Angewandte Informatik
Universität Hannover

gesetzt von Sebastian Meyer

Mitschrift zur Vorlesung

Logik

Sommersemester 04

Empfohlendes Manuskript:

Uwe Schöning
Logik für Informatiker
Spektrum Verlag, ISBN 3-8274-1005-3, $\geq$ 5. Auflage

Inhaltsverzeichnis

1 Aussagenlogik

1.1 Grundbegriffe

„Otto ist krank“ und „Der Arzt verschreibt Otto eine Medizin“

Logische Struktur:

$$A := \text{Otto ist krank}$$
$$B := \text{Der Arzt verschreibt Otto eine Medizin}$$

Hier ist also die logische Struktur: A und B

1.1.1 Syntax der Aussagenlogik

Syntax = Lehre von den Zeichen

Sprache der Aussagenlogik

Aussagenvariablen:	$A_0, A_1, A_2, \dots$
Logische Zeichen:	$\neg, \wedge, \vee$
Technische Zeichen:	$(,)$

Formeln

1. Jede Aussagenvariable ist eine (atomare) Formel
2. Sind F, G Formeln, so auch $(F \wedge G), (F \vee G)$
3. Ist F eine Formel, so auch $(\neg F)$

Ist G eine Formel und ist G ein Teil einer Formel F, so heißt G eine **Teilformel** von F.

Beispiel

$$\neg((\neg(A_0 \vee A_1) \wedge A_2) \vee A_3)$$

Teilformeln dieses Beispiels sind:

$$A_0, A_1, A_2, A_3, (A_0 \vee A_1), \neg(A_0 \vee A_1), (\neg(A_0 \vee A_1) \wedge A_2), ((\neg(A_0 \vee A_1) \wedge A_2) \vee A_3)$$

Definition 1.1.1
- *$A, B, C, \dots$ heißen Aussagenvariablen*

- $(F_1 \to F_2)$ *steht für* $(\neg F_1 \vee F_2)$
- $(F_1 \leftrightarrow F_2)$ *steht für* $((F_1 \wedge F_2) \vee (\neg F_1 \wedge \neg F_2))$
- $\bigwedge\limits_{i=1}^{n} F_i$ *steht für* $(\ldots((F_1 \wedge F_2) \wedge F_3) \wedge \cdots \wedge F_n)$
- $\bigvee\limits_{i=1}^{n} F_i$ *steht für* $(\ldots((F_1 \vee F_2) \vee F_3) \vee \cdots \vee F_n)$

1.1.2 Semantik der Aussagenlogik

Semantik = Lehre von der Bedeutung der Zeichen

Elemente der Menge $\{0,1\}$ heißen **Wahrheitswert**. 0 steht für falsch und 1 für wahr.

Es sei D eine Menge von Aussagenvariablen. Eine Abbildung $\Theta : D \to \{0,1\}$ heißt eine **Belegung**. Es sei $E \supseteq D$ die Menge aller Formeln, die aus D konstruiert werden können. $\hat{\Theta} : E \to \{0,1\}$ sei definiert durch:

1. Ist $A \in D$, so ist $\hat{\Theta}(A) = \Theta(A)$
2. Sind $F, G \in E$, so ist

$$\hat{\Theta}(F \wedge G) = \begin{cases} 1, & \hat{\Theta}(F) = 1 = \hat{\Theta}(G) \\ 0, & \text{sonst} \end{cases}$$

$$\hat{\Theta}(F \vee G) = \begin{cases} 1, & \hat{\Theta}(F) = 1 \text{ oder } \hat{\Theta}(G) = 1 \\ 0, & \text{sonst} \end{cases}$$

3. Ist $F \in E$, so ist

$$\hat{\Theta}(\neg F) = \begin{cases} 1, & \hat{\Theta}(F) = 0 \\ 0, & \hat{\Theta}(F) = 1 \end{cases}$$

Beispiel

Seien $\Theta(A) = 1 = \Theta(B)$, $\Theta(C) = 0$ und

$$\begin{aligned} & \hat{\Theta}(\neg(A \vee B) \wedge C) = 1 \\ \Leftrightarrow\ & \hat{\Theta}(\neg(A \vee B)) = 1 \vee \Theta(C) = 1 \\ \Leftrightarrow\ & \hat{\Theta}(A \vee B) = 0 \text{ und } \Theta(C) = 1 \\ \Leftrightarrow\ & \Theta(A) = 0 = \Theta(B) \text{ und } \Theta(C) = 1 \end{aligned}$$

Folglich ist $\hat{\Theta}(\neg(A \vee B) \wedge C) = 0$

Wahrheitstafelverfahren

$\hat{\Theta}(F)$	$\hat{\Theta}(G)$	$\hat{\Theta}(F \wedge G)$
0	0	0
0	1	0
1	0	0
1	1	1

$\hat{\Theta}(F)$	$\hat{\Theta}(G)$	$\hat{\Theta}(F \vee G)$
0	0	0
0	1	1
1	0	1
1	1	1

$\hat{\Theta}(F)$	$\hat{\Theta}(\neg F)$
0	1
1	0

$\hat{\Theta}(F)$	$\hat{\Theta}(G)$	$\hat{\Theta}(F \leftrightarrow G)$
0	0	1
0	1	0
1	0	0
1	1	1

$\hat{\Theta}(F)$	$\hat{\Theta}(G)$	$\hat{\Theta}(F \rightarrow G)$
0	0	1
0	1	1
1	0	0
1	1	1

Bezeichnungen:

$(\neg F)$ **Negation von F**

$(F \wedge G)$ **Konjunktion**

$(F \vee G)$ **Disjunktion**

$F \rightarrow G$ **Implikation**

$F \rightarrow G$ **Äquivalenz**

1.1.3 Induktionsbeweis über den Aufbau der Formeln

Es sei P eine Eigenschaft die auf eine Formel zutriff oder nicht zutrifft. Um zu zeigen, dass eine Eigenschaft P auf **alle** Formeln zutrifft, gehen wir wie folgt vor:

Induktionsanfang:

Jede atomare Formel hat die Eigenschaft P.

Induktionsschluss:

Haben F, G die Eigenschaft P, so haben auch $(\neg F), (F \wedge G), (F \vee G)$ die Eigenschaft P.

Definition 1.1.2
*Es sei D eine Menge von atomaren Formeln und $\Theta : D \rightarrow \{0, 1\}$ eine **Belegung**. Eine Formel F **passt zur Belegung** Θ, falls jede Aussagenvariable, die in F vorkommt ein Element von D ist. Passt Θ zu F und gilt $\hat{\Theta}(F) = 1$, so ist Θ ein **Modell** von F, geschrieben $\Theta \models F$. Ist*

$\hat{\Theta} = 0$, *so schreibe:* $\Theta| \neq F$*. Ist* $\mathbb{F}$ *eine Menge von Formeln, die zu* Θ *passen und gilt für alle* $F \in \mathbb{F} : \Theta \models F$*, so ist* Θ *ein* ***Modell*** *von* $\mathbb{F}$*.*

Ein Formel F *bzw. eine Menge* $\mathbb{F}$ *von Formeln heißt* ***unerfüllbar****, falls sie kein Modell besitzen. Die Formel* F *heißt* ***gültig*** *bzw.* ***Tautologie****, falls jede Belegung* Θ*, die zu* F *passt, ein Modell von* F *ist. Man schreibt* $\models F$

Satz 1.1.1
Ein Formel F *ist genau dann eine Tautologie, wenn* $\neg F$ *unerfüllbar ist.*

Beweis

„$\Rightarrow$“ Kontraposition

Es sei $\neg F$ erfüllbar. Dann existiert ein Modell Θ von $\neg F$, d.h.

$$\Theta(\neg F) = 1.$$

Folglich ist $\Theta(F) = 0$ und somit ist F keine Tautologie.

„$\Leftarrow$“ Kontraposition

Angenommen F ist keine Tautologie. Dann existiert Belegung Θ mit $\Theta(F) = 0$. Somit ist $\Theta(\neg F) = 1$ und somit erfüllt Θ die Formel nicht.

Es sei F eine Formel mit genau n Aussagenvariablen. Um zu untersuchen, ob F eine Tautologie ist, müssen 2^n Belegungen Θ getestet werden. Kein effektives Verfahren!

$$F := (\neg A \leftrightarrow (A \rightarrow B)) \quad \text{Tautologie?}$$

A	B	$\neg$	A	$\leftrightarrow$	$(A$	$\rightarrow$	B	$)$
0	0	1	0	1	0	1	0	
0	1	1	0	1	0	1	1	
1	0	0	1	1	1	0	0	
1	1	0	1	0	1	1	1	

Also keine Tautologie!

1.2 Äquivalenz und Normalform

Definition 1.2.1
Zwei Formeln F *und* G *heißen* ***semantisch äquivalent****, falls für jede Belegung* Θ *von* F *und* G *die zu* F *und* G *passt, gilt:*

$$\Theta(F) = \Theta(G)$$

Sind F, G *semantisch äquivalenzt, so schreibe*

$$F \equiv G$$

Bemerkung

$F \equiv G$ gdw. $F \leftrightarrow G$ Tautologie

Satz 1.2.1 (Ersetzbarkeitstheorem)
Ist H eine Formel, F eine Teilformel von H, G eine zu F semantisch äquivalente Formel und geht H' aus H dadurch hervor, dass man ein Vorkommen von F in H durch G ersetzt, so ist

$$H' \equiv H$$

Beweis

Induktion über den Aufbau von H.

Ist F die Formel H, so ist $H' = G$ und nach Vorraussetzung ist $G \equiv F$ und $F = H$. Also $H' \equiv H$. Nun sei F ein **echte** Teilformel von H.

1. Fall $H = \neg H_1$

Es ist F ein Teilformel von H_1. Nach Induktionsvorraussetzung $H_1' \equiv H_1$. Somit $\neg H_1' \equiv \neg H_1$, d.h. $H' \equiv H$

2. Fall $H = (H_1 \wedge H_2)$

O.B.d.A sei F eine Teilformel von H_1. Nach Induktionsvorraussetzung $H_1' \equiv H_1$. Also $H_1' \wedge H_2 \equiv H_1 \wedge H_2$, d.h. $H' \equiv H$.

3. Fall $H = (H_1 \vee H_2)$

Entsprechend.

