

II. Prädikatenlogik

II.1. Syntax prädikatenlogischer Formeln

II.1.1. Grundsymbole

11.04.2005

Die Syntax definiert eine Menge von Grundsymbolen. Diese lassen sich weiter aufteilen:

logische Symbole

\vee	Disjunktion	(„oder“)
\wedge	Konjunktion	(„und“)
\neg	Negation	(„nicht“)
\exists	Existenz	(„es existiert“)
\forall	Allquantor	(„für all“)
\rightarrow	Implikation	(„dann“)
\leftarrow	(umgekehrte) Implikation	(„wenn“)
\leftrightarrow	Äquivalenz	(„genau dann wenn“)
$()$	Hilfssymbol	

frei wählbare, nicht-logische Symbole

- Symbole, die für ein Individuum (Objekt) stehen
 - Konstanten, z. B. a, b, c , John, 0,1,2, ...
 - Variablen, z. B. x, y, z , Name, Organisation, ...
- Funktionssymbole, z. B. f, g, h , mother_of, sin, +, -, ...
- Prädikatensymbole, z. B. p, q, r , father, =, <, ≤, ...

Die freie Wählbarkeit wird durch Konventionen eingeschränkt, zum Beispiel in **Prolog**: Variablen beginnen mit Großbuchstaben, Konstanten sowie Funktions- und Prädikatensymbole mit Kleinbuchstaben.

II.1.2. Terme

- Konstanten
- Variablen
- strukturierte Terme der Form $\varphi(\tau_1, \tau_2, \dots, \tau_n)$, wobei φ ein n -stelliges Funktionssymbol und $\tau_1, \tau_2, \dots, \tau_n$ Terme sind.

Konvention nach Schöning: f ist ein Funktionssymbol mit Stelligkeit k und τ_i ($i = 1 \dots k$) sind Terme, dann ist auch $f^k(\tau_1, \tau_2, \dots, \tau_k)$ ein Term.

II.1.3. Atome

Auch atomare Formeln oder Primformeln genannt. Atome stehen für die Beziehung zwischen Individuen. Sie haben die Form $\pi(\tau_1, \tau_2, \dots, \tau_n)$ wobei π ein n -stelliges Prädikatensymbol ist und τ_i Terme sind.

II.1.4. Formeln

Formeln erlauben eine logische Verknüpfung von Atomen und Termen. Im folgenden sind Φ und Ψ Formeln und x eine Variable.

- Atome
- $(\Phi \vee \Psi)$
- $(\Phi \wedge \Psi)$
- $(\neg \Phi)$
- $(\exists x \Phi)$ („Es gibt ein x , so dass Φ gilt“)
- $(\forall x \Phi)$ („Für alle x gilt Φ “)

Die folgenden Formeln sind nur Abkürzungen:

- $(\Phi \rightarrow \Psi)$ Abkürzung für $(\neg \Phi \vee \Psi)$. („ Φ impliziert Ψ “), („wenn Φ , dann Ψ “).
- $(\Phi \leftarrow \Psi)$ Abkürzung für $(\Phi \vee \neg \Psi)$. („ Φ wenn Ψ “), („ Φ falls Ψ “)
- $(\Phi \leftrightarrow \Psi)$ Abkürzung für $((\Phi \leftarrow \Psi) \wedge (\Phi \rightarrow \Psi))$. („ Φ genau, dann wenn Ψ “)

Ferner gelten folgende Konventionen:

- \exists und \forall binden stärker als \neg
- \neg bindet stärker als \wedge
- \wedge bindet stärker als \vee
- \vee bindet stärker als $\leftarrow, \rightarrow, \leftrightarrow$
- „unnötige“ Klammern werden weggelassen

Dies ist die Syntax der Prädikatenlogik *erster Stufe* (First Level Order). Erste Stufe bedeutet, dass Variablen nur für Individuen stehen können, nicht aber für Funktionen oder Prädikate.

Beispiele für Formeln

1. loves(romeo,juliet)
2. $\forall x$ ancestor(adam, x)
3. $\forall x$ (person(x) \rightarrow ancestor(adam, x))
4. $\forall x$ (is_mother(x) $\rightarrow \exists y$ (is_child(y) \wedge mother(x,y)))
5. \forall sub \forall super (subset(sub,super) $\leftarrow \forall x$ (member(x ,sub) \rightarrow member(x ,super)))

Definition einige weitere Begriffe.

- Die Menge der Variablen eines Terms τ , $\text{vars}(\tau)$ kann induktiv wie folgt definiert werden:
 - Für eine Konstante c gilt: $\text{vars}(c) := \{\}$
 - Für eine Variable x gilt: $\text{vars}(x) := \{x\}$
 - Für einen Term $\varphi(\tau_1, \tau_2, \dots, \tau_n)$ gilt:
 $\text{vars}(\varphi(\tau_1, \tau_2, \dots, \tau_n)) := \text{vars}(\tau_1) \cup \text{vars}(\tau_2) \cup \dots \cup \text{vars}(\tau_n)$
- Eine Variable, die nicht durch einen Quantor gebunden ist, heißt *frei* (free). Die Menge der freien Variablen einer Formel Φ , $\text{free}(\Phi)$, ist induktiv wie folgt definiert:
 - Für ein Atom $\pi(\tau_1, \tau_2, \dots, \tau_n)$ gilt:
 $\text{free}(\pi(\tau_1, \tau_2, \dots, \tau_n)) := \text{vars}(\tau_1) \cup \text{vars}(\tau_2) \cup \dots \cup \text{vars}(\tau_n)$
 - $\text{free}(\neg\Phi) := \text{free}(\Phi)$
 - $\text{free}(\Phi \vee \Psi) := \text{free}(\Phi \wedge \Psi) := \text{free}(\Phi) \cup \text{free}(\Psi)$
 - $\text{free}(\exists x\Phi) := \text{free}(\forall x\Phi) := \text{free}(\Phi) \setminus \{x\}$
- In $\forall x\Phi$ bzw. $\exists x\Phi$ heißt die Variable x durch den jeweiligen Quantor gebunden (bound).
- Eine Formel mit freien Variablen heißt offen (open).
- Eine Formel ohne freie Variablen heißt geschlossen (closed).

