exists\ (B_1 \wedge \cdots \wedge B_m)$
$\square$	false

$\hookrightarrow$ **Notationen**:

Formel	logisches Programm	Prolog
• $\forall\ ((B_1 \wedge \cdots \wedge B_m) \rightarrow A)$	$A \leftarrow B_1, \ldots, B_m$	$A :- B_1, \ldots, B_m.$
• $\forall\ (A)$	$A \leftarrow$	$A.$
• $\neg\exists\ (B_1 \wedge \cdots \wedge B_m)$	$\leftarrow B_1, \ldots, B_m$	$?- B_1, \ldots, B_m.$

Horn-Logik (Forts.)

- Eine Menge von Regeln und Fakten ist ein **logisches Programm** P
  d. h. die Programmformeln sind entweder Regeln oder Fakten.
- Eine Struktur $\mathcal{R}$ ist **Modell** von P, falls $\mathcal{R}$ Modell der entsprechenden Formeln in P ist.
- $P \models \varphi$ hat die übliche Bedeutung.

! **Beachte**: Sei P logisches Programm und $? - B_1, \dots, B_m$ ein nichtleeres Ziel. Dann sind äquivalent

1. $P \cup \{? - B_1, \dots, B_m\}$ hat kein Modell (unerfüllbar)
2. $P \models \exists(B_1 \wedge \cdots \wedge B_m)$

$\hookrightarrow$ Herbrand Interpretationen reichen aus: d. h. Termmengen und Funktionen sind fest durch die Formeln (Programm) definiert.

Horn-Logik (Forts.)

- Offen ist nur noch die Interpretation der P-Konstanten.

$$\mathcal{R} \leftrightarrow I(\mathcal{R}) = \{r(t_1, \dots, t_n) \mid \begin{array}{l} r \ \ P\text{-Konstante, } n\text{-stellig} \\ t_1, \dots, t_n \in H_P \text{ Grundterme} \\ (t_1, \dots, t_n) \in r_{\mathcal{R}}\} \end{array}$$

Semantik logischer Programme (deklarative Semantik).
Sei P ein logisches Programm. Unter den Herbrand Interpretationen die Modelle von P sind, gibt es ein minimales Herbrand Modell M_P:

- $$M_P = \{r(t_1, \dots, t_n) \mid \begin{array}{l} t_1, \dots, t_n \text{ Grundterme und} \\ P \models r(t_1, \dots, t_n)\} \end{array}$$

- M_P lässt sich rekursiv definieren.

Horn-Logik (Forts.)

Satz 5.25

Sei $\exists X_1 \cdots \exists X_k (B_1 \wedge \cdots \wedge B_m)$ *eine abgeschlossene existentielle Formel und* P *logisches Programm. Dann sind äquivalent:*

1. $P \models \exists X_1 \cdots \exists X_k (B_1 \wedge \cdots \wedge B_m)$
2. $P \models (B_1 \wedge \cdots \wedge B_m)[X_1/t_1, \ldots, X_k/t_k]$ *für Grundterme* $t_1, \ldots, t_k$
3. M_p *ist Modell von* $\exists X_1 \cdots \exists X_k (B_1 \wedge \cdots \wedge B_m)$
4. $M_p \models (B_1 \wedge \cdots \wedge B_m)[X_1/t_1, \ldots, X_k/t_k]$ *für Grundterme* $t_1, \ldots, t_k$

! Grundlage für M_p ist die Semantik (Bedeutung) von P.
(Beachte der Satz gilt nicht für universelle Formeln!)

Horn-Logik (Forts.)

Beispiel 5.26

P über Signatur $0, \mathrm{succ}$ (Fkt. Symbole), add (3 St. Pr.-Konstante)

$$
\begin{aligned}
P: \quad & \mathrm{add}(X, 0, X). \\
& \mathrm{add}(X, \mathrm{succ}(Y), \mathrm{succ}(Z)) :- \mathrm{add}(X, Y, Z).
\end{aligned}
$$

$\hookrightarrow$ Offenbar ist

$M_p = \{\mathrm{add}(\mathrm{succ}^n(0), \mathrm{succ}^m(0), \mathrm{succ}^{n+m}(0)) \;\; n, m \in \mathbb{N}\}$

Wie werden existentielle Anfragen beantwortet?

- (Frage 1) $? -\mathrm{add}(\mathrm{succ}^3(0), \mathrm{succ}^8(0), Z)$
  JA mit $Z = \mathrm{succ}^{11}(0)$

- (Frage 2) $? -\mathrm{add}(X, \mathrm{succ}^8(0), Z)$
  $(P \overset{?}{\models} \exists X \exists Z\ \mathrm{add}(X, \mathrm{succ}^8(0), Z))$
  $\left.\begin{array}{ll} \text{JA} & \text{mit } X = 0, Z = \mathrm{succ}^8(0) \\ & \text{mit } X = \mathrm{succ}(0), Z = \mathrm{succ}^9(0) \ldots \end{array}\right\} Z = \mathrm{succ}^8(X)$
  ist allgemeinste Lösung.