Klammerersparnisregeln

1. Außenklammern weglassen
2. $\neg$ bindet stärker als $\wedge, \vee, \rightarrow, \leftrightarrow$
3. $\wedge, \vee$ binden stärker als $\rightarrow, \leftrightarrow$
4. Statt $(F \wedge G) \wedge H$ schreibe $F \wedge G \wedge H$
   Statt $F \wedge (G \wedge H)$ schreibe $F \wedge G \wedge H$
   Entsprechend für $\vee$

Beispiel

$\neg F \vee G \rightarrow H$ steht für $((\neg F) \vee G) \rightarrow H$

Satz 1.2.2
Ist 1 ein Tautologie, steht 0 für eine unerfüllbare Formel, so gilt für F, G, H:

(1a)	$F \wedge F \equiv F$	*Idempotenzgesetz*
(1b)	$F \vee F \equiv F$	*Idempotenzgesetz*
(2a)	$F \wedge G \equiv G \wedge F$	*Kommutativität*
(2b)	$F \vee G \equiv G \vee F$	*Kommutativität*
(3a)	$F \wedge (G \wedge H) \equiv (F \wedge G) \wedge H$	*Assoziativität*
(3b)	$F \vee (G \vee H) \equiv (F \vee G) \vee H$	*Assoziativität*
(4a)	$F \wedge (F \vee G) \equiv F$	*Absorbtion*
(4b)	$F \vee (F \wedge G) \equiv F$	*Absorbtion*
(5a)	$F \wedge (G \vee H) \equiv (F \wedge G) \vee (F \wedge H)$	*Distributivität*
(5b)	$F \vee (G \wedge H) \equiv (F \vee G) \wedge (F \vee H)$	*Distributivität*
(6)	$\neg\neg F \equiv F$	*Doppelnegation*
(7a)	$\neg(F \wedge G) \equiv \neg F \vee \neg G$	*de Morganschen Regeln*
(7b)	$\neg(F \vee G) \equiv \neg F \wedge \neg G$	*de Morganschen Regeln*
(8a)	$1 \wedge G \equiv G$	*Tautologieregeln*
(8b)	$1 \vee G \equiv 1$	*Tautologieregeln*
(9a)	$0 \wedge G \equiv 0$	*Unerfüllbarkeitsregeln*
(9b)	$0 \wedge G \equiv G$	*Unerfüllbarkeitsregeln*
(10a)	$F \wedge \neg F \equiv 0$	
(10b)	$F \vee \neg F \equiv 1$	

Bemerkung

Die Regeln sind nicht unabhängig. Die Regeln $1a, 1b, 2a, 2b, 3a, 3b, 5a, 5b, 8a, 9b, 10a, 10b$ implizieren die übrigen Regeln.

$$
\begin{aligned}
1 \wedge G &\equiv (G \vee \neg G) \vee G && 10b, ET \\
&\equiv (\neg G \vee G) \vee G && 2b, ET \\
&\equiv \neg G \vee (G \vee G) && 3b \\
&\equiv \neg G \vee G && 1b, ET \\
&\equiv G \vee \neq G && 2b \\
&\equiv 1 && 10b
\end{aligned}
$$

Beispiel:

$$\begin{aligned}F :=&(\neg A \wedge B \wedge C) \vee (A \wedge \neg B \wedge C) \vee (A \wedge B \wedge \neg C)\\ &\vee (A \wedge \neg B \wedge \neg C) \vee (\neg A \wedge B \wedge \neg C)\\ &\vee (\neg A \wedge \neg B \wedge C) \vee (\neg A \wedge \neg B \wedge \neg C)\end{aligned}$$

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

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

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

Also

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

Definition 1.2.2

1. *Ein **Literal** ist eine atomare Formel oder die Angaben einer atomaren Formel.*

   *Ein Literal heißt **positiv**, falls es eine atomare Formel ist anderenfalls heißt sie **negativ**.*

2. *Eine Formel F ist in **konjunktiver Normalform** (KNF), falls F eine Konjunktion von Disjunktionen von Literalen ist, d.h.*

$$F = \bigwedge_{i=1}^{n} \bigvee_{k=1}^{m_i} L_{i,k} \qquad KNF$$

   wobei $L_{i,k}$ Literale sind.

3. *Eine Formel F ist in* ***disjunktiver Normalform****, falls*

$$F = \bigvee_{i=1}^{n} \bigwedge_{k=1}^{m_i} L_{i,k} \qquad DNF$$

wobei $L_{i,k}$ Literale sind.

Satz 1.2.3 (Normalformtheorem)
Zu jeder Formel existiert eine semantisch äquivalente Formel in konjunktiver Normalform und eine äquivalente Formel in disjunktiver Normalform.

Beweis

Simultan über Aufbau der Formel.

Induktionsanfang F ist eine atomare Formel. Dann ist F ein Literal und somit in KNF und DNF.

Induktionsschluss

1. Fall $F = \neg G$

   Nach Induktionsvorraussetzung

$$G = \bigwedge_{i=1}^{n} \bigvee_{k=1}^{m_i} L_{i,k} \qquad KNF$$

$$G = \bigvee_{i=1}^{n} \bigwedge_{k=1}^{m_i} L_{i,k} \qquad DNF$$

   Also

$$\neg G \equiv \bigwedge_{i=1}^{n} \bigvee_{k=1}^{m_i} \neg L^*_{i,k}$$

$$\neg G = \bigvee_{i=1}^{n} \bigwedge_{k=1}^{m_i} \neg L^*_{i,k}$$

$$L^*_{i,k} = \begin{cases} A_{i,k} & , L_{i,k} = \neg A_{i,k} \\ \neg A_{i,k} & , L_{i,k} = A_{i,k} \end{cases}$$

2. Fall $F = G \vee H$

   Nach Induktionsvorraussetzung

$$G \equiv \bigwedge_{i=1}^{n} G_i$$

$$H \equiv \bigwedge_{k=1}^{m} H_i$$

G_i, H_i Disjunktionen

$$\begin{aligned} G \vee H &\equiv \bigwedge_{i=1}^{n} G_i \vee \bigwedge_{k=1}^{m} H_1 \\ &\equiv \bigwedge_{i=1}^{n} (G_i \vee \bigwedge_{k=1}^{m} H_1) \\ &\equiv \bigwedge_{i=1}^{n} \bigwedge_{k=1}^{m} (G_i \vee H_1) \quad KNF \end{aligned}$$

Nach Induktionsvorraussetzung

$$G \equiv \bigvee_{i=1}^{n} G_i$$

$$H \equiv \bigvee_{k=1}^{m} H_i$$

Dann ist:

$$\begin{aligned} G \vee H &\equiv \bigvee_{i=1}^{n} G_i \vee \bigvee_{k=1}^{m} H_1 \\ &\equiv \bigvee_{i=1}^{n} (G_i \vee \bigvee_{k=1}^{m} H_1) \\ &\equiv \bigvee_{i=1}^{n} \bigvee_{k=1}^{m} (G_i \vee H_1) \quad DNF \end{aligned}$$

3. Fall $F = G \wedge H$

   Entsprechend.

1.2.1 Algorithmus zur Erstellung einer KNF

Gegeben sei Formel F

Schritt 1: Ersetze jedes Vorkommen einer Teilformel der Bauart

$$\neg\neg G \text{ durch } G$$

$$\neg(G \wedge H(\text{ durch } \neg G \vee \neg H$$

$$\neg(G \vee H(\text{ durch } \neg G \wedge \neg H$$

Schritt 2: Ersetze jedes Vorkommen einer Teilformel der Bauart

$$F \vee (G \wedge H) \text{ durch } (F \vee G) \wedge (F \vee H)$$

$$(F \wedge G) \vee H) \text{ durch } (F \vee H) \wedge (G \vee H)$$

1.2.2 Algorithmus zur Erstellung einer DNF

Formel F habe folgende Wahrheitstafel

A	B	C	$\Theta(F)$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

$$(\neg A \wedge B \wedge \neg C) \vee (A \wedge \neg B \wedge \neg C) \vee (A \wedge \neg B \wedge C) \vee (A \wedge B \wedge C)$$

1.3 Hornformeln

Definition 1.3.1
*Eine Formel F in KNF heißt **Hornformel**, falls jedes Konjunktionsglied höchstens ein positives Literal hat.*

Beispiel:

$$F = (A \vee \neg B) \wedge (\neg C \vee \neg A \vee D) \wedge (\neg A \vee \neg B) \wedge D \wedge \neg E$$

Umformen in Implikationen:

$$A \vee \neg B \equiv \neg B \vee A = B \to A$$
$$\neg C \vee \neg A \vee D \equiv \neg(A \wedge C) \vee D \equiv A \wedge C \to D$$
$$\neg A \vee \neg B \equiv \neg(A \wedge B) \equiv \neg(A \wedge B) \vee 0 \equiv A \wedge B \to 0$$
$$D \equiv D \vee 0 \equiv D \vee \neg 1 = 1 \to D$$
$$\neg E \equiv \neg E \vee 0 = E \to 0$$
$$F = (B \to A) \wedge (A \wedge C \to D) \wedge (A \wedge B \to 0) \wedge (1 \to D) \wedge (E \to 0)$$

Um F zu erfüllen, muss $\Theta(D) = 1$ und $\Theta(E) = 0$ gesetzt werden. Ferner setze $\Theta(A) = \Theta(B) = \Theta(C) = 0$.

Algorithmus (Erfüllbarkeitstest für Hornformeln)

Eingabe: Hornformel F

1. Ist F eine atomare Formel und kommt die Teilformel $(1 \to A)$ in F vor, so markiere alle Vorkommen von A in F.

2. `while` Es gibt in F eine Teilformel G der Form $(A_1 \wedge \cdots \wedge A_n \to B)$ oder $(A_1 \wedge \cdots \wedge A_n \to 0), n \geq 1$, wobei $A_1, \ldots, A_n$ bereits markiert sind und B nicht markiert ist, `do`

   `if` G hat die erste Form `then`

   markiere jedes Vorkommen von B in F

   `else` gib „unerfüllbar" aus und stopp.

3. Gib „unerfüllbar" aus und stopp.
   Wir setzen:

   $\Theta(A_i) = 1$, falls A_i markiert

   $\Theta(A_i) = 0$, falls A_i unmarkiert

Satz 1.3.1
Der Markierungsalgorithmus für Hornformeln ist korrekt und stoppt nach höchstens k Markierungsschritten, wobei k die Anzahl der atomaren Teilformeln von F ist.

Beweis:

Der Algorithmus stoppt nach höchstens k Schritten. Wir unterscheiden drei Fälle:

1. $(1 \to A)$

   Es wird A markiert und $\Theta(A) = 1$ gesetzt. Das Konjunktionsglied $(1 \to A)$ bekommt den Wahrheitswert 1.

2. $(A_1 \wedge \cdots \wedge A_n \to B)$

   a) $A_1, \ldots, A_n$ markiert.
   Dann wird B markiert und $A_1 \wedge \cdots \wedge A_n \to B$ bekommt Wahrheitswert 1.

   b) Ein A_i unmarkiert
   Dann wird $\Theta(A_i) = 0$ gesetzt. Folglich ist $\Theta(A_1 \wedge \cdots \wedge A_n \to B) = 1$.

3. $A_1 \wedge \cdots \wedge A_n \to 0$

   a) Alle $A_1, \ldots, A_n$ sind markiert.
   Dann ist zunächst die Markierung zwangsläufig, d.h. wir müssen notwendigerweise $A_1, \ldots, A_n$ mit 1 belegen. Folglich

   $$A_1 \wedge \cdots \wedge A_n \to 0$$

   unerfüllbar, somit F unerfüllbar.

b) Ein A_i ist unmarkiert
Dann $\Theta(A_i) = 0$ und somit $A_1 \wedge \dots \wedge A_n \to$ wahr.

Spezialfälle:

1. F enthält keine Teilformel der Art $(A_1 \wedge \dots \wedge A_n \to 0)$

   Dann belege alle atomaren Formeln mit 1: F ist erfüllbar.