II.2. Semantik prädikatenlogischer Formeln

II.2.1. Definition

(exzerpt aus “Logik für Informatiker”, Seite 52 ff) Eine Struktur ist ein Paar $A(U_A, I_A)$, wobei U_A eine beliebige nicht-leere Menge ist, das *Universon* (auch Grundmenge oder Grundbereich) genannt wird. Ferner ist I_A eine Abbildung, die

- jedem k -stelligen Prädikatsymbol P ein k -stelliges Prädikat über U_A zuordnet,
- jedem k -stelligen Funktionssymbol F eine k -stellige Funktion über U_A zuordnet sowie
- jeder Variablen x ein Element der Grundmenge U_A zuordnet.

Wir schreiben abkürzend P^A statt $I_A(P)$, F_A statt $I_A(F)$ und x^A statt $I_A(x)$.

Das Herbrand-Universum ist die Menge aller aus den Objekt- und Funktionssymbolen einer Formelmengen Δ konstruierbaren Objekte.

Sei F eine Formel und $A = (U_A, I_A)$ eine Struktur. A heißt zu F passend, falls I_A für alle in F vorkommenden Symbole und freien Variablen definiert ist.

Beispiel:

$$F = \forall x P(x, f(x)) \wedge Q(g(a, z))$$

ist eine Formel. Eine zu F passende Struktur ist dann Beispielsweise:

$$\begin{aligned}
 U_A &= \{1, 2, 3, \dots\} \\
 I_A(P) &= \{(m, n) \mid m, n \in U_A \text{ und } m < n\} \\
 I_A(Q) &= \{n \in U_A \mid n \text{ ist Primzahl}\} \\
 I_A(f) &= f^A(n) = n + 1 \\
 I_A(g) &= g^A(m, n) = m + n \\
 I_A(a) &= 2 \\
 I_A(z) &= 3
 \end{aligned}$$

Beachte: x kommt gebunden vor.

II.2.2. Interpretation

Die Interpretation einer prädikatenlogischen Formel ergibt sich, wenn man jeder verwendeten Konstanten ein bekanntes Objekt eines Diskursbereichs D zuordnet und jedes verwendete Prädikatsymbol als Beziehungstyp zwischen zwei oder mehreren Objekten ansieht. Die Abbildung der Formel auf D wird als beabsichtigte Interpretation (*intended interpretation*) bezeichnet.

Beispiel: Einfacher Diskursbereich

$$D := \{\mathbb{J}, \mathbb{R}\}$$

Eine Interpretation I einer prädikatenlogischen Sprache ist ein Paar $\langle D, I \rangle$ wobei I eine Abbildung der nicht-logischen Symbole auf (nicht-leeres) D ist, wobei I

- jeder Konstanten der Sprache ein Element von D zuordnet
- jedem n -stelligen Funktionssymbol eine Abbildung $D^n \mapsto D$ zuordnet
- jedem n -stelligen Prädikatsymbol eine Abbildung $D^n \mapsto \{F, T\}$ zuordnet

Für Grundterme der Form $\varphi(\tau_1, \dots, \tau_n)$ wird die Interpretation wie folgt berechnet:

$$I(\varphi(\tau_1, \dots, \tau_n)) := I(\varphi)(I(\tau_1) \dots I(\tau_n))$$

Jeder Term τ_i wird demzufolge durch Anwendung der Abbildung I auf ein Element $I(\tau_i) \in D$ abgebildet. Auch φ wird durch I abgebildet auf $I(\varphi)$.

Für Grundatome $\pi(\tau_1, \dots, \tau_n)$ wird I wie folgt berechnet:

$$I(\pi(\tau_1, \dots, \tau_n)) := I(\pi)(I(\tau_1, \dots, \tau_n))$$

Beispiel: Interpretation von Elementen der Prädikatenlogik

Konstanten: $I(\text{juliet}) = \mathbb{J}$

$I(\text{romeo}) = \mathbb{R}$

Funktionen: $I(\text{girlfriend_of}) = \{\mathbb{R} \mapsto \mathbb{J}\}$

Prädikate: $I(\text{woman}) = \{\mathbb{J} \mapsto T, \mathbb{R} \mapsto F\}$

$I(\text{loves}) = \{\langle \mathbb{J}, \mathbb{J} \rangle \mapsto F, \langle \mathbb{J}, \mathbb{R} \rangle \mapsto T, \langle \mathbb{R}, \mathbb{J} \rangle \mapsto T, \langle \mathbb{R}, \mathbb{R} \rangle \mapsto F\}$

Grundterme: $I(\text{girlfriend_of}(\text{romeo})) = \{\mathbb{R} \mapsto \mathbb{J}\}(\mathbb{R}) = \mathbb{J}$

Grundatome: $I(\text{woman}(\text{juliet})) = \{\mathbb{J} \mapsto T, \mathbb{R} \mapsto F\}(\mathbb{J}) = T$

$I(\text{woman}(\text{romeo})) = \{\mathbb{J} \mapsto T, \mathbb{R} \mapsto F\}(\mathbb{R}) = F$

II.2.3. Definition (Fortsetzung)

(Exzerpt aus „Logik für Informatiker“, Schönig , Seite 53 ff.)

Sei F eine Formel und A eine zu F passende Struktur. Für jeden Term t , den man aus den Bestandteilen von F bilden kann, definieren wir nun den Wert von t in der Struktur A , den wir mit $A(t)$ bezeichnen.