Wie werden existentielle Anfragen beantwortet? (Forts.)

- (Frage 3) ? $-\mathrm{add}(\mathrm{succ}^3(0), Y, Z)$
  $(P \overset{?}{\models} \exists Y \exists Z\ \mathrm{add}(\mathrm{succ}^3(0), Y, Z))$

  JA mit $Y = 0, Z = \mathrm{succ}^3(0)$
  mit $Y = \mathrm{succ}(0), Z = \mathrm{succ}^4(0)$
  mit $Y = \mathrm{succ}^2(0), Z = \mathrm{succ}^5(0) \dots$

  - $Z = \mathrm{succ}^3(Y)$ ist jedoch keine allgemeinste Lösung:
  - Da $\forall Y\ \ \mathrm{add}(\mathrm{succ}^3(0), Y, \mathrm{succ}^3(Y))$ nicht logische Folgerung von P ist (Übung!)
    (Sie ist jedoch in M_p gültig!! Induktives Theorem)

Wie werden existentielle Anfragen beanwortet? (Forts.)

- (Frage 4) ? $-\mathrm{add}(X, Y, \mathrm{succ}^3(0))$
  $(P \overset{?}{\models} \exists X \exists Y \mathrm{add}(X, Y, \mathrm{succ}^3(0))$

Lösungssubstitutionen:

$$\begin{array}{ll} X = 0 & Y = \mathrm{succ}^3(0) \\ \mathrm{succ}(0) & \mathrm{succ}^2(0) \\ \mathrm{succ}^2(0) & \mathrm{succ}(0) \\ \mathrm{succ}^3(0) & 0 \end{array}$$

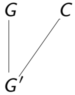
Lösungssubstitutionen

Sei $G = ?-B_1, \ldots, B_m$ ein goal, P logisches Programm, σ Substitution, $\sigma|_G$ Einschränkung von σ auf die Variablen, die in G vorkommen.
$\sigma = \{X_1 \leftarrow t_1, \ldots, X_n \leftarrow t_n\}$ ist eine **korrekte Lösungssubstitution** für $P \cup \{G\}$ gdw $X_1, \ldots, X_n$ kommen in G vor und $P \models \forall\ ((B_1 \wedge \cdots \wedge B_m)\sigma)$.
(Beachte: nicht äquivalent zu $M_p \models \forall\ ((B_1 \wedge \cdots \wedge B_m)\sigma)$ nur, falls variablenfrei).

- ? Wie operationalisiert man die Bestimmung von korrekten Lösungssubstitutionen ? $\leadsto$ **Operationale Semantik** **Varianten der Resolution**.

SLD-Resolution – (selective linear Resolution for definitive clauses)

- Sei G goal $?-A_1,\dots,A_m$ und $C \equiv A :- B_1,\dots,B_q$ eine Programmformel ($q=0$ erlaubt). Seien weiterhin G und C variablendisjunkt und sei μ ein MGU von A_k und A.
  (Die Klauseln können resolviert werden).
  Das goal
- $G' \equiv ?- A_1\mu,\dots,A_{k-1}\mu, B_1\mu,\dots,B_q\mu, A_{k+1}\mu,\dots,A_k\mu$ ist eine **SLD-Resolvente** von G und C über μ.



SLD als Regel

P-Klausel, goal $\rightsquigarrow$ goal$'$

SLD-Resolution: SLD-Ableitungen

- **SLD-Ableitungen**, wie üblich definieren:
  SLD-Ableitungen sind Ableitung der Form

  $G_0, G_1, \ldots, G_n, \ldots$ Programmformeln $C_0, C_1, \ldots, C_n, \ldots$
  Substitutionen $\mu_0, \mu_1, \ldots, \mu_n, \ldots$, so dass
  G_{n+1} SLD-Resolvente von G_n und C_n über μ_n

  (Hierbei kommen die Variablen in C_{n+1} nicht in
  $G_0, G_1, \ldots, G_n, C_0, \ldots, C_n, \mu_0, \ldots, \mu_n$ vor).

Lösungssubstitution

Definition 5.27 (Lösungssubstitution)

Sei P logisches Programm, G goal.

Eine **SLD-Widerlegung** von $P \cup \{G\}$ ist eine endliche SLD-Ableitung, bis zum Ziel G_n aus G, wobei G_n leer ist.

$C_0, \ldots, C_{n-1}$ sind Varianten (Umbenennungen) von Programmformeln aus P.

$\mu = (\mu_0 \mu_1 \cdots \mu_{n-1})|_G$ ist die berechnete **Lösungssubstitution**.