2. $(1 \to A)$ ist keine Teilformel von F

   Dann belege alle atomaren Formeln mit 0: F ist erfüllbar.

1.4 Kompaktheitssatz

Satz 1.4.1 (Kompaktheitssatz)
Ist M eine Menge von Formeln und besitzt jede endliche Teilmenge von M ein Modell, so besitzt M ein Modell.

Beweis:

„$\Leftarrow$“ Besitzt M ein Modell so auch jede endliche Teilmenge von M.

„$\Rightarrow$“ Für $n \geq 0$ sei

$$M_n := \{F \in M : \text{ Ist } A \in F \text{ atomar, so existiert } i \leq n \text{ mit } A = A_i\}$$

offenbar ist

$$M_0 \subseteq M_1 \subseteq M_2 \subseteq \dots \subseteq M_n \subseteq \dots$$

Behauptung:

Es gibt genau $2^{2^{n+1}}$ viele Wahrheitstafeln für Formel F, die alle atomaren Teilformeln aus $\{A_0, \dots, A_n\}$ haben.

A_0	A_1	A_2	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

$\Big\}\ 2^{2^{2+1}}$ Möglichkeiten für die Belegungen

Es kann M_n unendlich sein. Andererseits gibt es höchstens $2^{2^{n+1}}$ paarweise **nicht äquivalente** Formeln aus M_n. Dies seine die Formeln $F_0, F_1, \dots, F_k \in M_n$. Nach Vorraussetzung existiert zu $F_0, F_1, \dots, F_k$ ein Modell Θ_n, das nach Konstruktion auch ein Modell von M_n ist.

1.4.1 Die Konstruktion

Start:

$\Theta = \phi$

$I = \{0, 1, 2, 3, \dots\}$

Stufe n:

`if` es gibt unendlich viele Indizes $j \in I$ mit $\Theta_i(A_n) = 1$ `then`

`begin`

$\Theta := \Theta \cup \{(A_n, 1)\}$

$I := I - \{i \in I : \Theta_i(A_n) \neq 1\}$

`else`

$\Theta = \Theta \cup \{(A_n, 0)\}$

$I := I - \{i \in I : \Theta_i(A_n) \neq 0\}$

1. Θ wohldefiniert
2. Θ Modell von M

   Es sei $F \in M$. Wähle $n \in \mathbb{N}$ mit $F \in M_n \subseteq M_{n+1} \subseteq M_{n+2} \subseteq \dots$. Es sind $\Theta_n, \Theta_{n+1}, \Theta n+2, \dots$ Modelle von M. Sind die atomaren Formeln von F unter $A_0, \dots, A_n$ zu finden, so existieren aufgrund des Streichungsprozesses unendlich viele $i \geq n$ mit

   $$\Theta_i(A_0) = \Theta(A_0), \dots, \Theta_i(A_n) = \Theta(A_n).$$

   Diese Θ_i sind Modelle von $M_i \supseteq M_n$ und somit ist Θ ein Modell von M_n und so von F.

1.5 Resolutionskalkül

Definition 1.5.1
*F **folgt aus** $G_1, \dots, G_n$, falls $G_1 \wedge \dots \wedge G_n \rightarrow F$ eine Tautologie ist.*

F folgt aus $G_1, \dots, G_n$
gdw. $G_1 \wedge \dots G_n \rightarrow F$ Tautologie
gdw. $G_1 \wedge \dots \wedge G_n \wedge \neg F$ unerfüllbar

Definition 1.5.2
*Ist F in KNF und $F = (L_{11} \vee \dots \vee L_{1n_1}) \wedge \dots \wedge (L_{k1} \vee \dots \vee L_{kn_k})$, so heißt $\{L_{i1}, \dots, L_{in_i}\}, i \leq u \leq k$ eine **Klausel**. Statt F schreibe auch*

$$\{\{L_{11}, \dots, L_{1n_1}\}, \dots, \{L_{k1}, \dots, L_{kn_k}\}\}$$

Definition 1.5.3
Ist L ein Literal, so ist

$$\overline{L} = \begin{cases} \neg A, & L = A \\ A, & L = \neg A \end{cases}$$

Definition 1.5.4
*Es seien K_1, K_2, R Klauseln. R heißt **Resolvent** von K_1 und K_2, falls ein Literal L existiert mit $L \in K_1, \overline{L} \in K_2$ und $R = (K_1 \backslash \{L\}) \cup (K_2 \backslash \{\overline{L}\})$*

$$K_1 \qquad K_2$$

$$R$$

Definition 1.5.5
Ist ϕ ein Resolvent von $K_1 = \{L\}, K_2 = \{\overline{L}\}$, so schreiben wir $\Box$ statt ϕ. Kommt $\Box$ in einer Klauselmenge vor, so gilt sie als unerfüllbar.

Ist $F = A \wedge \neg A \wedge \ldots$ so ist $\Box$ ein Resolvent. Offenbar ist F unerfüllbar.

Beispiel:

$$\{A, \neg B, C\}, \{A, B, C\}, \{\neg A, C\}, \{\neg C\}$$

$$\{A, C\}$$

$$\{C\}$$

$$\{\Box\}$$

Resolution

Lemma 1.5.1 (Resolutionslemma)
Es sei F eine Formel in KNF, dargestellt als Klauselmenge. Ferner sei R ein Resolvent der Klauseln K_1, K_2 von F. Dann gilt:

$$F \equiv F \cup \{R\}$$

Beweis

„$\Leftarrow$“

Es sei Θ eine Belegung mit $\Theta(F \cup \{R\}) = 1$. Dann ist $\Theta(F) = 1$, denn $F \cup \{R\}$ stellt eine Konjunktion dar.

„$\Rightarrow$“

Nun sei $\Theta(F) = 1$. Ferner sei K_1, K_2 Klauseln von F und L ein Literal mit $L \in K_1, \overline{L} \in K_2$ und $R = (K_1 \backslash \{L\}) \cup (K_2 \backslash \{\overline{L}\})$.

Dann $\Theta(L) = 1$ oder $\Theta(\overline{L}) = 1$.

O.B.d.A. sei $\Theta(L) = 1$. Dann ist $\Theta(\overline{L}) = 0$. Wegen $\Theta(F) = 1$ ist $\Theta(K_2) = 1$ und somit existiert ein Literal $L^* \in K_2$ mit $\Theta(L^*) = 1$. Offenbar ist $\overline{L} \neq L^*$. Es ist

$$R = (K_1 \backslash \{L\}) \cup (K_2 \backslash \{\overline{L}\})$$

und $L^* \in R$. Also $\Theta(R) = 1$.

Definition 1.5.6
Es sei F eine Klauselmenge.

1. $Res(F) := F \cup \{R : R \text{ ist Resolvent von } F\}$
2. $Res^0(F) = F$
   $Res^{n+1}(F) = Res(Res^n(F))$

Ferner sei $Res^*(F) = \bigcup\{Res^n(F) : n \geq 0\}$

Offenbar $Res^0(F) \subseteq Res^1(F) \subseteq Res^2(F) \subseteq \dots$

Lemma 1.5.2
Es sei F eine Klauselmenge in der genau die atomaren Formeln $A_1, \dots, A_n$ vorkommen. Dann ist

$$|Res^*(F)| \leq 4^n$$

Beweis

In den Formeln $G \in Res^*(F)$ kommen höchstens $A_1, \dots, A_n, \neg A_1, \dots, \neg A_n$ vor.

[Beachte: Formel = Klauselmenge]

Jeder Resolvent ist eine Teilmenge von

$$\{A_1, \dots, A_n, \neg A_1, \dots, \neg A_n\}.$$

Es ist

$$\begin{aligned} |Pot(\{A_1, \dots, A_n, \neg A_1, \dots, \neg A_n\}| &= 2^{2n} \\ &= \left(2^2\right)^n \\ &= 4^n \end{aligned}$$

Satz 1.5.1 (Resolutionssatz)
Eine Klauselmenge F ist genau dann unerfüllbar, wenn $\square \in Res^*(F)$ ist.

Beweis

1. **Korrektheit** („$\Leftarrow$")
   Es sei $\square \in Res^*(F)$. Dann existiert ein $n \in \mathbb{N}$ mit
   $$\square \in Res^n(F).$$
   Ferner $F \subseteq Res(F) \subseteq Res^2(F) \subseteq \dots \subseteq Res^n(F)$. Wähle $K_1 = \{L\}, K_2 = \{\overline{L}\} \in Res^m(F)$ mit $m < n$ und $\square = (K_1 \backslash \{L\}) \cup (K_2 \backslash \{\overline{L}\})$.
   Nach Resolutions-Lemma ist
   $$\begin{aligned} F &\equiv Res(F) \\ &\equiv Res^2(F) \\ &\equiv \dots \\ &\equiv Res^n(F) \end{aligned}$$
   Wegen $\{L\}, \{\overline{L}\} \in Res^n(F)$ ist $Res^n(F)$ unerfüllbar. Somit ist F unerfüllbar.

2. **Vollständigkeit** („$\Rightarrow$“)

   Es ist möglich, dass F unendlich ist. Nach Vorraussetzung sei F unerfüllbar. Nach Kompaktheitssatz existiert eine endliche Teilmenge F' von F, die unerfüllbar ist. Daher sei F O.B.d.A endlich und unerfüllbar.

   Es seien $A_1, \dots, A_n$ atomare Formeln und die atomaren Formeln von F mögen in $A_1, \dots, A_n$ vorkommen; ferner $A_n \in F$.

   Wir müssen zeigen, $\square \in Res^*(F)$. Dies beweise wir durch Induktion über n.

   Induktionsanfang: $n = 0$

   Dann ist $F = \{\square\}$ und somit ist $\square \in F = Res^0(F) \subseteq Res^*(F)$.

   Induktionsschritt

   Induktionsvorraussetzung Für jede unerfüllbare Formel G mit atomaren Formeln aus $A_1, \dots, A_n$ gilt: $\square \in Res^*(G)$.

   Nun sei eine Formel mit $A_1, \dots, A_{n+1} \in F$ und ferner sei F unerfüllbar.

   Die Formelmenge F_0 gehe F hervor, indem man jedes Vorkommmen von A_{n+1} in einer Klausel von F streicht und jede Klausel von F gänzlich streicht, falls in ihr $\neg A_{n+1}$ vorkommt.

   $$F = \{\dots\{\dots \cancel{A_{n+1}}\}, \dots, \cancel{\{\dots \neg A_{n+1}\}} \dots\}$$

   Nun sei Θ eine Belegung von F mit $\Theta(A_{n+1}) = 0$. Somit ist

   $$\Theta(F_0) = \Theta(F)$$

   Angenommen F_0 ist erfüllbar durch eine Belegung Θ'. Konstruiere Belegung Θ durch

   $$\Theta(B) = \begin{cases} \Theta'(B) & , B \in \{A_1, \dots, A_n\} \\ 0 & , B = A_{n+1} \end{cases}$$

   Es ist nach Annahme $\Theta'(F_0) = 1$. Nach obiger Bemerkung ist $\Theta(F_0) = \Theta(F)$. Wegen $\Theta \restriction F_0 = \Theta' \restriction F_0$ gilt:

   $$\Theta(F) = 1$$

   im Widerspruch zur Unerfüllbarkeit von F. Somit F_0 unerfüllbar.