1. Falls t eine Variable ist (also $t = x$), so ist $A(t) = x^A$.
2. Falls t die Form $t = f(t_1, \dots, t_k)$ hat, wobei t_1, \dots, t_k Terme sind und f ein k -stelliges Funktionssymbol ist, so ist $A(t) = f^A(A(t_1), \dots, A(t_k))$

Auf analoge Weise definieren wir den (Wahrheits-) Wert der Formel F wobei wir ebenfalls die Bezeichnung $A(F)$ verwenden.

1. Falls F die Form $F = P(t_1, \dots, t_k)$ mit Termen t_1, \dots, t_k , so ist

$$A(F) = \begin{cases} 1, & \text{falls } ((A(t_1), \dots, A(t_k)) \in P^A) \\ 0, & \text{sonst} \end{cases}$$

2. Falls „ $F = \neg G$ “ hat, so ist

$$A(F) = \begin{cases} 1, & \text{falls } A(G) = 0 \\ 0, & \text{sonst} \end{cases}$$

3. Falls „ $F = (G \wedge H)$ “ so ist

$$A(F) = \begin{cases} 1, & \text{falls } A(G) = 1 \text{ und } A(H) = 1 \\ 0, & \text{sonst} \end{cases}$$

4. Falls „ $F = (G \vee H)$ “ so ist

$$A(F) = \begin{cases} 1, & \text{falls } A(G) = 1 \text{ oder } A(H) = 1 \\ 0, & \text{sonst} \end{cases}$$

5. Falls „ $F = \forall x G$ “ so ist

$$A(F) = \begin{cases} 1, & \text{falls für alle } d \in U_A \text{ gilt } A_{[x/d]}(G) = 1 \\ 0, & \text{sonst} \end{cases}$$

6. Falls „ $F = \exists x G$ “ so ist

$$A(F) = \begin{cases} 1, & \text{falls es ein } d \in U_A \text{ gibt, mit } A_{[x/d]}(G) = 1 \\ 0, & \text{sonst} \end{cases}$$

Hierbei bedeute $A_{[x/d]}$ diejenige Struktur A' , die überall mit A identisch ist, bis auf die Definition von $x^{A'}$. Hier ist $x^{A'} = d$ wobei $d \in U_A = U_{A'}$

Bemerkungen

1. F ist gültig $\Leftrightarrow \neg F$ unerfüllbar ist.

Beweis: es gilt F ist gültig

\Leftrightarrow jede zu F passende Belegung ist ein Modell für F

\Leftrightarrow jede zu F und damit auch zu $\neg F$ passende Belegung ist kein Modell für $\neg F$

$\Leftrightarrow \neg F$ besitzt kein Modell

$\Leftrightarrow \neg F$ ist unerfüllbar.

2. Nicht jede mathematische Aussage kann im Rahmen der Prädikatenlogik formuliert werden. Erst wenn auch Quantoren über Prädikaten und Funktionssymbolen erlaubt werden ist dies möglich. Dies ist dann die Prädikatenlogik der zweiten Stufe. Die obige ist die Prädikatenlogik der ersten Stufe.

Beispiel Interpretation eines Atoms mit einer Variablen

Die Notation für eine Variablenbelegung (*valuation*) ist $[x \mapsto c]$ ($c \in D$).

Atom mit Variable

Für die Variablenbelegung $\mu = [x \mapsto \mathbb{J}]$ ist $I_\mu(\text{woman}(x)) = T$

Für die Variablenbelegung $\nu = [x \mapsto \mathbb{R}]$ ist $I_\nu(\text{woman}(x)) = F$

Seien Φ und Ψ Formeln, deren Interpretationen $I_\mu(\Phi)$ und $I_\mu(\Psi)$ unter der Variablenbelegung μ bekannt sind, dann folgt daraus, dass eine Interpretation für „komplexe“ Formeln existiert und induktiv wie folgt bestimmt werden kann:

$$\begin{aligned}
 I_\mu(\Phi \vee \Psi) &:= T \Leftrightarrow I_\mu(\Phi) = T \vee I_\mu(\Psi) = T \\
 I_\mu(\Phi \wedge \Psi) &:= T \Leftrightarrow I_\mu(\Phi) = T \wedge I_\mu(\Psi) = T \\
 I_\mu(\neg \Phi) &:= T \Leftrightarrow I_\mu(\Phi) = F \\
 I_\mu(\Phi \rightarrow \Psi) &:= T \Leftrightarrow I_\mu(\Phi) = F \vee I_\mu(\Psi) = T \\
 I_\mu(\Phi \leftarrow \Psi) &:= T \Leftrightarrow I_\mu(\Phi) = T \vee I_\mu(\Psi) = F \\
 I_\mu(\Phi \leftrightarrow \Psi) &:= T \Leftrightarrow (I_\mu(\Phi) = T \vee I_\mu(\Psi) = T) \wedge (I_\mu(\Phi) = F \vee I_\mu(\Psi) = F) \\
 I_\mu(\exists x \Phi) &:= T \Leftrightarrow \text{ein } c \in D \text{ existiert, so dass } I_{\mu[x \mapsto c]}(\Phi) = T \\
 I_\mu(\forall x \Phi) &:= T \Leftrightarrow \text{für alle } c \in D \text{ gilt } I_{\mu[x \mapsto c]}(\Phi) = T
 \end{aligned}$$

Bemerkung: Eine Formel kann auf verschiedene Arten interpretiert werden.

