Logik Grundvorlesung SS 18
Folien Teil 2: Prädikatenlogik

Gerhard Brewka

Institut für Informatik
Universität Leipzig
brewka@informatik.uni-leipzig.de

Motivation

Aussagenlogik: Struktur von Sätzen nur relevant, insofern sie durch "und", "oder", "wenn ... dann", "nicht", "genau dann ... wenn" entsteht.

Für viele logische Schlüsse zu wenig:

Alle Informatiker sind schlau.
<u>Der Vater von Peter ist ein Informatiker.</u>
Der Vater von Peter ist schlau.

In der Aussagenlogik verschiedene Sätze, repräsentiert etwa durch A, B, C. Natürlich gilt nicht: $A \wedge B \models C$

Weitere Aspekte der Struktur dieser Sätze sind zu berücksichtigen, um Zusammenhang zwischen Prämissen und Konklusion in Beispiel zu erfassen.

Motivation 2

Prädikatenlogik erfasst neben den aussagenlogischen Verknüpfungen folgende strukturelle Aspekte:

1. Sätze können aus Prädikaten (schlau), Funktionen (Vater von ...) und Objektbezeichnern (Peter) aufgebaut sein
2. Sätze können aus Quantoren (für alle, es gibt) aufgebaut sein.

Das Beispiel oben könnte etwa so repräsentiert werden:

$$\frac{\begin{array}{l}\forall x (\mathit{Informatiker}(x) \rightarrow \mathit{Schlau}(x)) \\ \mathit{Informatiker}(\mathit{Vater}(\mathit{Peter}))\end{array}}{\mathit{Schlau}(\mathit{Vater}(\mathit{Peter}))}$$

Für Definition der Formeln festzulegen, welche Variablen, welche Prädikatensymbole und welche Funktionssymbole verwendet werden.

Bei letzteren zusätzlich die Anzahl der Argumente anzugeben.

Objektbezeichner (Konstanten) als nullstellige Funktionssymbole.

Syntax: Signaturen und Terme

Definition (Signatur)

Eine (prädikatenlogische) Signatur $\Sigma = (\mathcal{V}, \mathcal{P}, \mathcal{F}, s)$ besteht aus

- einer Menge von Variablensymbolen $\mathcal{V}$,
- einer Menge von Prädikatensymbolen $\mathcal{P}$,
- einer Menge von Funktionssymbolen $\mathcal{F}$, sowie
- einer Stelligkeitsfunktion $s : \mathcal{P} \cup \mathcal{F} \rightarrow \mathbb{N}$.

Hierbei sind $\mathcal{V}$, $\mathcal{P}$ und $\mathcal{F}$ paarweise disjunkt.

Definition (Terme)

Sei $\Sigma = (\mathcal{V}, \mathcal{P}, \mathcal{F}, s)$ eine Signatur. Terme (über Σ) werden induktiv wie folgt definiert:

1. Jede Variable $x \in \mathcal{V}$ ist ein Term.
2. Falls $f \in \mathcal{F}$ mit $s(f) = k$ und $t_1, ..., t_k$ Terme, so ist $f(t_1, ..., t_k)$ Term.

Syntax: Formeln

Definition (Formeln)

Formeln (über Σ) werden induktiv wie folgt definiert:

1. Falls $P \in \mathcal{P}$ mit $s(P) = k$ und $t_1, \ldots, t_k$ Terme, so ist $P(t_1, \ldots, t_k)$ Formel.
2. Falls F Formel, so ist auch $\neg F$ Formel.
3. Falls F und G Formeln, so sind $(F \wedge G)$ und $(F \vee G)$ Formeln.
4. Falls $x \in \mathcal{V}$ und F Formel, so sind auch $\forall x F$ und $\exists x F$ Formeln.

Atomare Formel: nach 1. erzeugt.

Literal: atomare oder negierte atomare Formel.

F Teilformel von G: F Formel und kommt in G vor.

Vorkommen von $x \in \mathcal{V}$ in Teilformel von F der Form $\forall x G$ oder $\exists x G$ heißt gebunden in F. Andernfalls heißt das Vorkommen frei.

F heißt geschlossen (Aussage): keine Variable kommt in F frei vor.

Beispiele

Häufig vorkommende Zusammenhänge und ihre Darstellung:

Unterklassen:	$\forall x\,(Mann(x) \rightarrow Person(x))$
Disjunkte Klassen:	$\forall x\,(Mann(x) \rightarrow \neg Frau(x))$
alle Unterklassen:	$\forall x\,(Person(x) \rightarrow Mann(x) \vee Frau(x))$
Typrestriktionen:	$\forall x \forall y\,(Verheiratet(x, y) \rightarrow Person(x) \wedge Person(y))$
Symmetrie:	$\forall x \forall y\,(Verheiratet(x, y) \rightarrow Verheiratet(y, x))$
Definitionen:	$\forall x\,(ReicheFrau(x) \leftrightarrow Frau(x) \wedge Reich(x))$

$\rightarrow$ und $\leftrightarrow$ Abkürzungen wie bisher.