   Die Formelmenge F_1 gehe aus F hervor, indem man jedes Vorkommen der Art $\neg A_{n+1}$ in Klauseln von F streicht und eine Klausel von F gänzlich streicht, wenn in ihr A_{n+1} vorkommt.

   $$F = \{\dots\{\dots \cancel{A_{n+1}}\}, \dots, \cancel{\{\dots \neg A_{n+1}\}} \dots\}$$

   Nun sei Θ Belegung mit $\Theta(A_{n+1}) = 1$. Dann ist $\Theta(F) = \Theta(F_1)$

   Wie oben zeigt man F_1 ist unerfüllbar.

   Nach Induktionsvorraussetzung gilt:

   $$\square \in Res^*(F_0), \square \in Res^*(F_1)$$

   Also existieren Klauseln $K_1, \dots, K_m \in Res^*(F_0)$ mit

a) $\square = K_m$

b) $K_i \in F_0$ oder K_i ist Resolvent zweier Klauseln K_j, K_l mit $j, l < i$.

[Man kann $K_1, \ldots, K_m$ als Beweis der Unerfüllbarkeit von F_0 bezeichnen]

Wegen $\square \in Res^*(F_1)$ existiert eine Folge $K_1', \ldots, K_n' \in Res^*(F_1)$ mit Eigenschaften (i), (ii) entsprechend.

Wir führen die gestrichene atomare Formel A_{n+1} in jedem Resolutionsschritt wieder ein. Dadurch erhält man eine Folge von Klauseln mit

$$\{A_{n+1}\} \in Res^*(F) \text{ oder } \square \in Res^*(F)$$

Entsprechendes für die Formelmenge F_1. Also

$$\{\neg A_{n+1}\} \in Res^*(F) \text{ oder } \square \in Res^*(F)$$

Daraus folgt $\square \in Res^*(F)$ oder $\{A_{n+1}\} \in Res^*(F), \{\neg A_{n+1}\} \in Res^*(F)$. Dann $\square \in Res^*(F)$.

2 Prädikatenlogik

2.1 Syntax und Semantik

Sprache L

- Variablen: $x_1, , x_2, x_3, \ldots$
- Prädikatssymbole: P_i^k
- Funktionssymbole: f_i^k
- Logische Symbole: $\neg, \wedge, \vee, \exists, \forall$
- Technische Zeichen: $(,)$

Bemerkung:

P_i^k i Unterscheidungsindex, k Stellenzahl
f_i^k i Unterscheidungsindex, k Stellenzahl

Beispiel:

1. $2 < 4$ wird ersetzt durch $< (2, 4)$, dies wird ersetzt durch $P_1^2(2, 4)$
2. $2 + 4$ wird ersetzt durch $+(2, 4)$, dies wird ersetzt durch $f_1^2(2, 4)$

Definition 2.1.1
*0-stellige Funktionssymbole heißen **Konstanten**.*

Definition 2.1.2 (Term)

1. *Jede Variable ist ein Term*
2. *Ist f_i^k ein Funktionssymbol und sind $t_1, \ldots, t_k$ Terme, so ist $f_i^k(t_, \ldots, t_k)$ ein Term*

Definition 2.1.3 (Formeln)

1. *Ist P_i^k ein Prädikatsymbol und sind $t_1, \ldots, t_k$ Terme, so ist $\left(P_i^k(t_1, \ldots, t_k)\right)$ eine (atomare) Formel.*
2. *Sind F, G Formeln, so auch $(\neg F), (F \wedge G), (F \vee G)$.*
3. *Ist F eine Formel, so auch $(\exists x_i\ F), (\forall x_i\ F)$.*

Definition 2.1.4

1. *Ist G eine Formel, die ein Teil einer Formel F ist, so ist G eine **Teilformel von** F.*

2. *Das Vorkommen der Variablen x in F heißt* **gebunden**, *falls x in einer Teilformel $\exists xG, \forall xG$ von F vorkommt. Kommt x in $\exists xG$ und in G vor, so ist x im Wirkungsbereich des Existenzquantors $\exists$. Entsprechendes für alle $\forall xG$.*

3. *x* ***kommt frei in F vor****, wenn x dort nicht im Wirkungsbereich eines Quantors steht.*

4. *Eine* ***Aussage*** *ist eine Formel in der keine Variable frei vorkommt.*

Beispiel:

Angeordneter Körper

Sprache

Prädikatsymbole: $=, <$

Funktionssymbole: $0, 1$ (0-stellig), $+, \cdot$ (2-stellig)

1. $\forall x \forall y \forall z((x + y) + z = x + (y + z))$
   $\forall x((x + 0) = x)$
   $\forall x \exists y(x + y = 0)$
   $\forall x \forall y(x + y = y + x)$

2. $\forall x \forall y \forall z((x \cdot y) \cdot z = x \cdot (y \cdot z))$
   $\forall x(x \cdot 1 = x)$
   $\forall x(\neg(x = 0) \Rightarrow \exists y(x \cdot y = 1))$
   $\forall x \forall y(x \cdot y = y \cdot x)$

3. $\forall x \forall y \forall z(x \cdot (y + z) = x \cdot + x \cdot z$

4. $\neg \exists x(x < x)$
   $\forall x \forall y \forall z((x < y) \wedge (y < z) \rightarrow x < z)$
   $\forall x \forall y(\neg(x = y) \rightarrow x < y \vee y < x)$

5. $\forall x \forall y \forall z(x < y \rightarrow x + z < y + z)$
   $\forall x \forall y \forall z(x < y \wedge 0 < z \rightarrow x \cdot z < y \cdot z)$

Bemerkung:

Jede Variable ist gebunden.

Vereinbarung

- Variablen: u, v, w, x, y, z
- Konstanten: a, b, c, d
- Funktionssymbole: f, g, h
- Prädikatssymbole: P, Q, R

Beispiel:

$$\forall x \exists y \exists z (f(x,y) = z) \land \forall y (y < x)$$

Definition 2.1.5 (Semantik)
Es sei U_A eine nichtleere Menge und I_A eine Abbildung.

$$\mathcal{A} = (U_A, I_A)$$

heißt eine ***zur Sprache L passende Struktur****, falls folgendes gilt:*

1. *I_A ordnet jedem k-stelligen Prädikatensymbol P eine k-stellige Relation über U_A zu, d.h.*

$$I_A(P) \subseteq U_A^k \left(= \underbrace{U_A \times \cdots \times U_A}_{\text{k-Faktoren}} \right)$$

2. *I_A ordnet jedem k-stelligen Funktionssymbol f eine k-stellige Funktion zu, d.h.*

$$I_A(f) : U_A^k \to U_A$$

3. *I_A ordnet jeder Variablen ein Element aus U_A zu, d.h.*

$$I_A(x_i) \in U_A$$

Schreibweise:

- $I_A(P)$ wird ersetzt durch $P^{\mathcal{A}}$
- $I_A(f)$ wird ersetzt durch $f^{\mathcal{A}}$
- $I_A(x)$ wird ersetzt durch $x^{\mathcal{A}}$

Beispiel:

Sprache

Prädikatensymbol: P (2-stellig)

Funktionssymbole: c, f (2-stellig)

$F := \forall x \forall y \, (P(x,y) \to P(x, f(c,y))) \land P(c,x)$

$U_A = \mathbb{N} \backslash \{0\}$

$I_A(P) = \{(m,n) m, n \in U_A \;\&\; m \text{ teilt } n\}$

$I_A(c) = 2$

$I_A(x) = 7 = I_A(y)$

$I_A(f) = \{((m,n),k) : m,n,k \in U_A \;\&\; m \cdot n = k\}$

Da 2 nicht 7 teilt ist F nicht gültig in $\mathcal{A}$.

Definition 2.1.6 (Wert eines Terms)
Es sei L eine Sprache und $\mathcal{A}$ eine zu L passende Struktur $\mathcal{A}$. Ist t ein Term, so definieren wir induktiv den Wert $\mathcal{A}(t) \in U_A$.

1. *Ist x eine Variable, so sei $\mathcal{A}(x) = x^{\mathcal{A}}$*
2. *Sind $t_1, \ldots, t_k$ Terme und ist f ein k-stelliges Funktionssymbol, so sei*

$$\mathcal{A}(f(t_1, \ldots, t_k)) = f^{\mathcal{A}}(\mathcal{A}(t_1), \ldots, \mathcal{A}(t_k))$$

Bemerkung

$\mathcal{A}(c) = c^{\mathcal{A}}$, falls c Konstante

Definition 2.1.7
Es sei $\mathcal{A}$ eine zu L passende Struktur. Induktiv definieren wir

$$\mathcal{A} : \{F : F \text{ Formel in } L\} \to \{0, 1\}$$

$\mathcal{A}(F)$ heißt **Wahrheitswert von** *F.*

1. *Ist $F = P(t_1, \ldots, t_k)$, so gilt:*
   $\mathcal{A}(P(t_1, \ldots, t_k)) = 1$ *gdw.* $(\mathcal{A}(t_1), \ldots, \mathcal{A}(t_k)) \in P^{\mathcal{A}}$
   Ist P das Gleichheitszeichen, so soll gelten:

$$\mathcal{A}(t_1 = t_2) = 1 \text{ gdw. } \mathcal{A}(t_1) = \mathcal{A}(t_2)$$

2. *$F = \neg G$, so gilt:*

$$\mathcal{A}(\neg G) = 1 \text{ gdw. } \mathcal{A}(G) = 0$$

3. $\mathcal{A}(F \wedge G) = 1$ *gdw* $\mathcal{A}(F) = 1 = \mathcal{A}(G)$
4. $\mathcal{A}(F \vee G) = 1$ *gdw* $\mathcal{A}(F) = 1$ *oder* $\mathcal{A}(G) = 1$
5. *Ist $F = \exists x G$, so*

$$\mathcal{A}(\exists x G) = 1 \text{ gdw ein } d \in U_A \text{ existiert mit } \mathcal{A}_{[x/d]}(G) = 1$$

6. *Ist $F = \forall G$, so*

$$\mathcal{A}(\forall x F) = 1 \text{ gdw für alle } d \in U_A \text{ gilt: } \mathcal{A}_{[x/d]}(G) = 1$$

Ist $\mathcal{A}' := \mathcal{A}_{[x/d]}$, so geht die Struktur $\mathcal{A}'$ aus $\mathcal{A}$ dadurch hervor, dass x mit d interpretiert wird und alle anderen Interpretationen von $\mathcal{A}$ erhalten bleiben.

Definition 2.1.8
Es sei F eine Formel aus L und $\mathcal{A}$ eine zu L passende Struktur

1. *F **gilt** in $\mathcal{A}$, geschrieben*
$$\mathcal{A} \models F,$$
*falls $\mathcal{A}(F) = 1$. Man sagt auch: $\mathcal{A}$ ist ein **Modell von** F.*

2. *F ist (allgemein-)**gültig**, falls F in jeder Struktur von L gilt.*

3. *F heißt **erfüllbar**, falls F ein Modell besitzt.*

Beispiel:

1. $\forall x \forall y (x + y = y + x)$ erfüllbar aber nicht allgemeingültig.