Beispiel: verschiedene Interpretationen einer Formel

Sei $\Phi = \forall x(g(x, f, t) \rightarrow l(t, x))$

Die Interpretation I sei wie folgt definiert:

- $I(f) := \text{„(Katzen-)Futter“}$
- $I(t) := \text{„Katze namens Tom“}$
- $I(g) := \text{„Die Abbildung, die einem Tripel } \langle X, Y, Z \rangle \text{ } T \text{ zuordnet, falls Objekt } X \text{ dem Objekt } Z \text{ das Objekt } Y \text{ gibt und } F \text{ sonst“}$
- $I(l) := \text{„Die Abbildung, die einem Paar } \langle X, Y \rangle \text{ } T \text{ zuordnet, falls das Objekt } X \text{ das Objekt } Y \text{ liebt und } F \text{ sonst“}$

Die Aussage „Falls X der Katze namens Tom Katzenfutter gibt, dann liebt Tom dieses X “ kann sehr wohl für jede Variablenbelegung von X aus einem bestimmten Diskursbereich als wahr betrachtet werden. Damit wäre die Formel Φ bezüglich der Interpretation I erfüllt, das heißt $I(\Phi) = T$.

A	B	$A \vee B$	$A \wedge B$	$A \rightarrow B \ (\equiv \neg A \vee B)$	$A \leftrightarrow B \ (\equiv (A \wedge B) \vee (\neg A \wedge \neg B))$
W	W	W	W	W	W
W	F	W	F	F	F
F	W	W	F	W	F
F	F	F	F	W	W

Abbildung II.1.: Wahrheitswerte der Aussagen

II.3. Normalformen

18.04.2005

II.3.1. Definition (Äquivalenzbegriff)

Zwei prädikatenlogische Formeln F und G sind äquivalent, falls für alle sowohl zu F als auch zu G passenden Strukturen A gilt:

$$A(F) = A(G)$$

Beispiel (de Morgansches Gesetz)

$$\neg(F \wedge G) \equiv \neg F \vee \neg G$$

II.3.2. Satz

Seien F und G beliebige Formeln. Dann gilt:

1.

$$\begin{aligned}\neg \forall x F &\equiv \exists x \neg F \\ \neg \exists x F &\equiv \forall x \neg F\end{aligned}$$

2. Falls x in G nicht frei vorkommt, gilt:

$$\begin{aligned}(\forall x F \wedge G) &\equiv \forall x (F \wedge G) \quad (*) \\ (\forall x F \vee G) &\equiv \forall x (F \vee G) \\ (\exists x F \wedge G) &\equiv \exists x (F \wedge G) \\ (\exists x F \vee G) &\equiv \exists x (F \vee G)\end{aligned}$$

3.

$$\begin{aligned}(\forall x F \wedge \forall x G) &\equiv \forall x (F \wedge G) \\ (\exists x F \vee \exists x G) &\equiv \exists x (F \vee G)\end{aligned}$$

4.

$$\begin{aligned}\forall x \forall y F &\equiv \forall y \forall x F \\ \exists x \exists y F &\equiv \exists y \exists x F\end{aligned}$$

Beweis (*) (exemplarisch)

Sei $A = (U_A, I_A)$ eine zu den beiden Seiten der zu beweisenden Äquivalenz passende Struktur. Es gilt:

$$\begin{aligned}
 & A(\forall x F \wedge G) = 1 \\
 \Leftrightarrow & A(\forall x F) = 1 \text{ und } A(G) = 1 \\
 \Leftrightarrow & \text{für alle } d \in U_A \text{ gilt:} \\
 & A_{[x/d]}(F) = 1 \text{ und } A(G) = 1 \\
 \Leftrightarrow & \text{für alle } d \in U_A \text{ gilt:} \\
 & A_{[x/d]}(F) = 1 \text{ und } A_{[x/d]}(G) = 1 \\
 & \text{(\textbf{Bem.:} da } x \text{ in } G \text{ nicht frei vorkommt, ist nämlich } A(G) = A_{[x/d]}(G)) \\
 \Leftrightarrow & \text{für alle } d \in U_A \text{ gilt:} \\
 & A_{[x/d]}(F \wedge G) = 1 \\
 \Leftrightarrow & A(\forall x (F \wedge G)) = 1
 \end{aligned}$$

II.3.3. Definition (NNF, KNF, DNF)

- Eine Formel ist in *Negationsnormalform*, wenn sie außer \neg, \wedge, \vee keine Junktoren enthält und wenn in ihr \neg nur vor atomaren Teilformen auftritt (NNF).
- Eine Formel ist in *konjunktiver* (bzw. *disjunktiver*) *Normalform*, wenn sie eine Konjunktion (bzw. Disjunktion) von Disjunktionen (bzw. von Konjunktionen) von Literalen ist.

II.3.4. Definition (Substitution in Formeln)

Sei F eine Formel, x eine Variable und t ein Term. Dann bezeichnet $F_{[x/t]}$ diejenige Formel, die man aus F erhält, indem man jedes freie Vorkommen der Variablen x in F durch den Term t ersetzt. $[x/t]$ beschreibt eine *Substitution*. Substitutionen (oder Folgen von Substitutionen) behandeln wir auch als selbstständige Objekte, z. B. $sub = [x/t_1][y/t_2]$ (wobei t_1 auch y enthalten darf).

II.3.5. Lemma (gebundene Umbenennung)

Sei $F = QxG$ eine Formel mit $Q \in \{\exists, \forall\}$. Es sei y eine Variable, die in G nicht vorkommt. Dann gilt:

$$F \equiv QyG_{[x/y]}$$

II.3.6. Lemma

Zu jeder Formel F gibt es eine äquivalente Formel G in *bereinigter* Form.

Hierbei heißt die Formel *bereinigt*, sofern es keine Variable gibt, die in der Formel sowohl gebunden als auch frei vorkommt, und sofern hinter allen vorkommenden Quantoren verschiedene Variablen stehen.

Beispiel

$$\begin{aligned}
 F &:= \forall x \exists y (P(x, y) \wedge Q(x, a)) \\
 G &:= \exists u (\forall v P(u, v) \vee Q(v, v))
 \end{aligned}$$

F : x, y sind gebunden, a ist frei $\rightarrow F$ ist bereinigt.