Bemerkung und Beispiel

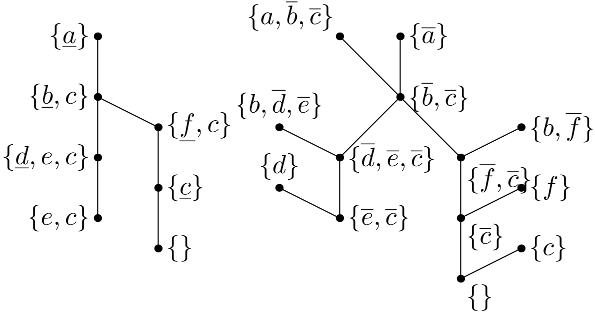
Bemerkung 5.28

- *Korrektheit und Vollständigkeit lassen sich beweisen!*
Siehe etwa Leitsch: Resolutionskalküle

Beispiel 5.29

	Klauseln
$a :- b, c.$	$\{a, \bar{b}, \bar{c}\}$
$a :- d.$	$\{a, \bar{d}\}$
$b :- d, e.$	$\{b, \bar{d}, \bar{e}\}$
$b :- f.$	$\{b, \bar{f}\}$
$c.$	$\{c\}$
$c :- d, f.$	$\{c, \bar{d}, \bar{f}\}$
$d.$	$\{d\}$
$f.$	$\{f\}$

Beispiel: Goal ? - a



Operationale Semantik von PROLOG

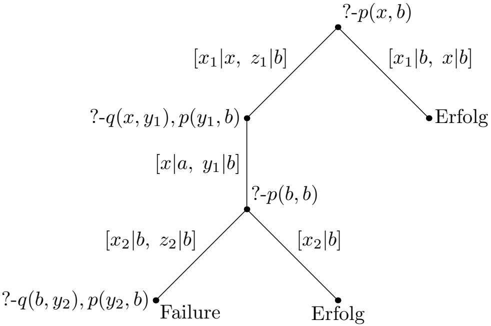
- **Prolog: Logik + Kontrolle**
- Fixiere Reihenfolge der SLD-Schritte
  - Ordne Programmformeln (Liste)
  - Goals als Listen - erstes Literal
  - Bilde stets Resolventen mit Kopf der ersten Programmformel mit erstem Literal des Goals, **normale SLD-Resolution**.
  - **Probleme**: Laufzeiten sind abhängig von den Reihenfolgen! Oft hilft ein Umordnen der Literale im Goal oder in den Programmklauseln.

Beispiele

- P sei gegeben durch:

1 $p(X,Z) :- q(X,Y), p(Y,Z).$
2 $p(X,X).$
3 $q(a,b).$

$G \equiv ?-p(X,b)$ SLD(P,G) Baum aller N-SLD Resolventen



Beispiele (Forts.)

- **Umordnung der Literale in Programmformeln**

  1 $p(X, Z) :- p(Y, Z), q(X, Y).$
  2 $p(X, X).$
  3 $q(a, b).$
  $G \equiv ?- \ p(X, b)$

$\longrightarrow$ Depth First $\rightsquigarrow$ kein Ergebnis.

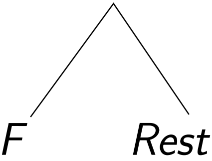
$\rightsquigarrow$ Umordnung des SLD-Baums

Beispiel 5.30 (Listen über Σ)

Gegeben Signatur Σ definiere Listen über Σ:

- 2-stellige Funktion Infix-Notation $[\cdot|\cdot] : L(\Sigma)^2 \to L(\Sigma)$
- Konstante [ ] bezeichne die leere Liste.

- **Rekursive Definition:**
  - [ ] ist Liste über Σ
  - $[F|Rest]$ ist Liste über Σ, falls F Σ-Term oder Liste über Σ
  - $Rest$ Liste über Σ



- $[F_1, F_2, \ldots, F_n|Rest] := [F_1|[F_2|\cdots|[F_n|Rest]\cdots]]$
- $[F_1, \ldots, F_n] := [F_1, \ldots, F_n|[\ ]]$

Beispiel: Listen über Σ (Forts.)

- **Operationen auf Listen**
  - $\text{append}([\ ], M, M).$
  - $\text{append}([X|L], M, [X|N]) :- \text{append}(L, M, N).$

    $?-\text{append}([a, b], [a, Y], Z)$

    Ergibt: $.Z \leftarrow [a, b, a, Y]$

$\hookrightarrow$ **Weitere Operationen**

$\text{element}(X, [X|L]).$
$\text{element}(X, [Y|L]) :- \text{unequal}(Y, X), \text{element}(X, L).$
$\text{mirror}([\ ], [\ ]).$
$\text{mirror}([X|L], M) :- \text{mirror}(L, N), \text{append}(N, [X], M).$
$\text{delete}([\ ], X, [\ ]).$
$\text{delete}([X|L], X, M) :- \text{delete}(L, X, M)$
$\text{delete}([Y|L], X, [Y|M]) :- \text{unequal}(Y, X), \text{delete}(L, X, M).$
$\text{delete1}([X|L], X, L).$
$\text{delete1}([Y|L], X, [Y|M]) :- \text{unequal}(Y, X), \text{delete1}(L, X, M).$

Schlussaufgabe:

Formalisiere und beweise folgenden Satz:

If the professor is happy if all his students like logic, then he is happy if he has no students.