2. $\forall x (x = x)$ allgemeingültig

Definition 2.1.9

1. *F **folgt aus** $F_1, \ldots, F_k$, wenn für jede Struktur $\mathcal{A}$ mit $\mathcal{A}(F_1) = \cdots = \mathcal{A}(F_k) = 1$ gilt $\mathcal{A}(F) = 1$*

2. *F_1, F_2 sind (semantisch) **äquivalent**, falls $\mathcal{A}(F_1) = \mathcal{A}(F_2)$ für alle zu L passende Strukturen $\mathcal{A}$.*

Lemma 2.1.1
F ist allgemeingültig gdw $\neg F$ unerfüllbar ist.

2.2 Normalformen

Beispiel
Sind F, G, H Formeln der Prädikatenlogik, so gill

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

zum Beweis sei $\mathcal{A}$ eine zu L passende Struktur, $A_0 := F, A_1 := G, A_2 := H$ und die Belegung Θ sei definiert durch $\Theta(A_0) := \mathcal{A}(F), \Theta(A_1) := \mathcal{A}(G), \Theta(A_2) := \mathcal{A}(H)$. Dann gilt

$\mathcal{A}(F \wedge (G \vee H)) = 1$

gdw. $\mathcal{A}(F) = 1 = \mathcal{A}(G \vee H)$

gdw. $\mathcal{A}(F) = 1$ und $(\mathcal{A}(G) = 1$ oder $\mathcal{A}(H) = 1)$

gdw. $\Theta(A_0) = 1$ und $(\Theta(A_1) = 1$ oder $\Theta(A_2) = 1)$

gdw. $\Theta(A_0 \wedge (A_1, \vee A_2)) = 1$

gdw. $\Theta((A_0 \wedge A_1) \vee (A_0 \wedge A_2)) = 1$

gdw. $\Theta(A_0 \wedge A_1) = 1$ oder $\Theta(A_0 \wedge A_2) = 1$

gdw. $\Theta(A_0) = 1 = \Theta(A_1)$ oder $\Theta(A_0) = 1 = \Theta(A_2)$

gdw. $\mathcal{A}(F) = 1 = \mathcal{A}(G)$ oder $\mathcal{A}(F) = 1 = \mathcal{A}(H)$

gdw. $\mathcal{A}((F \wedge G) \vee (F \wedge H)) = 1$

■

Lemma 2.2.1
Sind F, G, H Formeln der Prädikatenlogik, so gelten die (semantischen) Äquivalenzen aus Satz 1.2.2.

Ferner gilt: Geht F' aus F hervor indem man gewisse Vorkommnisse von $G_1, \ldots, G_n$ durch $G'_1, \ldots, G'_n$ ersetzt und sind $G_1 \equiv G'_1, \ldots, G_n \equiv G'_n$, so ist $F' \equiv F$.

Satz 2.2.1
Es seien F, G Formeln.

1. $\neg\forall xF \equiv \exists x\neg F$
   $\neq \exists xF \equiv \forall x\neg F$
2. *Kommt x **nicht** frei vor in G, so gilt*
   $\forall xF \wedge G \equiv \forall x(F \wedge G)$
   $\forall xF \vee G \equiv \forall x(f \vee G)$
   $\exists xF \wedge G \equiv \exists x(F \wedge G)$
   $\exists xF \vee G \equiv \exists x(f \vee G)$
3. $\forall xF \wedge \forall xG \equiv \forall x(F \wedge G)$
   $\exists xF \vee \exists xG \equiv \exists(F \vee G)$
4. $\forall x\forall yF \equiv \forall y\forall xF$
   $\exists x\exists yF \equiv \exists y\exists xF$

Beweis zu 2.

$\mathcal{A}(\exists xF \vee G) = 1$

gdw. $\mathcal{A}(\exists xF) = 1$ oder $\mathcal{A}(G) = 1$

gdw. es existiert $d \in U_A$ mit $\mathcal{A}_{[x/d]}(F) = 1$ **oder** $\mathcal{A}(G) = 1$

gdw. es existiert $d \in U_A$ mit $(\mathcal{A}_{[x/d]}(F) = 1$ **oder** $\mathcal{A}_{[x/d]}(G) = 1)$

gdw. $\mathcal{A}(\exists x(F \vee G)) = 1$

Beispiel

$F = \exists z(z + z = x)$

$U_A := \mathbb{N}, I_A(+) = +, (U_A, I_A)$ Struktur, $I_A : \{x_i : i \geq 1\} \to [0, 1]$

$(U_A, I_A) \models \exists z(z + z = x)$

gdw. es existiert ein $d \in U_A$ mit $d + d = I_A(x)$

gdw. $I_A(x)$ ist gerade

1. Ersetze in F x durch y. Dann $(U_A, I_A) \models \exists z(z + z = y)$
   gdw. $I_A(y)$ gerade

2. Ersetze in F x durch z. Dann $(U_A, I_A) \models \exists z(z + z = z)$
   gdw. es existiert $d \in U_A$ mit $d + d = d$
   Also gilt die Formel $\exists z(z + z = z)$ in (U_A, I_A) immmer!

Dieses Beispiel zeigt, dass sich der Wahrheitsgehalt einer Formel ändert, wenn man in F das freie Vorkommen einer Variablen durch einen Term ersetzt, der eine Variable erhält, die nach Ersetzung in den Wirkungsbereich eines Quantors kommt.

Definition 2.2.1
Es sei F eine Formel, x eine Variable und t ein Term.

1. *t **ist frei für** x **in** F, falls keine Variable von t nach Ersetzung eines jeden freien Vorkommens von x in F in den Wirkungsbereich eines Quantors kommt.*
2. *Ist t frei für x in F, so berechne*
   $$F[x/t]$$
   diejenige Formel, die aus F durch Ersetzen aller freien Vorkommen von x durch t entsteht.

Definition 2.2.2
Sind t, t' Terme und ist x eine Variable, so bezeichne

$$t'[x/t]$$

denjenigen Term, der entsteht, wenn man jedes Vorkommen von x in t' durch t ersetzt.

Satz 2.2.2 (Überführungslemma)
Sind t, t' Terme und ist $\mathcal{A}$ eine zu L passende Struktur, so gilt:

$$\mathcal{A}(t'[x/t]) = \mathcal{A}_{[x/\mathcal{A}(t)]}(t')$$

Beweis

Durch Induktion über Aufbau der Terme.

Satz 2.2.3 (Überführungssatz)
Ist F eine Formel, x eine Variable, t ein Term und ist t frei für x in F, so ist

$$\mathcal{A}(F[x/t]) = \mathcal{A}_{[x/\mathcal{A}(t)]}(F)$$

Beweis

Induktion über den Aufbau von F

1. $F = P(t_1, \ldots, t_n)$
   Es ist
   $$\begin{aligned} \mathcal{A}(P(t_1, \ldots, t_n)[x/t]) &= \mathcal{A}(O(t_1[x/t], \ldots, t_n[x/t])) \\ &= O^{\mathcal{A}}(\mathcal{A}(t_1[x/t]), \ldots, \mathcal{A}(t_n[x/t])) \\ \text{Überführungslemma} &= P^{\mathcal{A}}(\mathcal{A}_{[x/\mathcal{A}(t)]}(t_1), \ldots, \mathcal{A}_{[x/\mathcal{A}(t)]}(t_n)) \\ &= P^{\mathcal{A}_{[x/\mathcal{A}(t)]}}(\mathcal{A}_{[x/\mathcal{A}(t)]}(t_1), \ldots, \mathcal{A}_{[x/\mathcal{A}(t)]}(t_n)) \\ &= \mathcal{A}_{[x/\mathcal{A}(t)]}(P(t_1, \ldots, t_n)) \end{aligned}$$

2. $F = G \wedge H$
   $F[x/t] = (G \wedge H)[x/t] = G[x/t] \wedge H[x/t]$
   $\mathcal{A}(F[x/t]) = 1$
   gdw. $\mathcal{A}(G[x/t] \wedge H[x/t]) = 1$
   gdw. $\mathcal{A}(G[x/t]) = 1 = \mathcal{A}(H[x/t])$
   gdw. (I.V.) $\mathcal{A}_{[x/\mathcal{A}(t)]}(G) = 1 = \mathcal{A}_{[x/\mathcal{A}(t)]}(H)$
   gdw. $\mathcal{A}_{[x/\mathcal{A}(t)]}(G \wedge H) = 1$
3. $F = G \vee H$ entsprechend
4. $F = \neg G$ entsprechend
5. $F = \forall y G, x \neq y$
   $\mathcal{A}(\forall y G_{[x/t]}) = 1$
   gdw. für alle $d \in U_A$ gilt $\mathcal{A}_{[y/d]}(G)[x/t] = 1$ für alle $d \in U_A$ gilt $\mathcal{A}_{[y/d]}(G)_{[x/\mathcal{A}(t)]} = 1$
   gdw. für alle $d \in U_A$ gilt $\mathcal{A}_{[x/\mathcal{A}(t)]}(G)_{[y/d]} = 1$
   gdw. $\mathcal{A}_{[x/\mathcal{A}(t)]}(\forall y G) = 1$
6. $\exists y G$ entsprechend

Satz 2.2.4 (Satz von der gebundenen Umbenennung)
Ist $Q \in \{\forall, \exists\}, F = Q \times G$ eine Formel und y eine Variable, die in G nicht vorkommt, so gilt

$$F \equiv F[x/y], \text{ d.h.}$$
$$F \equiv QyG[x/y]$$

Beweis

Es sei $Q = \forall$. Dann gilt: $\mathcal{A}(\forall \times G) = 1$ **gdw.** für alle $f \in U_A$ gilt $\mathcal{A}_{[x/d]}(G) = 1$. Da y in G nicht vorkommt, gilt:

$$\mathcal{A}_{[x/d][y/d]}(G) = 1$$
$$\mathcal{A}_{[y/d]}(G[x/y]) = 1$$

Folglich gilt:

$\mathcal{A}(\forall \times G) = 1$

gdw. für alle $d \in U_A$ $\mathcal{A}_{[y/d]}(G[x/y]) = 1$

gdw. $\mathcal{A}(\forall y G[x/y]) = 1$

Analog für $\exists$.

Definition 2.2.3
*F heißt **bereinigt**, falls*

1. *keine Variable kommt in F frei und gebunden vor*

2. *hinter allen Quantoren stehen verschiedene Variablen*

Definition 2.2.4
F heißt in pränexer ***Normalform****, falls F die Gestalt*

$$Q_1 y_1 \dots Q_n y_n G$$

mit $Q_i \in \{\forall, \exists\}$ für alle $i \in \{1, \dots, n\}$ hat und $n \geq 0$ ist; ferner kommt in G ***kein*** *Qunator vor.*

G heißt ***Kern*** *oder* ***Matrix*** *von F.*

Satz 2.2.5
Zu jeder Formel F existiert eine bereinigte Formel G in pränexer Normalform, die zu F äquivalent ist.

Beweis

Induktion über Aufbau.