G : alle Vorkommen der Variablen u sind gebunden, allerdings tritt v sowohl gebunden als auch frei auf $\rightarrow G$ ist nicht in bereinigter Form.

II.3.7. Definition (Pränexform)

Eine Formel heißt pränex oder in Pränexform, falls sie die Form

$$Q_1 y_1 Q_2 y_2 \dots Q_n y_n F$$

hat, wobei $Q_i \in \{\exists, \forall\}$, $n \geq 0$, y_i Variablen sind. Es kommt ferner kein Quantor in F vor.

II.3.8. Satz

Für jede Formel F gibt es eine äquivalente (und bereinigte) Formel G in Pränexform.

Beweis (Induktion über den Formelaufbau)

Induktionsanfang: F ist atomare Formel. Dann liegt F bereits in der gewünschten Form vor.
Wähle also $G = F$.

Induktionsschritt: Wir betrachten wieder die verschiedenen Fälle.

1. F hat die Form $\neg F_1$ und $G_1 = Q_1 y_1 Q_2 y_2 \dots Q_n y_n G'$ sei die nach Induktionsvoraussetzung existierende zu F_1 äquivalente Formel. Dann gilt:

$$F \equiv \overline{Q}_1 y_1 \overline{Q}_2 y_2 \dots \overline{Q}_n y_n \neg G'$$

wobei $\overline{Q}_i = \exists$, falls $Q_i = \forall$ und $\overline{Q}_i = \forall$ falls $Q_i = \exists$. Diese Formel hat die gewünschte Form.

2. F hat die Form $(F_1 \circ F_2)$ mit $\circ \in \{\wedge, \vee\}$. Dann gibt es zu F_1 und F_2 äquivalente Formeln G_1 und G_2 in bereinigter Pränexform. Durch gebundenes Umbenennen können wir die gebundenen Variablen von G_1 und G_2 disjunkt machen. Dann hat:

- G_1 die Form $Q_1 y_1 Q_2 y_2 \dots Q_k y_k G'_1$
- G_2 die Form $Q'_1 z_1 Q'_2 z_2 \dots Q'_l z_l G'_2$

mit $Q_i, Q'_j \in \{\exists, \forall\}$. Daraus folgt, dass F zu

$$Q_1 y_1 Q_2 y_2 \dots Q_k y_k Q'_1 z_1 \dots Q'_l z_l (G'_1 \circ G'_2)$$

äquivalent ist.

3. F hat die Form $Qx F_1$, $Q \in \{\exists, \forall\}$. Die nach Induktionsvoraussetzung existierende bereinigte Pränexform habe die Bauart $Q_1 y_1 Q_2 y_2 \dots Q_k y_k F'_1$. Durch gebundenes Umbenennen kann die Variable x verschieden gemacht werden von y_1, \dots, y_k . Dann ist F zu $Qx Q_1 y_1 Q_2 y_2 \dots Q_k y_k F'_1$ äquivalent.

II.3.9. Definition (Skolemform)

Für jede Formel F in BPF (bereinigte Pränexform) definieren wir ihre *Skolemform(-el)* als das Resultat der Anwendung folgenden Algorithmus' auf F :

SKOLEMFORM-ALGORITHMUS

- 1 **while** F enthält einen Existenzquantor
- 2 **do** F habe die Form $F = \forall y_1 \forall y_2 \dots \forall y_n \exists z G$ für eine Formel G in BPF
und $n \geq 0$ (der Allquantorblock kann auch leer sein);
- 3 Sei f ein neues, bisher in F nicht vorkommendes n -stelliges
Funktionssymbol;
- 4 $F := \forall y_1 \forall y_2 \dots \forall y_n G_{[z/f(y_1, y_2, \dots, y_n)]}$; (d.h. der Existenzquantor in F wird
gestrichen und jedes Vorkommen der Variable z in G durch
 $f(y_1, y_2, \dots, y_n)$ ersetzt.

II.3.10. Satz

Für jede Formel F in BPF gilt: F ist erfüllbar \Leftrightarrow die Skolemform ist erfüllbar.

Beweis siehe Schöning, U.: „Logik für Informatiker“

II.3.11. Definition (Herbrand-Universum)

Das Herbrand-Universum $D(F)$ einer geschlossenen Formel F in Skolemform ist die Menge aller variablenfreien Terme, die aus den Bestandteilen von F gebildet werden können. Falls keine Konstante vorkommt, wählen wir eine beliebige Konstante, z. B. a und bilden dann die variablenfreien Terme.

$D(F)$ wird wie folgt induktiv definiert:

1. Alle in F vorkommenden Konstanten sind in $D(F)$.
Falls F keine Konstanten enthält, so ist a in $D(F)$.
2. Für jedes in F vorkommene n -stellige Funktionssymbol f und Terme t_1, t_2, \dots, t_n in $D(F)$ ist der Term $f(t_1, t_2, \dots, t_n)$ in $D(F)$.

Beispiel

$$\begin{aligned} F &= \forall x \forall y \forall z R(x, f(y), g(z, x)) \\ G &= \forall x \forall y Q(c, f(x), h(y, b)) \end{aligned}$$

Für F liegt der Spezialfall aus 1. vor (kein Vorkommen einer Konstanten).

$$\begin{aligned} D(F) &= \{a, f(a), g(a, a), f(g(a, a)), f(f(a)), \\ &\quad g(a, f(a)), g(f(a), a), g(f(a)), \dots\} \\ D(G) &= \{c, b, f(c), f(b), h(c, c), h(c, b), h(b, c), \\ &\quad h(b, b), f(f(c)), f(f(b)), f(h(c, c)), f(h(c, b)), \dots\} \end{aligned}$$

II.3.12. Definition (Herbrand-Expansion)

Sei $F = \forall y_1 \forall y_2 \dots \forall y_n F^*$ eine Aussage in Skolemform, dann ist $E(F)$ die Herbrand-Expansion von F definiert als

$$E(F) = \{F^*_{[y_1/t_1][y_2/t_2]\dots[y_n/t_n]} | t_1, t_2, \dots, t_n \in D(F)\}$$

Anmerkung, nicht im Skript: Es werden also alle gebundenen Variablen von F in der Matrix F^* von F durch beliebige Terme aus dem Herbrand-Universum $D(F)$ ersetzt.