Vorrangregeln wie Aussagenlogik; zusätzlich: Quantoren binden stärker als Junktoren.

$\forall x\, P(x) \vee Q(x)$ steht für $(\forall x\, P(x)) \vee Q(x)$, nicht für $\forall x\,(P(x) \vee Q(x))$.

Semantik

Voraussetzung: Menge festlegen, über die etwas ausgesagt wird.

Funktions-/Prädikatensymbole zu Funktionen/Prädikaten auf dieser Menge interpretieren.

Definition (Struktur)

Eine Struktur ist ein Paar $A = (U_A, I_A)$. Dabei ist U_A eine nichtleere Menge (Universum, Individuenbereich), I_A eine Abbildung, für die gilt:

1. jedem k-stelligen Prädikatensymbol P im Definitionsbereich von I_A wird ein k-stelliges Prädikat über U_A zugeordnet.
2. jedem k-stelligen Funktionssymbol f im Definitionsbereich von I_A wird eine k-stellige Funktion über U_A zugeordnet.
3. jeder Variablen x im Definitionsbereich von I_A wird ein Element von U_A zugeordnet.

Wir schreiben P^A statt $I_A(P)$, f^A statt $I_A(f)$ und x^A statt $I_A(x)$.

Semantik: Terme

Definition (passende Struktur)

Sei F eine Formel, A eine Struktur. A heißt zu F passend, wenn alle in F vorkommenden Funktions- und Prädikatensymbole sowie alle freien Variablen im Definitionsbereich von I_A liegen.

Definition (Wert eines Terms)

Sei F eine Formel, A eine zu F passende Struktur. Der Wert eines Terms t in A, $A(t)$, ist induktiv wie folgt definiert:

1. Falls $t = x$ für Variable x, so ist $A(t) = x^A$.
2. Falls $t = f(t_1, ..., t_k)$, so ist $A(t) = f^A(A(t_1), ..., A(t_k))$.

Semantik: Formeln

Definition (Wahrheitswert einer Formel)

Sei F eine Formel, A eine zu F passende Struktur. Der Wahrheitswert von F unter A, $A(F)$, ist induktiv wie folgt definiert:

1. $F = P(t_1, ..., t_k)$: $A(F) = 1$ wenn $(A(t_1), ..., A(t_k)) \in P^A$, 0 sonst.
2. $F = \neg G$: $A(F) = 1$ wenn $A(G) = 0$, 0 sonst.
3. $F = (G \wedge H)$: $A(F) = 1$ wenn $A(G) = 1$ und $A(H) = 1$, 0 sonst.
4. $F = (G \vee H)$: $A(F) = 1$ wenn $A(G) = 1$ oder $A(H) = 1$, 0 sonst.
5. $F = \forall x G$: $A(F) = 1$ wenn für alle $d \in U_A$: $A_{[x/d]}(G) = 1$, 0 sonst.
6. $F = \exists x G$: $A(F) = 1$ wenn für ein $d \in U_A$: $A_{[x/d]}(G) = 1$, 0 sonst.

$A_{[x/d]}$ ist die Struktur, die mit A identisch ist bis auf den Wert von x. Es gilt $x^{A[x/d]} = d$.

Semantik: Modell/Folgerung

- Falls $A(F) = 1$, so heißt A Modell von F. Wir schreiben $A \models F$.
- F heißt erfüllbar, wenn F ein Modell besitzt.
- F heißt (allgemein)gültig, falls jede passende Struktur F zu 1 auswertet.
- Begriffe entsprechend auch für Mengen von Formeln definiert.
- Wieder gilt: F gültig gdw $\neg F$ unerfüllbar.

Folgerung und Äquivalenz direkt aus der Aussagenlogik auf die Prädikatenlogik übertragbar:

Definition

G folgt logisch aus $\{F_1, ..., F_k\}$ (Notation: $\{F_1, ..., F_k\} \models G$) gdw. jedes Modell von $\{F_1, ..., F_k\}$ auch Modell von G ist.

Beispiel

Beispiel: $F = \forall x \exists y P(x, y)$

Struktur $\mathcal{A}_1$: Universum $\{Peter, Claudia, Klaus\}$

$P(x, y)$ soll bedeuten: x liebt y. F also: Jeder liebt jemanden.
Peter liebt Claudia, Claudia liebt Klaus, Klaus liebt Peter. Also:

$P^{\mathcal{A}_1} = \{(Peter, Claudia), (Claudia, Klaus), (Klaus, Peter)\}$

$\mathcal{A}_1(F) = 1$ gdw
1. $\mathcal{A}_{1[x/Peter]}(\exists y P(x, y)) = 1$ und
2. $\mathcal{A}_{1[x/Claudia]}(\exists y P(x, y)) = 1$ und
3. $\mathcal{A}_{1[x/Klaus]}(\exists y P(x, y)) = 1$

1.: Gibt es d, so dass $\mathcal{A}_{1[x/Peter, y/d]} P(x, y)) = 1$?
Ja: $d = Claudia$, denn $(Peter, Claudia) \in P^{\mathcal{A}_1}$.