Induktionsanfang Atomare Formel

Induktionsschluss

1. $F = \neg F_1$

   Nach Induktionsvorraussetzung: $F_1 \equiv Q_1 y_1 \dots Q_n y_n F_1'$

   Dann:

$$\begin{aligned} \neg F &\equiv neg Q_1 y_1 \dots Q_n y_n F_1' \\ &\equiv \overline{Q_1} y_1 \dots \overline{Q_n} y_n \neg F_1' \end{aligned}$$

2. $F = F_1 \circ F_2, \circ \in \{\wedge, \vee\}$

   Induktionsvorraussetzung:

   $F_1 = Q_1 y_1 \dots Q_n y_n F_1'$

   $F_2 = Q_1' z_1 \dots Q_l' z_l F_2'$

   Durch gegebene Umbenennung erreicht man

   $\{y_1, \dots, y_n\} \cap \{z_1, \dots, z_l\} = \phi$

   $F_1 \circ F_2 = Q_1 y_1 \dots Q_n y_n Q_1' z_1 \dots Q_l' z_l (F_1' \circ F_2')$

3. $F = Q \times G$

   Nach Induktionsvoraussetzung: $G \equiv Q_1 y_1 \dots Q_n y_n H$

   Nach Umbennung: $X \notin \{y_1, \dots, y_n\}$

   Also $F \equiv Q \times Q_1 y_1 \dots Q_n y_n H$.

Beispiel:

$$
\begin{aligned}
&\exists y \forall x (P(x, f(y)) \wedge R(z)) \vee \neg \exists x Q(x, h(x)) \\
2.2.5 &\Leftrightarrow \exists y \forall x (P(x, f(y)) \wedge R(z)) \vee \neg \exists u Q(u, h(u)) \\
2.2.2 &\Leftrightarrow \exists y \forall x (P(x, f(y)) \wedge R(z)) \vee \forall u \neg Q(u, h(u)) \\
2.2.2 &\Leftrightarrow \exists y (\forall x (P(x, f(y)) \wedge R(z)) \vee \forall u \neg Q(u, h(u))) \\
2.2.2 &\Leftrightarrow \exists y \forall x ((P(x, f(y)) \wedge R(z)) \vee \forall u \neg Q(u, h(u))) \\
2.2.2 &\Leftrightarrow \exists y \forall x \forall u ((P(x, f(y)) \wedge R(z)) \vee \forall u \neg Q(u, h(u)))
\end{aligned}
$$

Definition 2.2.5
Ist F eine bereinigte Formel in pränexer Normalform, so ist F ein BPF.

Definition 2.2.6
Zu der Formel F in BPF definieren wir ihre ***Skolenform****, die durch Anwendung des folgenden Algorithmus auf F aus F hervorgeht.*

`while` F enthält einen Existenzquantor `do`

`begin`

F habe die Form $\forall y_1 \dots \forall y_n \exists z G$ für eine Formel G in BPF und $n \geq 0$ ($n = 0$ bedeutet, dass der Allquantorblock leer ist); es sei f ein neues, n-stelliges Funktionssymbol, das in F nicht vorkommt;

$$F' = \forall y_1 \dots \forall y_n G[z/f(y_1, \dots, y_n)]$$

`end`

Beispiel:

$\exists x \forall y \exists z \exists u \forall v \exists w P(x, y, z, u, v, w)$

$\forall y \exists z \exists u \forall v \exists w P(a, y, z, u, v, w)$

$\forall y \exists u \forall v \exists w P(a, y, f(y), u, v, w)$

$\forall y \forall v \exists w P(a, y, f(y), g(y), v, w)$

$\forall y \forall v P(a, y, f(y), g(y), v, h(y, v))$

Satz 2.2.6
Für jede Formel F in BPF gilt:

F ist erfüllbar genau dann, wenn die Skolemform von F erfüllbar ist.

Beweis:

Ist $F = \forall y_1 \dots \forall y_n \exists z G$ mit G in BPF und entsteht in der `while`-Schleife die Formel

$$F' = \forall y_1 \dots \forall y_n G[z/f(y_1, \dots, y_n)],$$

so zeigen wir:

F erfüllbar gdw. F' erfüllbar.

Iteration liefert Resultat.

„⇒“ Es sei $\mathcal{A}$ ein Modell von F, d.h.

$$\mathcal{A}(\forall y_1 \dots \forall y_n \exists z G) = 1$$

Daher gilt: Zu $u_1, \dots, u_n \in U_A$ existiert ein $v \in U_A$ mit

$$\mathcal{A}_{[y_1/u_1]\dots[y_n/u_n][z/v]}(G) = 1$$

Ist L die Sprache von F, so erweitere L durch n-stelliges Funktionszeichen f.

L' sei die erweiterte Sprache. Erweitere die Struktur $\mathcal{A}$ durch Hinzunahme der Funktion

$$f^{\mathcal{A}'} : U_A^n \to U_A$$

definiert durch: Sind $u_1, \dots, u_n \in U_A = U_{A'}$, so sei $f^{\mathcal{A}'}(u_1, \dots, u_n) = v$, wobei

$$(*) \mathcal{A}_{[y_1/u_1]\dots[y_n/u_n][z/v]}(G) = 1$$

(*) ergibt:

$$\mathcal{A}'_{[y_1/u_1]\dots[y_n/u_n][z/f^{\mathcal{A}'}(u_1,\dots,u_n)]}(G) = 1.$$

Nach Überführungssatz ist (*) äquivalent zu:

$$\mathcal{A}'_{[y_1/u_1]\dots[y_n/u_n]}(G[z/f^{\mathcal{A}'}(y_1, \dots, y_n)]) = 1$$

Somit: Für alle $u_1, \dots, u_n \in U_A$ gilt:

$$\mathcal{A}'_{[y_1/u_1]\dots[y_n/u_n]}(G[z/f^{\mathcal{A}'}(y_1, \dots, y_n)]) = 1$$

Folglich $\mathcal{A}'(\forall y_1 \dots \forall y_n G[z/f(y_1, \dots, y_n)]) = 1$

„⇐“ Es gelte:

$$\mathcal{A}'(\forall y_1 \dots \forall y_n G[z/f(y_1, \dots, y_n)]) = 1$$

Dann gilt: Für alle $u_1, \dots, u_n \in U_{A'}$ gilt:

$$\mathcal{A}'_{[y_1/u_1]\dots[y_n/u_n]}(G[z/f(y_1, \dots, y_n)]) = 1$$

Nach Überführungssatz: Für alle $u_1, \dots, u_n \in U_{A'}$ gilt

$$\mathcal{A}'_{[y_1/u_1]\dots[y_n/u_n][z/f^{\mathcal{A}'}(u_1,\dots,u_n)]}(G) = 1$$

Dies ist äquivalent zu:

$$\mathcal{A}'(\forall y_1 \dots \forall y_n \exists z G) = 1$$

Bemerkung:

Sind $y_1, \dots, y_n$ alle freien Variablen von F, so gilt:

F erfüllbar gdw. $\exists y_1 \dots \exists y_n F$ erfüllbar

Umwandlung einer Formel in eine Klauselmenge

Gegeben sei F mit freien Variablen $y_1, \dots, y_n$ ($n = 0$ zugelassen)

1. Schritt Bringe F durch gebundene Umbenennung in bereingte Form. Man erhält F_1.

2. Schritt Sind $y_1, \dots, y_n$ die freien Variablen von F_1, so ersetze $y_1, \dots, y_n$ durch paarweise verschiedene, neue Konstanten. Man erhält due erfüllbarkeitsäquivalente Formel F_2.

3. Schritt Erstelle zu F_2 eine äquivalente Formel F_3 in pränexer Normalform.

4. Schritt Eliminiere die Existenzquantoren von F_3 durch Übergang zur Skolemform F_4.

5. Schritt Schreibe die Matrix von F_4 in KNF und schreibe diese Formel F_5 als Klauselmenge.

Bemerkung: Ist $F = \forall y_1 \dots \forall y_n G$ eine Formel in Skolemform, G quantorenfrei und in KNF, so gilt: Ist F erfüllbar, so auch G.

Beweis:

Sei $\mathcal{A}$ ein Modell von F. Wähle $a_1, \dots, a_n \in U_A$ aus. Wegen $\mathcal{A}(\forall y_1 \dots \forall y_n G) = 1$, gilt:

$$\mathcal{A}_{[y_1/a_1]\dots[y_n/a_n]}(G) = 1,$$

d.h.

$$\mathcal{A}_{[y_1/a_1]\dots[y_n/a_n]} \models G$$

Umkehrung gilt im allgemeinen nicht.

Betrachte $F = \forall c(c \neq c)$

F ist unerfüllbar. Aber ist $\mathcal{A}$ eine Struktur und hat U_A mindestens 2 Elemente, so ist $\mathcal{A}$ ein Modell von $x \neq c$.

Beispiel:

$$F = (\neg\forall x P(x,z) \land \exists y Q(x, f(y))) \lor \forall y Q(g(y,x), z)$$

1. Schritt

$$F_1 = (\neg\forall u P(u,z) \land \exists y Q(x, f(y))) \lor \forall v Q(f(v,x), z)$$

2. Schritt

$$F_2 = (\neg\forall u P(u,a) \land \exists y Q(b, f(y))) \lor \forall v Q(g(v,b), a)$$

3. Schritt

$$F_3 = \exists u \exists y \forall v((\neg P(u,a) \land Q(b, f(y))) \lor Q(g(v,b), a))$$

4. Schritt

$$F_4 = \forall v((\neg P(c,a) \land Q(b, f(d))) \lor Q(g(v,b), a))$$

5. Schritt

$$F_5 = \forall v((\neg P(c,a) \vee Q(g(v,b),a)) \wedge Q(b,f(d)) \vee Q(g(v,b),a)))$$

6. Schritt

$$F_6 = \{\{\neg P(c,a), Q(g(v),b),a)\}, \{Q(b,f(d)), Q(g(v,b),a)\}\}$$

2.3 Herbrand-Expansion

Es sei F eine Aussage (ohne freie Variablen). Enthält F keine Konstante, so erweitere die Sprache L von F um eine Konstante a. Ist F in Skolemform, so definiere rekursiv eine Menge $D^n(F)$ von Termen:

$$D^0(F) = \begin{cases} \text{Menge aller Konstanten in } F & \text{,falls } F \text{ eine Konstante enthält} \\ \{a\} & \text{,sonst.} \end{cases}$$

Es sei $D^n(F)$ bereits definiert.

$D^{n+1}(F) = D^n(F) \cup \{f(t_1, \ldots, t_k)\}$: f kommt in F vor und $t_1, \ldots, t_k \in D^n(F)$.

$$D(F) := \bigcup \{D^n(F) : n \geq 0\}$$

Herband-Universum von F

Beispiel:

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

   F enthält keine Konstante

   $D^0(F) = \{a\}$

   $D(F) = \{a, f(a), g(a,a), f(f(a)), f(g(a,a)), f(f(f(a))), g(a, f(a)), g(f(a), a), g(f(a), f(a)), \ldots\}$

2. $G = \forall x \forall y Q(c, f(x), h(y,b))$

   $D(G) = \{b, c, f(b), f(c), h(b,b), h(b,c), h(c,b), h(c,c), f(f(b)), f(f(c)), f(h(b,b)), \ldots\}$

Definition 2.3.1 (Herbrand-Struktur)
Es sei F eine Ausssage in Skolemform.