II.3.13. Satz (Gödel-Herbrand-Skolem)

Für jede Aussage F in Skolemform gilt: F ist erfüllbar genau dann, wenn die Formelmengende $E(F)$ (im aussagenlogischen Sinn) erfüllt ist.

Beispiel

$$F = \forall x \forall y \forall z P(x, f(y, y), g(z, x))$$

Die einfachsten Formeln in $E(F)$ sind die folgenden:

$$\begin{aligned} P(a, f(a, a), g(a, a)) &\quad \text{mit} \quad [x/a], \dots, [z/a] \\ P(f(a), f(a, a), g(a, f(a))) &\quad \text{mit} \quad [x/f(a)], [y/a], [z/a] \end{aligned}$$

II.4. Prolog und Prädikatenlogik

II.4.1. Logikprogrammierung

Prolog wurde um 1970 von Alain Colmerauer und seinen Mitarbeitern mit dem Ziel entwickelt, die Programmierung von Computern mit den Mitteln „der Logik“ zu ermöglichen.

II.4.2. Pure Prolog

Das sogenannte Pure Prolog oder Database-Prolog entspricht einer Teilmenge der Sprachdefinition eines praktischen Prolog-Entwicklungssystems und enthält keine extra- oder metalogischen Komponenten wie:

- Cut, Type-Checking
- Arithmetische Operationen
- Datenbasismanipulation zur Laufzeit

II.4.3. Prolog und Logik

Pure Prolog-Programme entsprechen den Ausdrücken der Hornklausellogik, die eine Teilmenge der Prädikatenlogik 1. Stufe ist. Das Beweisverfahren Resolution ermöglicht Inferenzen aufgrund von Prologprogrammen oder Hornklauseln.

Bemerkung Zur Transformation von prädikatenlogischen Sachverhalten nach Prolog wird der Weg über die Hornklauseln genutzt.

$$\text{Prädikatenlogik 1. Stufe} \xRightarrow{\text{KNF}} \text{Hornklauseln} \implies \text{Prolog}$$

II.4.4. Prädikatenlogik 1. Stufe

Inventar der Syntax:

- Individuenkonstanten a, b, c, \dots
- Individuenvariablen x, y, z, \dots
- Prädikate $P(t_1, \dots, t_n)$, $t_i \in \text{TERM}$
- Quantoren \forall, \exists
- Junktoren $\neg, \vee, \wedge, \rightarrow, \leftrightarrow$

II.4.5. Formeln der Prädikatenlogik

- Wenn P ein n -stelliges Prädikat ist und t_1, \dots, t_n Terme sind, dann ist $P(t_1, \dots, t_n)$ ein Literal.
- Literale sind Formeln.
- Wenn Φ und Ψ Formeln sind, dann sind auch $\neg\Phi$, $\Phi \vee \Psi$, $\Phi \wedge \Psi$, $\Phi \rightarrow \Psi$, $\Phi \leftrightarrow \Psi$ Formeln.
- Wenn Φ eine Formel ist und x eine Variable, dann sind auch $\forall x\Phi$, $\exists x\Phi$ Formeln.

II.4.6. Klauseln

- Wenn P ein n -stelliges Prädikat ist und t_1, \dots, t_n Terme sind, dann ist $P(t_1, \dots, t_n)$ ein Literal.
- Literale sind Klauseln.
- Wenn Φ ein Literal ist, dann sind auch $\neg\Phi$, eine Klausel.
- Wenn Φ und Ψ Klauseln sind, dann ist auch $\Phi \vee \Psi$ eine Klausel.

II.4.7. Hornklauseln

Hornklauseln sind Klauseln, die genau ein nicht-negiertes Literal und beliebig viele negierte Literale enthalten.

Beispiele

- $\text{vater}(x, y) \vee \neg\text{elternteil}(x, y) \vee \neg\text{männlich}(x)$
- $\text{sterblich}(\text{sokrates})$

Anmerkung, nicht in der Vorlesung: Hornklauseln lassen sich als Implikationen darstellen. Die beiden Beispielen sind zu den folgenden äquivalent:

- $(\text{elternteil}(x, y) \wedge \text{Männlich}(x)) \Rightarrow \text{vater}(x, y)$
- $\text{wahr} \Rightarrow \text{sterblich}(\text{sokrates})$

II.4.8. Konjunktive Normalform

Eine Formel ist in konjunktiver Normalform, wenn sie eine Konjunktion von Klauseln repräsentiert.

$$K_1 \wedge \dots \wedge K_n, K_i \in \text{KLAUSEL}$$

Formeln der Prädikatenlogik können durch Anwendung logischer Äquivalenzregeln in die konjunktive Normalform gebracht werden.