2., 3.: Ebenso für $d = Klaus$ bzw. $d = Peter$.

Damit $\mathcal{A}_1(F) = 1$. F also erfüllbar.

Beispiel: Fortsetzung

Beispiel: $F = \forall x \exists y\, P(x, y)$

Struktur A_2: Universum natürliche Zahlen

$P(x, y)$ soll bedeuten: x größer als y. F also: Jede natürliche Zahl ist größer als eine andere.

$$P^{A_2} = \{(n, m) \mid n > m\}$$

$A_2(F) = 0$ gdw. es gibt d, so dass $A_{2[x/d]}(\exists y P(x, y)) = 0$

für $d = 0$ gilt: es gibt kein d' mit (d, d') in P^{A_2}. Also

$A_{2[x/0]}(\exists y P(x, y)) = 0$ und damit $A_2(F) = 0$.

F ist also nicht allgemeingültig.

Prädikatenlogik mit Identität

Häufig verwendete Erweiterung: $=$ als 2-stelliges Prädikatensymbol. Standardinterpretation in allen Strukturen A:

$$=(t_1, t_2) \text{ bzw. } (t_1 = t_2) \text{ ist wahr gdw. } t_1^A = t_2^A.$$

Gesucht: erfüllbare Formel F, so dass für jedes Modell von F gilt: die Kardinalität des Individuenbereichs ist nicht größer als 2:

$$\forall x \forall y \forall z\, ((x = y) \vee (x = z) \vee (y = z))$$

Gesucht: Formel, die besagt

1. dass P antisymmetrische Relation ist:
   $\forall x \forall y\, (P(x, y) \to \neg P(y, x))$ oder $\forall x \forall y\, \neg(P(x, y) \wedge P(y, x))$
2. dass f injektive Funktion ist:
   $\forall x \forall y\, (f(x) = f(y) \to x = y)$
3. dass f surjektive Funktion ist:
   $\forall x \exists y\, (f(y) = x)$

Äquivalenz

Äquivalenzbegriff der Aussagenlogik direkt übertragbar:

Definition (Äquivalenz)

Die Formeln F und G sind äquivalent, falls für alle zu F und G passenden Strukturen $\mathcal{A}$ gilt: $\mathcal{A}(F) = \mathcal{A}(G)$.

Alle aussagenlogischen Äquivalenzen gelten auch in Prädikatenlogik. Weitere Äquivalenzen:

$$\begin{array}{rcl} \neg\forall x\, F & \equiv & \exists x\, \neg F \\ \neg\exists x\, F & \equiv & \forall x\, \neg F \end{array}$$

Falls x in G nicht frei vorkommt:

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

Weitere Äquivalenzen

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

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

Nicht äquivalent sind:

$(\forall x\ F \lor \forall x\ G)$ und $\forall x\ (F \lor G)$
$(\exists x\ F \land \exists x\ G)$ und $\exists x\ (F \land G)$

Gegenbeispiel: Universum $\mathbb{N}$, F ungerade, G gerade.

Beobachtung: obige Äquivalenzen können benutzt werden, um Quantoren nach vorne zu schieben.
Dabei sind möglicher Weise Variablen umzubenennen.

Exemplarischer Beweis

Wir beweisen: falls x in G nicht frei vorkommt, so gilt

$$(\forall x\, F \wedge G) \equiv \forall x\, (F \wedge G)$$

Sei A beliebige zu $(\forall x\, F \wedge G)$ und $\forall x\, (F \wedge G)$ passende Struktur.

Es gilt:

$$\begin{array}{ll}
 & A(\forall x\, F \wedge G) = 1 \\
\text{gdw.} & A(\forall x\, F) = 1 \text{ und } A(G) = 1 \\
\text{gdw.} & \text{für alle } d \in U_A \text{ gilt } A_{[x/d]}(F) = 1 \text{ und } A(G) = 1 \\
\text{gdw.} & \text{für alle } d \in U_A \text{ gilt } A_{[x/d]}(F) = 1 \text{ und } A_{[x/d]}(G) = 1 \\
 & (\text{da } x \text{ nicht frei in } G \text{ vorkommt, gilt } A(G) = A_{[x/d]}(G)) \\
\text{gdw.} & \text{für alle } d \in U_A \text{ gilt } A_{[x/d]}(F \wedge G) = 1 \\
\text{gdw.} & A(\forall x\, (F \wedge G)) = 1
\end{array}$$

Äquivalenzumformung: Beispiel

Betrachte die Formel: $(\forall x\, F \wedge \exists y\, G)$

Es gelte: x kommt in G nicht frei vor, y kommt in F nicht frei vor.

Damit gilt:

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

Aber auch:

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

Es ist vorteilhaft, Existenzquantoren möglichst weit links zu haben.

Beispiel ist Spezialfall! Offensichtlich $\exists y \forall x\, P(x, y) \not\equiv \forall x \exists y\, P(x, y)$.