Die Herbrand-Struktur $\mathcal{A}$ sei definiert durch

1. $U_A := D(F)$

2. *Ist f ein n-stelliges Funktionssymbol, das in F vorkommt und sind $t_1, \ldots, t_n \in D(F)$, so sei $f^{\mathcal{A}}(t_1, \ldots, t_n) := f(t_1, \ldots, t_n)$.*

Definition 2.3.2
Ist F eine Aussage in Skolemform, so heißt eine Struktur $\mathcal{A}$ ein **Herbrand-Modell** *von F, falls $\mathcal{A}$ ein Herbrand-Struktur ist mit*

$$\mathcal{A} \models F$$

Satz 2.3.1
Ist F eine Aussage in Skolemform ohne Gleichheitszeichen, so ist F genau dann erfüllbar, wenn F ein Herbrand-Modell besitzt.

Beweis:

„$\Leftarrow$" klar!

„$\Rightarrow$" Es sei $\mathcal{A}$ ein Modell von F. Gesucht ist ein Herbrand-Modell $\mathcal{B}$ von F. Es sei $U_B := D(F)$ das Universum von B.

- Ist f ein n-stelliges Funktionssymbol, das in F vorkommt und sind $t_1, \ldots, t_n \in D(F)$, so ist
$$f^{\mathcal{B}}(f_1, \ldots, t_n) := f(t_1, \ldots, t_n)$$
- Ist P ein n-stelliges Prädikatssymbol das in F vorkommt und sind $t_1, \ldots, t_n \in D(F)$, so sei

$$\boxed{(t_1, \ldots, t_n) \in P^{\mathcal{B}} \text{ gdw. } {}_D f(\mathcal{A}(t_1), \ldots, \mathcal{A}(t_n)) \in P^{\mathcal{A}}}$$

Behauptung: $\mathcal{B} \models F$

Es habe F die Gestalt
$$F = \forall y_1 \ldots \forall y_n G,$$
G quantorenfrei. Wir beweisen die Behauptung über die Anzahl n der Quantoren.

Induktionsanfang n=0

F hat keinen Allquantor.

Zunächst sei $F = P(t_1, \ldots, t_k)$. Nach Vorraussetzung gilt $\mathcal{A}(P(t_1, \ldots, t_k)) = 1$.

Ferner gilt nach Konstruktion von $\mathcal{B}$:

$\mathcal{B}(P(t_1, \ldots, t_k)) = 1$

gdw. $P^{\mathcal{A}}(\mathcal{A}(t_1), \ldots, \mathcal{A}(t_k)) = 1$

gdw. $\mathcal{A}(P(t_1, \ldots, t_k)) = 1$

Ist F eine boolsche Kombination von atomaren Formeln, so führe einen Induktionsbeweis über den aussagenlogischen Aufbau von F.

Induktionsschluss

Es sei $F = \forall x H$, H in Skolemform und H hat $n-1$ Allquantoren. Nach Vorraussetzung gilt: $\mathcal{A} \models \forall x H$, d.h. für alle $d \in U_A$ gilt: $\mathcal{A}_{[x/d]}(H) = 1 \quad (*)$.

Nun sei $t \in D(F)$

Für $d := \mathcal{A}(t)$ gilt nach $(*)$
$$\mathcal{A}[x/\mathcal{A}(t)](H) = 1.$$
Nach Überführungssatz gilt
$$\mathcal{A}(H[x/t]) = 1.$$
Nach Induktionsvorraussetzung gilt: $\mathcal{B}(H[x/t]) = 1$.

Nach Überführungssatz

$$\mathcal{B}_{[x/\mathcal{B}(t)]}(H) = 1.$$

Wegen $\mathcal{B}(t) = t$ folgt

$$\mathcal{B}_{[x/t]}(H) = 1.$$

Insgesamt gilt: $\mathcal{B}_{[x/t]}(H) = 1$ für alle $t \in D(F)$. Folglich: $\mathcal{B}(\forall x H) = 1$.

Korollar 2.3.1 (Löwenheim-Skolem)
Jede erfüllbare Formel ohne Gleichheitszeichen besitzt ein Modell mit einem abzählbaren Universum.

Bemerkung

Die Ausssage

$$F := 2 + 3 = 1 + 4$$

ist im Standarmodell $\mathbb{N}$ der natürlichen Zahlen wahr.

Ist $\mathcal{B}$ die Herbrand-Struktur von F, so gilt F **nicht** in $\mathcal{B}$, denn die Terme $2 + 3, 1 + 4$ sind verschiedene.

Definition 2.3.3 (Herbrand-Expansion)
Es sei $F = \forall y_1, \dots \forall y_n F^*$ eine Aussage in Skolemform. Die Menge

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

*heißt **Herbrand-Expansion**.*

Beispiel

$$\begin{aligned} F &= \forall \forall y \forall z P(x, f(y), g(y, z)) \\ F^* &= P(x, f(y), g(y, z)) \\ P(x, f(y), g(y, z))[x/a][y/f(a)][z/g(a, a)] &= P(a, f(f(a)), g(f(a), g(a, a))) \end{aligned}$$

Definition 2.3.4
Es sei F^* eine quantorenfreie Formel ohne Gleichheitszeichen. Nun ersetzen wir jedes Vorkommen einer atomaren Teilformel von F^* durch eine Aussagenvariable, wobei gleich atomare Teilformeln durch gleiche Aussagenvariablen und verschiedene atomare Teilformeln durch verschiedene Aussagenvariablen ersetzt werden. Diese aussagenlogische Formel wird mit $A(F^*)$ bezeichnet.

$$F^* P(f(x), g(x, y)) \vee \neg Q(x, h(x))$$

$$A(F^*) = A_0 \vee \neg A_1$$

Lemma 2.3.1
Ist F^* eine Aussage der Prädikatenlogik, besitzt F^* keinen Quantor und enthält F^* nicht das Gleichheitssymbol, so ist F^* genau dann erfüllbar, wenn $A(F^*)$ im aussagenlogischen Sinne erfüllbar ist.

Beweis:

„⇒“ Es sei $\mathcal{A}$ ein Modell von F^*. Ist der atomare Formel $P(t_1, \dots, t_n)$ in F^* die Aussagenvariable A_i zugeordnet, so setze

$$\mathcal{A}(A_1) = 1 \Leftrightarrow_{Df} \mathcal{A}(P(t_1 1, \dots, t_n)) = 1.$$

Nach Konstruktion erfüllt Θ die Formel $A(F^*)$.

„⇐“ Es sei Θ eine Belegung, die $A(F^*)$ wahr macht.

Idee: Herbrand-Modell $\mathcal{B}$.

$U_{\mathcal{B}} := D(F^*)$

$f^{\mathcal{B}}$ (siehe Konstruktion Herbrand-Struktur)

Das n-stellige Prädikat P komme in F vor. Ferner seien $t_1, \dots, t_n \in D(F^*)$.

$P^{\mathcal{B}}(t_1, \dots, t_n)$ gdw.$_{Df}\Theta(A_i) = 1$, wobei A_i der Form $P(t_1, \dots, t_n)$ zugeordnet ist. Nach Konstruktion $\mathcal{A}(F^*) = 1$.

Bemerkung:

Satz wird falsch, falls Gleicheitszeichen zugelassen.

$$F^* = P(a) \land \neg P(b) \land (a = b)$$

F^* unerfüllbar.

$$A(F^*) = A_0 \land \neg A_1 \land A_2$$

ist erfüllbar.

Satz 2.3.2 (Godel, Herbrand, Skolem)
Eine Aussage F in Skolemform ohne Gleichheitszeichen ist genau dann erfüllbar, wenn $E(F)$ im aussagenlogischen Sinne erfüllbar ist.

Beweis:

Satz 2.3.1 besagt: F erfüllbar gdw. F ein Herbrand-Modell besitzt.

Es reicht aus zu zeigen: F besitzt Herbrand-Modell von $\mathcal{A}$ geanu dann, wenn $E(F)$ im aussagenlogischen Sinne erfüllbar ist.

Es sei $F = \forall y_1 \dots \forall y_n F^*$.

Dann $\mathcal{A}$ ist ein Herbrand-Modell von F genau dann, wenn für alle $t_1, \dots, t_n \in D(F)$ gilt

$$\mathcal{A}_{[y_1/t_1]\dots[y_n/t_n]}(F^*) = 1$$

gdw. (Überführungssatz) für alle $t_1, \dots t_n \in D(F)$ gilt

$$\mathcal{A}(F^*[y_1/t_1] \dots [y_n/t_n]) = 1$$

gdw. für alle $G \in E(F)$ gilt

$$\mathcal{A}(G) = 1$$

gdw. für alle $G \in E(F)$ gilt

$$\Theta(G) = 1$$

gdw. $E(F)$ im aussagenlogischen Sinne erfüllbar

Korollar 2.3.2 (Herbrand)
Ist F eine Aussage in Skolemform ohne Gleichheitszeichen, so ist F unerfüllbar genau dann, wenn endlich viele Formeln der Herbrand-Expansion $E(F)$ existieren, die im aussagenlogischen Sinne unerfüllbar sind.

Beweis: Kompaktheitssatz

Definition 2.3.5
*Ist U eine nichtleere Menge und $P \subseteq U$, so heißt (P, U) ein **Entscheidungsproblem**. Ist $x \in U$, so entscheide, ob $x \in P$.*

*Ein **Entscheidungsverfahren** für (P, U) ist ein Algorithmus, der angesetzt auf ein $x \in U$ nach endlich vielen Schritten stoppt und die Frage $x \in P$ mit ja oder nein beantwortet. Ein **Semi-Entscheidungsverfahren** für (P, U) ist ein Algorithmus der angesetzt auf $x \in U$ genau dann, nach endlich vielen Schritten, stoppt, wenn er die Frage $x \in P$ mit ja beantwortet.*

2.3.1 Unerfüllbarkeitstest von Gilmore

Eingabe: Eine prädikatenlogische Aussage F in Skolemform ohne Gleichheitszeichen.

Es sei $F_1, F_2, F_3, \ldots$ eine beliebige Aufzählung von $E(F)$.

`repeat` $n := n + 1$

`until` $(F_1 \wedge \cdots \wedge F_n)$ ist unerfüllbar (Wahrheitstafelverfahren)

Gib „unerfüllbar“ und stopp.

Ist L eine Sprache der Prädikatenlogik ohne Gleichheitszeichen, U die Menge der Aussagen in L in Skolemform, so sei P die Menge der unerfüllbaren Formeln von U. Es ist (P, U) ein Entscheidungsproblem. Der Gilmore-Test ist ein Semi-Entscheidungsverfahren für die Unerfüllbarkeit einer Formel.