II.4.9. Logische Äquivalenzregeln

Kommutativität

$$\begin{aligned} P \wedge Q &\Leftrightarrow Q \wedge P \\ P \vee Q &\Leftrightarrow Q \vee P \end{aligned}$$

Assoziativität

$$\begin{aligned} (P \wedge Q) \wedge R &\Leftrightarrow P \wedge (Q \wedge R) \\ (P \vee Q) \vee R &\Leftrightarrow P \vee (Q \vee R) \end{aligned}$$

De Morgan

$$\begin{aligned} \neg(P \vee Q) &\Leftrightarrow \neg P \wedge \neg Q \\ \neg(P \wedge Q) &\Leftrightarrow \neg P \vee \neg Q \end{aligned}$$

Konditional- & Bikonditionalgesetz

$$P \rightarrow Q \Leftrightarrow \neg P \vee Q$$

$$P \leftrightarrow Q \Leftrightarrow (P \rightarrow Q) \wedge (Q \rightarrow P)$$

Idempotenz

$$P \vee P \Leftrightarrow P$$

$$P \wedge P \Leftrightarrow P$$

Identität

$$P \vee 0 \Leftrightarrow P$$

$$P \vee 1 \Leftrightarrow 1$$

$$P \wedge 0 \Leftrightarrow 0$$

$$P \wedge 1 \Leftrightarrow P$$

Komplementarität

$$P \vee \neg P \Leftrightarrow 1 \text{ (Tautologie, allgemeingültig)}$$

$$P \wedge \neg P \Leftrightarrow 0 \text{ (Kontradiktion, Inkonsistenz)}$$

$$\neg \neg P \Leftrightarrow P \text{ (Doppelte Negation)}$$

II.4.10. Quantorengesetze**Quantoren-Negation**

$$\neg \forall x \Phi \Leftrightarrow \exists x \neg \Phi$$

$$\forall x \Phi \Leftrightarrow \neg \exists x \neg \Phi$$

$$\neg \forall x \neg \Phi \Leftrightarrow \exists x \Phi$$

$$\forall x \neg \Phi \Leftrightarrow \neg \exists x \Phi$$

Quantoren-Distribution

$$\forall x (\Phi \wedge \Psi) \Leftrightarrow \forall x \Phi \wedge \forall x \Psi$$

$$\exists x (\Phi \vee \Psi) \Leftrightarrow \exists x \Phi \vee \exists x \Psi$$

$$\forall x \Phi \vee \forall x \Psi \Rightarrow \forall x (\Phi \vee \Psi)$$

$$\exists x (\Phi \wedge \Psi) \Rightarrow \exists x \Phi \wedge \exists x \Psi$$

Quantoren-Dependenz

$$\forall x \forall y \Phi \Leftrightarrow \forall y \forall x \Phi$$

$$\exists x \exists y \Phi \Leftrightarrow \exists y \exists x \Phi$$

$$\exists x \forall y \Phi \Rightarrow \forall y \exists x \Phi$$

Quantoren-Bewegung

$$\begin{aligned}\Phi \rightarrow \forall x \Psi &\Leftrightarrow \forall x (\Phi \rightarrow \Psi) \\ \Phi \rightarrow \exists x \Psi &\Leftrightarrow \exists x (\Phi \rightarrow \Psi) \\ (\forall x \Phi) \rightarrow \Psi &\Leftrightarrow \exists x (\Phi \rightarrow \Psi) \\ (\exists x \Phi) \rightarrow \Psi &\Leftrightarrow \forall x (\Phi \rightarrow \Psi)\end{aligned}$$

II.4.11. Pränex-Normalform

Eine prädikatenlogische Formel befindet sich in PNF, wenn alle Quantoren am Anfang der Formel stehen.

$$\begin{aligned}(\exists x F(x)) \rightarrow (\forall y G(y)) \\ \Downarrow \\ \forall y \forall x (F(x) \rightarrow G(y))\end{aligned}$$

II.4.12. Skolemisierung

Existenzquantoren können eliminiert werden, indem existenzquantifizierte Variablen durch Skolemkonstanten substituiert werden. Liegt ein Existenzquantor im Skopus von Allquantoren, werden die Skolemkonstanten mit den jeweiligen allquantifizierten Variablen durch Parametrisierung in Abhängigkeit gebracht.

Beispiele

- $\exists y \forall x ((\text{man}(x) \wedge (\text{woman}(y)) \rightarrow \text{loves}(x, y))$
 $\implies \forall x ((\text{man}(x) \wedge \text{woman}(G)) \rightarrow \text{loves}(x, G))$
- $\forall x (\text{man}(x) \rightarrow \exists y (\text{woman}(y) \wedge \text{loves}(x, y)))$
 $\implies \forall x (\text{man}(x) \rightarrow (\text{woman}(G(x)) \wedge \text{loves}(x, G(x))))$

II.5. Aussagenlogische Resolution

Sie dient zum Nachweis der Unerfüllbarkeit einer Formel. Viele Aufgaben können auf einen Test auf Unerfüllbarkeit reduziert werden:

- F ist Tautologie gdw. $\neg F$ ist unerfüllbar
- Folgt G aus F_1, F_2, \dots, F_n ?
d.h.: Ist $F_1 \wedge F_2 \wedge \dots \wedge F_n \rightarrow G$ eine Tautologie?
d.h.: Ist $F_1 \wedge F_2 \wedge \dots \wedge F_n \wedge \neg G$ unerfüllbar?

Die aussagenlogische Resolution ist eine Umformungsregel und geht von der *Klauselschreibweise* der KNF Formel aus. Die Formel F liegt in KNF vor, wenn gilt:

$$F = (L_{1,1} \vee \dots \vee L_{1,n}) \wedge \dots \wedge (L_{m,1} \vee \dots \vee L_{m,n})$$

wobei $L_{i,j}$ Literale sind, also

$$L_{i,j} \in \{A_1, A_2, \dots\} \cup \{\neg A_1, \neg A_2, \dots\}$$

Die Klauselschreibweise von F ist dann:

$$F = \{\{L_{1,1}, \dots, L_{1,n}\}, \dots, \{(L_{m,1}, \dots, L_{m,n})\}\}$$

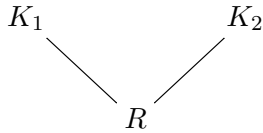
II.5.1. Definition (Resolvent)

- Es seien K_1 und K_2 Klauseln mit $A \in K_1$ und $\neg A \in K_2$, wobei A eine atomare Formel ist. Dann heißt

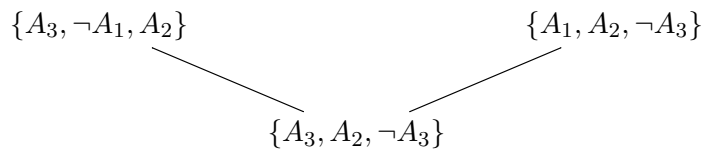
$$R = (K_1 \setminus \{A\}) \cup (K_2 \setminus \{\neg A\})$$

der *Resolvent* von K_1 und K_2 (bezüglich A).