Bereinigte Formeln

Definition

Sei F Formel, x Variable, t Term. $F_{[x/t]}$ bezeichnet die Formel, die man aus F erhält, indem jedes freie Vorkommen von x in F durch t ersetzt wird. Eine solche Ersetzung nennt man Substitution.

Lemma: Sei $F = Qx\, G$ eine Formel mit Quantor Q, y eine Variable, die nicht in G vorkommt. Dann gilt: $Qx\, G \equiv Qy\, G_{[x/y]}$.

Definition

F heißt bereinigt, wenn keine Variable sowohl gebunden wie frei in F auftritt, und wenn alle Quantoren verschiedene Variablen haben.

Lemma: Sei F Formel. Es gibt eine bereinigte Formel G mit $F \equiv G$.

Beispiel:	$\forall x \exists y\, P(x, f(y)) \wedge \forall y (S(x, y) \vee R(x))$
bereinigt:	$\forall u \exists v\, P(u, f(v)) \wedge \forall y (S(x, y) \vee R(x))$

Pränexform

Definition

Eine Formel F ist in Pränexform (PNF), falls

$$F = Q_1 y_1 Q_2 y_2 \ldots Q_n y_n F',$$

wobei $n \geq 0$, Q_i Quantoren und F' (die Matrix von F) quantorenfrei.

Satz: Für jede Formel F gibt es eine äquivalente Formel in PNF.

Bew.: Induktion über Formelaufbau.

Induktionsanfang: F atomar $\Rightarrow$ F in PNF

Induktionsschritt: wir zeigen, dass das Resultat für jeden möglichen Aufbau einer komplexen Formel F gilt, unter der Induktionsannahme, dass für jede echte Teilformel von F eine äquivalente Formel in PNF existiert.

Beweis PNF: Induktionsschritt

1. $F = \neg F_1$: $G_1 = Q_1 y_1 Q_2 y_2 \ldots Q_n y_n G'$ sei PNF von F_1 (existiert nach Induktionsvoraussetzung). Dann gilt $F \equiv -Q_1 y_1 - Q_2 y_2 \ldots - Q_n y_n \neg G'$, wobei $-$ den All- in Existenzquantor verwandelt und umgekehrt.

2. $F = (F_1\, op\, F_2)$ mit $op \in \{\wedge, \vee\}$: $G_1 = Q_{11} y_{11} Q_{12} y_{12} \ldots Q_{1n} y_{1n} G_1'$ und $G_2 = Q_{21} y_{21} Q_{22} y_{22} \ldots Q_{2m} y_{2m} G_2'$ seien zu F_1 bzw. F_2 äquivalente Formeln in PNF. Die gebundenen Variablen in G_1 und G_2 seien disjunkt (evtl. umbenennen). Dann gilt

$$F \equiv Q_{11} y_{11} Q_{12} y_{12} \ldots Q_{1n} y_{1n} Q_{21} y_{21} Q_{22} y_{22} \ldots Q_{2m} y_{2m} (G_1'\, op\, G_2').$$

3. $F = QxF_1$, $Q \in \{\forall, \exists\}$: $Q_1 y_1 Q_2 y_2 \ldots Q_n y_n F_1'$ sei PNF von F_1. x sei verschieden von allen y_i (evtl. umbenennen). Dann gilt

$$F \equiv Qx Q_1 y_1 Q_2 y_2 \ldots Q_n y_n F_1'.$$

Umformung in PNF

Im vorigen Beweis steckt implizit ein Algorithmus zur Umformung:

Negation nach innen (hinter Quantoren) schieben (1.)
Quantoren nach vorne schieben (2.)
Dabei gegebenenfalls Variablen umbenennen (3.)

Beispiel:

$$\neg\exists x R(x) \vee (P(a) \wedge \forall x Q(a,x))$$
$$\forall x \neg R(x) \vee (P(a) \wedge \forall x Q(a,x))$$
$$\forall x \neg R(x) \vee \forall x (P(a) \wedge Q(a,x))$$
$$\forall x \neg R(x) \vee \forall y (P(a) \wedge Q(a,y))$$
$$\forall x \forall y (\neg R(x) \vee (P(a) \wedge Q(a,y)))$$

Lemma: Sei F eine Formel, $x_1, ..., x_n$ die freien Variablen in F. Es gilt:

1. F ist allgemeingültig gdw. $\forall x_1 ... \forall x_n\, F$ allgemeingültig ist.
2. F ist erfüllbar gdw. $\exists x_1 ... \exists x_n\, F$ erfüllbar ist.

Skolemisierung

Definition (Skolemform)

Sei $F = Q_1 y_1 Q_2 y_2 ... Q_n y_n F_1$ eine Formel in bereinigter Pränexform (BPF). Die Skolemform von F entsteht aus F durch

1. Ersetzen jeder $\exists$-quantifizierten Variable x in F_1 durch Term $f(y_1, ...y_k)$. f neues, nicht in F vorkommendes Funktionssymbol, Argumente y_i die Variablen der in F links vom Quantor von x vorkommenden Allquantoren, von links nach rechts.
2. Streichen aller Existenzquantoren.