- Diagramm:



- Falls $K_1 = \{A\}$ und $K_2 = \{\neg A\}$, dann ist $R = \emptyset$
- Eine Klauselmeng, die \emptyset enthält, ist unerfüllbar.
- Beispiel:

**II.5.2. Definition (Res(F))**

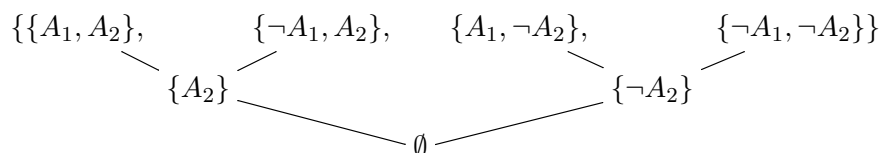
Für eine Klauselmeng F sei

$$\text{Res}(F) = F \cup \{R \mid R \text{ ist Resolvent zweier Klauseln in } F\}.$$

Weiter sei

- $\text{Res}^0(F) = F$
- $\text{Res}^{n+1}(F) = \text{Res}(\text{Res}^n(F))$
- $\text{Res}^*(F) = \bigcup_{n \geq 0} \text{Res}^n$

Resolutionssatz der Aussagenlogik: Eine Klauselmeng F ist genau dann unerfüllbar, wenn $\emptyset \in \text{Res}^*(F)$



Beispiel Die Formel ist unerfüllbar!

II.5.3. Resolution in der Prädikatenlogik

Es gibt zwei Varianten

1. Zurückführen auf die aussagenlogische Resolution (Grundresolution)
2. Direkte Resolution (Unifikation)

Grundresolution

- Stelle die Matrix der Formel F in KNF dar. Dann sind die Formeln in $E(F)$ ebenfalls in KNF.
- Da die Formeln in $E(F)$ variablenfrei sind kann die Resolution der Aussagenlogik verwendet werden.

Bemerkung Dieser Algorithmus kann nicht terminieren, falls die Formel erfüllbar ist!

Beispiel

$$F = \forall x(P(x) \wedge \neg P(f(x)))$$

Matrix von F ist schon in KNF:

$$F^* = P(x) \wedge P(f(x))$$

als Klauselmenge:

$$\begin{aligned} F^* &= \{\{P(x)\}, \{\neg P(f(x))\}\} \\ D(F) &= \{a, f(a), f(f(a)), \dots\} \\ E(F) &= \{P(a) \wedge P(f(a)), P(f(a)) \wedge P(f(f(a))), \dots\} \end{aligned}$$

als Klauselmenge

$$E(F) = \{\{P(a)\}, \{\neg P(f(a))\}, \{P(f(a))\}, \{\neg P(f(f(a)))\}, \dots\}$$

wegen

$$\begin{array}{ccc} \{\{\neg P(f(a))\}, & & \{P(f(a))\}\} \\ & \searrow \quad \swarrow & \\ & \emptyset & \end{array}$$

ist F unerfüllbar.

Verbesserung Bei diesem Verfahren werden i. A. zu viele Klauseln generiert, die für die Herleitung von \emptyset nicht gebraucht werden, da die gleiche Substitution auf die ganze Klauselmenge F^* angewendet wird.

Im vorherigen Beispiel genügt es aber für die Klausel $\{P(f(x))\}$ die Substitution $[x/f(a)]$ und für die Klausel $\{\neg P(f(x))\}$ die Substitution $[x/a]$ durchzuführen um \emptyset herleiten zu können.

II.5.4. Allgemeinsten Unifikator

- Es sei $L = \{L_1, L_2, \dots, L_k\}$ eine Menge von prädikatenlogischen Literalen und sub eine Substitution. Wir schreiben L_{sub} für die Menge $\{L_{1\text{sub}}, \dots, L_{k\text{sub}}\}$.
- sub ist ein *Unifikator* für L , falls $|L_{\text{sub}}| = 1$, dann heißt L *unifizierbar*
- Ein Unifikator sub von L heißt *allgemeinster Unifikator* von L , wenn es für jeden Unifikator sub' von L eine Substitution s gibt mit $\text{sub}' = \text{sub}s$

II.6. Prädikatenlogische Resolution

Es seien K_1, K_2 prädikatenlogische Klauseln mit

1. Es gibt Substitutionen s_1 und s_2 , so dass K_1s_1 und K_2s_2 keine gemeinsamen Variablen enthalten.
2. Es gibt Literale $L_1, \dots, L_m \in K_1s_1$ ($m \geq 1$) und $L'_1, \dots, L'_n \in K_2s_2$ ($n \geq 1$), so dass $L = \{\neg L_1, \dots, \neg L_m, L'_1, \dots, L'_n\}$ unifizierbar ist. Es sei sub ein allgemeinsten Unifikator von L .

Dann heißt die Klausel

$$R := ((K_1s_1 \setminus \{L_1, \dots, L_m\}) \cup (K_2s_2 \setminus \{L'_1, \dots, L'_n\}))sub$$

ein *prädikatenlogischer Resolvent* von K_1 und K_2 .