Beispiel:

$\forall x \exists y \exists z (S(y, x) \wedge P(z, x))$ neue 1-stellige Funktionssymbole s, p
$\forall x (S(s(x), x) \wedge P(p(x), x))$ Skolemform

Skolemisierung keine Äquivalenzumformung, aber:

Satz: F in BPF erfüllbar gdw. die Skolemform von F erfüllbar.

Umformungsschritte zusammengefasst

Gegeben: beliebige Formel F

1. Bereinigen von F durch Umbenennen gebundener Variablen.
2. Binden freier Variablen durch Voranstellen von Existenzquantoren (erfüllbarkeitsgleich).
3. Herstellen der PNF.
4. Herstellen der Skolemform.
5. Umformen der Matrix in KNF.
6. Weglassen der Allquantoren und Notieren in Klauselform.

Anmerkung: in (2) gebundene freie Variablen in (4) durch Skolemkonstanten ersetzt $\Rightarrow$ kann auch gleich in (2) erfolgen.

Umformung Beispiel

$\neg\exists x(P(x,z) \lor \forall y Q(x,y)) \lor \forall y P(y,z)$	
$\neg\exists x(P(x,z) \lor \forall y Q(x,y)) \lor \forall w P(w,z)$	bereinigt
$\exists z(\neg\exists x(P(x,z) \lor \forall y Q(x,y)) \lor \forall w P(w,z))$	z gebunden
$\exists z(\forall x(\neg P(x,z) \land \neg\forall y Q(x,y)) \lor \forall w P(w,z))$	Negation
$\exists z(\forall x(\neg P(x,z) \land \exists y \neg Q(x,y)) \lor \forall w P(w,z))$	Negation
$\exists z \forall x \exists y \forall w((\neg P(x,z) \land \neg Q(x,y)) \lor P(w,z))$	PNF
$\forall x \forall w((\neg P(x,s) \land \neg Q(x, sk(x))) \lor P(w,s))$	Skolemform
$(\neg P(x,s) \lor P(w,s)) \land (\neg Q(x, sk(x)) \lor P(w,s))$	Matrix in KNF
$\{\{\neg P(x,s), P(w,s)\}, \{\neg Q(x, sk(x)), P(w,s)\}\}$	Klauselform

Alle freien Variablen der Klauselform implizit allquantifiziert!

Unentscheidbarkeit

Aussagenlogik: für jede Formel endliche Menge von Belegungen.
Prädikatenlogik: Beschränkung auf endliche Menge von Strukturen nicht möglich.

Sogar Strukturen mit unendlichem Universum zu betrachten:

$$\forall x\, P(x, f(x)) \wedge \forall y\, \neg P(y, y) \wedge \forall u \forall v \forall w((P(u, v) \wedge P(v, w)) \rightarrow P(u, w))$$

nur unendliche Modelle, etwa: $A = (U_A, I_A)$ mit

- $U_A = \{0, 1, 2, ...\}$
- $P^A = \{(m, n) \mid m < n\}$
- $f^A(n) = n + 1$

Entscheidungsverfahren, die alle Strukturen der Reihe nach prüfen, also ausgeschlossen.

Unentscheidbarkeit

Definition

Ein Problem P heißt entscheidbar, wenn es ein Berechnungsverfahren gibt, das

1. für jede Eingabe terminiert,
2. für jede Eingabe die richtige JA/NEIN-Antwort liefert.

Gültigkeitsproblem der Prädikatenlogik:
gegeben: Formel F
gefragt: ist F gültige Formel?

Erfüllbarkeitsproblem der Prädikatenlogik:
gegeben: Formel F
gefragt: ist F erfüllbare Formel?

Unentscheidbarkeitsresultate

Satz (Church): Das Gültigkeitsproblem der Prädikatenlogik ist unentscheidbar.

Bew.: Reduktion eines unentscheidbaren Problems (z.B., Korrespondenzproblem, 4. Semester) auf das Gültigkeitsproblem. Idee: übersetze Problem A so in Problem B, dass Lösung von B Lösung von A beinhaltet. Dann ist es mindestens so schwer, B zu entscheiden, wie A zu entscheiden. Wenn A unentscheidbar, dann auch B.

Korollar: Das Erfüllbarkeitsproblem der Prädikatenlogik ist unentscheidbar.

Bew.: Wäre Erfüllbarkeit entscheidbar, so auch Gültigkeit, denn eine Formel ist gültig, gdw. ihre Negation nicht erfüllbar ist.

Korollar: Folgerbarkeit in der Prädikatenlogik ist unentscheidbar.

Bew.: Wäre Folgerbarkeit entscheidbar, so auch Gültigkeit, denn eine Formel ist gültig, gdw. sie aus einer Tautologie folgerbar ist.

Herbrand-Theorie

Beliebige Mengen als Universum zulässig. Suche nach Modellen kann aber kanonisch erfolgen:

Definition (Herbrand-Universum)

Das Herbrand-Universum $D(F)$ einer geschlossenen Formel F in Skolemform ist die Menge aller variablenfreien Terme, die aus in F vorkommenden Symbolen gebildet werden können. Falls in F keine Konstante vorkommt, kann neue Konstante a verwendet werden.

Induktiv:

1. c Konstante in $F \Rightarrow c \in D(F)$; keine Konstante in $F \Rightarrow a \in D(F)$.
2. n-stelliges f in F und $t_1, ..., t_n \in D(F) \Rightarrow f(t_1, ..., t_n) \in D(F)$.

$$\text{Beispiel: } \forall x(\neg P(a, f(x)) \vee Q(g(b)))$$

$$D(F) =$$

$$\{a, b, f(a), f(b), g(a), g(b), f(f(a)), f(f(b)), f(g(a)), f(g(b)), g(f(a)), ...\}$$

Herbrand-Strukturen

Definition (Herbrand-Strukturen)

Sei F geschlossene Formel in Skolemform. Eine zu F passende Struktur $A = (U_A, I_A)$ heißt Herbrand-Struktur, falls gilt:

1. $U_A = D(F)$,
2. für jedes n-stellige in F vorkommende Funktionssymbol f und $t_1, ..., t_n \in D(F)$ gilt: $f^A(t_1, ..., t_n) = f(t_1, ..., t_n)$.

Herbrand-Strukturen lassen also nur die Wahl von P^A für jedes Prädikatensymbol P offen.

Terme werden "durch sich selbst" interpretiert.

Satz: Sei F eine geschlossene Formel in Skolemform. Dann ist F erfüllbar gdw. F ein Herbrand-Modell besitzt.

Herbrand-Expansion

Definition (Instanz, Grundinstanz)

Sei A ein Ausdruck (Term oder Formel) mit den freien Variablen $x_1, ..., x_n$. Für Terme $t_1, ..., t_n$ heißt der Ausdruck $A[x_1/t_1, ..., x_n/t_n]$, in dem jedes Auftreten von x_i in A durch t_i ersetzt ist:

1. Instanz von A (die t_i können beliebige Terme sein),
2. Grundinstanz von A, wenn alle t_i variablenfrei sind.

Definition (Herbrand-Expansion)

Sei $F = \forall y_1 ... \forall y_n F'$ eine geschlossene Formel in Skolemform. Die Herbrand-Expansion von F, $E(F)$, ist wie folgt definiert:

$$E(F) = \{F'[y_1/t_1, ..., y_n/t_n] \mid t_1, ..., t_n \in D(F)\}$$

Herbrand-Expansion also die Menge aller Grundinstanzen von F'.

Semi-Entscheidbarkeit

Satz (Gödel, Herbrand, Skolem): Für jede geschlossene Formel F in Skolemform gilt: F ist erfüllbar gdw. $E(F)$ erfüllbar ist.

Daraus und mit Endlichkeitssatz der Aussagenlogik folgt:

Satz (Herbrand): Für jede geschlossene Formel F in Skolemform gilt: F ist unerfüllbar gdw. es eine endliche Teilmenge von $E(F)$ gibt, die unerfüllbar ist.

Hieraus Semi-Entscheidungsverfahren für Unerfüllbarkeit von Formeln ableitbar:

- falls F unerfüllbar Terminierung mit korrekter Antwort;
- falls F erfüllbar keine Terminierung.

Algorithmus von Gilmore

Eingabe: geschlossene Formel F in Skolemform
Sei $E(F) = \{F_1, F_2, F_3, ...\}$;
n := 0;
repeat n := n+1 **until** $\{F_1, ..., F_n\}$ unerfüllbar;
stoppe mit Ausgabe "unerfüllbar";

Satz: Folgende Probleme der Prädikatenlogik sind semi-entscheidbar:

a) das Unerfüllbarkeitsproblem,

b) das Gültigkeitsproblem,

c) das Folgerbarkeitsproblem.

Beweis:

a) siehe Algorithmus,
b) wende Algorithmus auf $\neg F$ an,
c) teste $F \wedge \neg C$ (F Prämisse, C Folgerung) auf Unerfüllbarkeit.

Grundresolutionsalgorithmus:

> Eingabe: geschlossene Formel F in Skolemform, Matrix KNF
> Sei $E(F) = \{F_1, F_2, F_3, ...\}$;
> n := 0;
> $M := \emptyset$;
> **repeat** n := n+1; $M := M \cup \{F_n\}$; $M := Res^*(M)$ **until** $\square \in M$;
> stoppe mit Ausgabe "unerfüllbar";

Aus Satz von Herbrand und aussagenlogischem Resolutionssatz folgt:

Satz: Bei Eingabe einer geschlossenen Formel in Skolemform mit Matrix F^* in KNF stoppt Grundresolutionsalgorithmus nach endlich vielen Schritten mit Ausgabe "unerfüllbar" gdw. F unerfüllbar ist.

Bemerkung: Algorithmus erzeugt wesentlich mehr Formeln, als für Beweis von $\square$ nötig.

Grundresolution

Grundresolutionssatz:

Eine Aussage in Skolemform F mit Matrix F^* in KNF ist unerfüllbar gdw. es eine Folge von Klauseln $K_1, ..., K_n$ gibt mit:

1. $K_n = \square$, und
2. für alle $i \in \{1, ..., n\}$ gilt: entweder K_i ist Grundinstanz einer Klausel aus F^*, oder K_i ist (aussagenlogische) Resolvente zweier Klauseln K_j, K_k mit $j, k < i$.

Suche nach Grundinstanzen oft ineffizient: sehr viele Instanzierungsmöglichkeiten.

Deshalb Instanzierung so zurückhaltend wie möglich anwenden: nur soweit es für Resolutionsschritt erforderlich.

Bsp.: $\{P(x), \neg Q(g(x))\}, \{\neg P(f(y))\}$
$[x/f(y)]$ genügt und ergibt: $\{\neg Q(g(f(y)))\}$. Substitution für y unnötig.

Unifikation

Definition (Substitution, Unifikator)

Eine Substitution $sub = [x_1/t_1, ..., x_k/t_k]$ ist eine Abbildung von Variablen x_i auf Terme t_i. Sei A Ausdruck (Term oder Formel). $A\,sub$ entsteht aus A, indem jedes Vorkommen von x_i durch t_i ersetzt wird.

sub heißt Unifikator einer Menge von Literalen $L = \{L_1, ..., L_n\}$, falls $L_1\,sub = ... = L_n\,sub$.

sub heißt allgemeinster Unifikator (most general unifier, mgu) von L, falls für jeden Unifikator sub' von L: $sub' = sub\,sub''$ für geeignete Substitution sub''.

Unifikationssatz (Robinson): Jede unifizierbare Menge von Literalen besitzt auch einen allgemeinsten Unifikator.

Unifikationsalgorithmus

Eingabe: nicht-leere Literalmenge L;
$sub := []$;
while $|L\,sub| > 1$ **do**
  begin
  durchsuche Literale in $L\,sub$ von links nach rechts bis zur ersten Position, an der sich zwei Literale unterscheiden;
  if keines der sich unterscheidenden Zeichen ist Variable
  then stoppe mit "fail"
  else begin
    Sei x die Variable, t der Term, der im anderen Literal beginnt
    if x kommt in t vor **then** stoppe mit "fail" (occur check)
    else $sub := sub\,[x/t]$ (Hintereinanderausführen der Substitut.)
    end;
  end;
gib mgu sub aus;

Unifikation: Beispiel

$$P(a, f(x, b))$$
$$P(y, f(g(y), z)) \qquad sub = [y/a]$$

$$P(a, f(x, b))$$
$$P(a, f(g(a), z)) \qquad sub = [y/a][x/g(a)]$$

$$P(a, f(g(a), b))$$
$$P(a, f(g(a), z)) \qquad sub = [y/a][x/g(a)][z/b]$$

$$P(a, f(g(a), b))$$
$$P(a, f(g(a), b))$$

hier: $[y/a][x/g(a)][z/b] = [y/a, x/g(a), z/b]$ (parallele Substitution)
allgemeinen: spätere Substitutionen auf vorherige angewenden:

$$[y/f(x, b)][x/g(a)] = [y/f(g(a), b), x/g(a)] \neq [y/f(x, b), x/g(a)]$$

Notwendigkeit des Occur Checks

Was würde passieren ohne Occur Check?

$$P(x, x)$$
$$P(y, f(y)) \qquad sub = [x/y]$$

$$P(y, y)$$
$$P(y, f(y)) \qquad sub = [x/y][y/f(y)]$$

$$P(f(y), f(y))$$
$$P(f(y), f(f(y))) \qquad sub = [x/y][y/f(y)][y/f(y)]$$

usw.

Keine Terminierung! Nicht-Unifizierbarkeit unentdeckt.

Prädikatenlogische Resolution

Definition (Resolvente)

Seien K_1, K_2 und R Klauseln, s_1 und s_2 Substitutionen, die Variablen so umbenennen, dass $K_1\,s_1$ und $K_2\,s_2$ keine Variablen gemeinsam haben. R heißt Resolvente von K_1 und K_2, wenn Folgendes gilt:

1. Es gibt Literale $L_1, \ldots, L_m$ in $K_1\,s_1$ und $L'_1, \ldots, L'_n$ in $K_2\,s_2$, so dass $\{-L_1, \ldots, -L_m, L'_1, \ldots, L'_n\}$ durch mgu *sub* unifizierbar ist.
2. $R = ((K_1\,s_1 \setminus \{L_1, \ldots, L_m\}) \cup (K_2\,s_2 \setminus \{L'_1, \ldots, L'_n\}))sub$.

Wieder:

$$\begin{array}{rcl} Res(F) & = & F \cup \{R \mid R \text{ Resolvente zweier Klauseln in } F\} \\ Res^0(F) & = & F \\ Res^{n+1}(F) & = & Res(Res^n(F)) \\ Res^*(F) & = & \bigcup_{n \geq 0} Res^n(F) \end{array}$$

Resolutionssatz: Sei F eine geschlossene Formel in Skolemform mit Matrix F^* in KNF. F ist unerfüllbar gdw. $\Box \in Res^*(F)$.

Warum funktioniert das?

Können relevante Grundinstanzen übersehen werden? NEIN

Lifting Lemma:
Seien C_1 und C_2 prädikatenlogische Klauseln mit Grundinstanzen G_1 bzw. G_2. Sei R_G Resolvente von G_1 und G_2. Dann gibt es eine Resolvente R von C_1 und C_2, so dass R_G Grundinstanz von R ist.

$$\begin{array}{ccccc} C_1 & & & & C_2 \\ \Downarrow & \searrow & & \swarrow & \Downarrow \\ \Downarrow & & R & & \Downarrow \\ G_1 & & \Downarrow & & G_2 \\ & \searrow & \Downarrow & \swarrow & \\ & & R_G & & \end{array}$$

Doppelpfeil: Instanzierung, Einfachpfeil: Resolution

$$
\begin{tikzcd}
\{\neg P(f(x)), Q(x)\} \arrow[dr] \arrow[dd, Rightarrow] & & \{P(f(g(y)))\} \arrow[dl] \arrow[dd, Rightarrow] \\
& \{Q(g(y))\} \arrow[dd, Rightarrow] & \\
\{\neg P(f(g(a))), Q(g(a))\} \arrow[dr] & & \{P(f(g(a)))\} \arrow[dl] \\
& \{Q(g(a))\} &
\end{tikzcd}
$$

Doppelpfeil: Instanzierung, Einfachpfeil: Resolution

Resolutionsbeispiel

Normale Vögel fliegen.	$\forall x(Vogel(x) \wedge Normal(x) \rightarrow Fliegt(x))$
Pinguine fliegen nicht.	$\forall x(Pinguin(x) \rightarrow \neg Fliegt(x))$
Strausse fliegen nicht.	$\forall x(Strauss(x) \rightarrow \neg Fliegt(x))$
Pinguine sind Vögel.	$\forall x(Pinguin(x) \rightarrow Vogel(x))$
Strausse sind Vögel.	$\forall x(Strauss(x) \rightarrow Vogel(x))$
Tweety Ping. oder Strauss.	$Pinguin(Tweety) \vee Strauss(Tweety)$
Tweety ist nicht normal.	$\neg Normal(Tweety)$

Resolutionsbeweis

(1)	$\{\neg Vogel(x), \neg Normal(x), Fliegt(x)\}$	
(2)	$\{\neg Pinguin(x), \neg Fliegt(x)\}$	
(3)	$\{\neg Strauss(x), \neg Fliegt(x)\}$	
(4)	$\{\neg Pinguin(x), Vogel(x)\}$	
(5)	$\{\neg Strauss(x), Vogel(x)\}$	
(6)	$\{Pinguin(Tweety), Strauss(Tweety)\}$	
(7)	$\{Normal(Tweety)\}$	neg. Ziel
(8)	$\{Strauss(Tweety), Vogel(Tweety)\}$	6,4
(9)	$\{Vogel(Tweety)\}$	8,5
(10)	$\{Pinguin(Tweety), \neg Fliegt(Tweety)\}$	6,3
(11)	$\{\neg Fliegt(Tweety)\}$	10,2
(12)	$\{\neg Normal(Tweety), Fliegt(Tweety)\}$	9,1
(13)	$\{\neg Normal(Tweety)\}$	11,12
(14)	$\square$	13,7

Resolutionsstrategien

Einschränkung der Wahlfreiheit bei Resolventenbildung:

1. P-Restriktion: eine Elternklausel positiv.
2. N-Restriktion: eine Elternklausel negativ.
3. lineare Resolution: eine Elternklausel zuletzt erzeugte Klausel.
4. Stützmengenrestriktion: sei $F = F^* \cup T$, so dass F^* erfüllbar; eine Elternklausel ist aus T (Stützmenge).
   Etwa: F^* erfüllbare Wissensbasis, T negierte Anfrage.
5. Input-Restriktion: eine Elternklausel aus der Klauselmenge F.
6. Einheitsrestriktion: eine Elternklausel einelementig.
7. SLD Resolution: nur definiert für Hornklauseln: Input-Resolution, wobei grundsätzlich mit negativer Klausel (Zielklausel) begonnen wird, die mit nicht-negativer (= definiter) Klausel resolviert wird. Es entstehen nur negative Klauseln.

Resolutionsstrategien: Vollständigkeitsresultate

1. Resolution unter P-Restriktion ist vollständig.
2. Resolution unter N-Restriktion ist vollständig.
3. Resolution unter linearer Resolutionsrestriktion ist vollständig.
4. Resolution unter Stützmengen-Restriktion ist vollständig.
5. Input-Resolution ist vollständig für Hornklauseln.
6. Einheitsresolution ist vollständig für Hornklauseln.
7. SLD-Resolution ist vollständig für Hornklauseln.