3 Prädikatenlogische Resolution

Definition 3.0.6
Sind $y_1, \dots, y_n$ Variablen und sind $t_1, \dots, t_n$ variablenfreie Terme, so heißt eine Substitution, die die Variablen $y_1, \dots, y_n$ durch $t_1, \dots, t_n$ ersetzt, eine ***Grundsubstitution****.*

Ist $F = \forall y_1, \dots, \forall y_n F^*$ eine Aussage in Skolemform, so sind die Substitutionen, die in $E(F)$ vorkommen, Grundsubstitutionen. Es heißt $F^*[y_1/t_1] \dots [y_n/t_n]$ ***Grundinstanz von*** *F^*. Geht G aus F durch Substitution der Variablen von F durch Terme hervor, die nicht notwendigerweise variablenfrei sind, so heißt G eine* ***Instanz von*** *F.*

Es sei $F = \forall y_1 \dots \forall y_n F^*$ Aussage in Skloemform ohne Gleichheitszeichen. Ferner sei $F_1, F_2, F_3, \dots$ eine Aufzählung von E(F).

3.1 Grundresolutionsalgorithmen

Eingabe: F Aussage in Skolemform ohne Gleicheitszeichen, $F = \forall y_1 \dots \forall y_n F^*$, F^* Matrix von F in KNF. $i := 0$, $M := \emptyset$

`repeat` $i := i + 1$, $M = M \cap \{F_i\}$, $M = Res^*(M)$

`until` $\square \in M$

Gib „unerfüllbar“ aus und stopp.

Beispiel:

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

$D(F) = \{a, f(a), f(f(a)), f^3(a), \dots\}$

Matrix von F: $P(x) \wedge \neg P(f(x))$

Klauselform: $\{\{P(x)\}, \{\neg P(f(x))\}\}$

$E(F) = \{P(a) \wedge \neg P(f(a)), P(f(a)) \wedge \neg P(f^2(a)), \dots\}$

$$\begin{aligned} F_1 &= P(a) \wedge \neg P(f(a)) \\ F_2 &= P(f(a)) \wedge \neg P(f(f(a))) \\ &\vdots \end{aligned}$$

Klauselform

$$\big\{\{\{P(a)\}, \{\neg P(f(a))\}\}, \{\{P(f(a))\}, \{\neg P(f^2(a))\}\}\big\}$$

Daraus lässt sich die leere Klausel resolvieren, damit ist die Formel unerfüllbar.

Klauseln in F^*	$\{P(x)\}$	$\neg P(f(x))$
Grundsubstitution	$[x/f(a)]$	$[x/a]$
Grundinstanzen	$\{P(f(a))\}$	$\{\neg P(f(a))\}$

$\square$

Beispiel:

$F = \forall x \forall y((\neg P(x) \vee \neg P(f(a)) \vee Q(y)) \wedge P(y) \wedge (\neg P(y(b,x)) \vee \neg Q(b)))$

$F^* = \{\{\neg P(x), \neg P(f(a)), Q(y)\}, \{P(y)\}, \{\neg P(g(b,x)), \neg Q(b)\}\}$

Satz 3.1.1 (Grundresolutionssatz)
Ist $F = \forall y_1 \ldots \forall y_n F^*$ eine Aussage in Skolemform ohne Gleichheitszeichen und einer Matrix F^* in KNF, so ist F genau dann unerfüllbar, wenn es eine Folge $K_1, K_2, \ldots, K_n$ gibt, mit folgenden Eigenschaften:

1. *K_n ist die leere Klausel*
2. *Entweder ist K_i eine Grundinstanz einer Klausel $K \in F^*$, d.h.*

$$K_i = K[y_1/t_1] \ldots [y_n/t_n]$$

für geeignete $t_1, \ldots, t_n \in D(F)$ oder K_i ist eine aussagenlogische Resolvente zweier Klauseln K_l, K_s mit $l, s \in \mathbb{N}$ und $l, s < i$.

Definition 3.1.1
Ein ***Literal*** *einer prädikatenlogischen Formel in KNF ist eine atomare Formel oder die Negation einer atomaren Formel.*

Definition 3.1.2

1. *Die Ersetzung von Variablen durch Terme nennt man* ***Substitution****.*

2. *Sind $t, t_1, \dots, t_n$ Terme und $y_1, \dots, y_n$ Variablen und ist $sub = [y_1/t_1] \dots [y_n/t_n]$, so sei $t\ sub$ derjenige Term, der durch Anwendung der Substitution sub auf den Term t aus t hervorgeht.*

3. *Es sei $\mathbb{L} = \{L_1, \dots, L_n\}$ eine Menge prädikatenlogischer Literale. Ferner sei sub eine Substitution. sub heißt* ***Unifikation*** *von $\mathbb{L}$, falls $L_1\ sub = \dots = L_n\ sub$.*

   Man schreibt: $|\mathbb{L}\ sub| = 1$. Man sagt: sub unifiziert $\mathbb{L}$

4. *Ein Unifikator sub von $\mathbb{L}$ heißt* ***allgemeinster Unifikator*** *von $\mathbb{L}$, falls zu jedem Unifikator sub' von $\mathbb{L}$ eine Substitution s existiert mit $sub' = sub\ s$*

   [$sub\ s$ heißt: erst sub dann s]

Satz 3.1.2 (Unifikationssatz)
Jede Menge von Literalen besitzt einen allgemeinsten Unifikator.

Beweis:

Unifikationsalgorithmus

Eingabe: Eine nichtleere Literalmenge $\mathbb{L}$ `sub := [ ]` (leere Substitution)

`while` $|\mathbb{L}$ `sub`$| > 1$ `do`

`begin`

Durchsuche die Literale in $\mathbb{L}$ `sub` von links nach rechts, bis die erste Position gefunden ist, wo sich mindestens zwei Literale, sagen wir L_1 und L_2, in den vorkommenden Zeichen unterscheiden.

`if` keines der Zeichen ist eine Variable `then`

stoppe mit der Ausgabe „nicht unifizierbar"

`else`

`begin`

Sei x die Variable und t sei der im anderen Literal beginnende Term;

`if` x kommt in t vor `then`

stoppe mit der Ausgabe „nicht unifizierbar"

`else sub` := `sub` [x/t]

`end`

`end`

Ausgabe: `sub` ist allgemeinster Unifikator.

Korrektheit des Unifikationsalgorithmus

Wirft der Algorithmus „nicht unifizierbar" aus, so stoppt. Anderenfalls wird in der `while`-Schleife eine Variable x aus $\mathbb{L}$ `sub` ersetzt durch einen Term t, der x nicht enthält. Daher durchläuft der Algorithmus die `while`-Schleife höchstens sooft, wie Variablen in $\mathbb{L}$ vorkommen. Also stoppt der Algorithmus.

Ist $\mathbb{L}$ nicht unifizierbar, so kann der Algorithmus **nicht** erfolgreich durchlaufen werden, **somit stoppt er mit der Ausgabe „nicht unifizierbar"**

Nun stoppe der Algorithmus mit der Ausgabe „nicht unifizierbar".

1. Fall Keines der Zeichen ist eine Variable. Dann sind es Terme der Art $f(t_1, \ldots, t_n), g(t_1^*, \ldots, t_n^*)$, die natürlich nicht unifizierbar sind.

2. Fall Das eine Zeichen ist ein Variable x und das andere Zeichen ist der Beginn eines Terms t, der x enthält und verschieden von x ist.

Es sei s irgendein Term. Dann gilt: $x[x/s] \;\neq\; t[x/s]$, d.h. $s \neq t[x/s]$

Behauptung:

Ist $\mathbb{L}$ unifizierbar, so liefert der Algorithmus einen allgemeinsten Unifikator von $\mathbb{L}$.

Da $\mathbb{L}$ unifizierbar, wird der Algorithmus erfolgreich beendet. Er möge die Substitution sub produzieren.

Nach dem i-ten `while`-Schleifendurchlauf sie die Substitution sub_i produziert.

Behauptung:

Ist $\mathbb{L}$ unifizierbar, so gilt für jedes i: Ist sub' ein Unifikator für $\mathbb{L}$, so existiert eine Substitution s_i mit

$$sub' = sub_i \; s_i.$$

Wird der letzte Schleifendurchlauf erfolgreich beendet, so ist $sub = sub_i$.

Induktionsbeweis über i

i=0

Es ist $sub_0 = [\;]$ und $sub' = [\;]\; sub' = sub_0 \; sub'$

i>0

Nach Induktionsvoraussetzung existiert Substitution s_{i-1} mit

$$(1) \qquad sub' = sub_{i-1} \; s_{i-1}$$

Ist $|\mathbb{L} \; sub_{i-1}| = 1$, so fertig.

Daher sei $|\mathbb{L} \; sub_{i-1}| > 1$. Der Algorithmus ersetzt eine Variable x durch einen Term t, der x nicht enthält. Wegen (1) ist

$$(2) \qquad x \; s_{i-1} = t \; s_{i-1},$$

denn sub' unifiziert $\mathbb{L}$. Andererseits liefert der Algorithmus

$$(3) \qquad sub_i = sub_{i-1} \; [x/t]$$

$$\begin{aligned} sub_i \; s_i &\overset{(3)}{=} sub_{i-1}[x/t] \; s_i \\ &= sub_{i-1} \; s_i[x/t \; s_i] \quad x \text{ wird in } s_i \text{ nicht ersetzt} \end{aligned}$$

$$= sub_{i-1}\ s_i[x/t\ s_{i-1}] \quad x \text{ kommt in } t \text{ nicht vor}$$
$$\overset{(2)}{=} sub_{i-1}\ s_i[x/x\ s_{i-1}] \quad t\ s_{i-1} = x\ s_{i-1}$$
$$= sub_{i-1}\ s_{i-1} = sub'$$

s_i ist eine Restriktion von s_{i-1} derart, dass x durch s_i nicht ersetzt wird. Alle anderen Substitutionen bleiben.

Beispiel

$$\mathbb{L} = \{\neg P(f(t, g(a, y)), h(z)), negP(f(f(u, v), w), h(f(a, b)))\}$$

1. Schritt

$$Subst\ [z/f(u, v)]$$

2.Schritt

$$\neg P(f(f(u, v)), g(a, y)), h(f(u, v)))$$
$$\neg P(f(f(u, v), w), h(f(a, b))$$
$$[z/f(u, v)][w/g(a, y)]$$

3. Schritt

$$\neg P(f(f(u, v), g(a, y)), h(f(u, v)))$$
$$\neg P(f(f(u, v), g(a, y)), h(f(a, b)))$$

$$[z/f(u, v)][w/g(a, y)][u/a]$$

4. Schritt

$$\neg P(f(f(a, v), g(a, y)), h(f(a, v)))$$
$$\neg P(f(f(a, v), g(a, y)), h(f(a, b)))$$
$$[z/f(u, v)][w/g(a, y)][u/a][v/b]$$

5. Schritt

$$\neg P(f(f(a, b), g(a, y)), h(f(a, b)))$$
$$\neg P(f(f(a, b), g(a, y)), h(f(a, b)))$$

$$sub = [z/f(u, v)][w/g(a, y)][u/a][v/b]$$

ist allgemeinster Unifikator

Definition 3.1.3 (Prädikatenlogische Resolution)
Es seien K_1, K_2 und R prädikatenlogische Klauseln. Es ist R eine prädikatenlogische Resolution von K_1 und K_2, falls folgendes gilt:

1. *Es existieren Substitutionen s_1, s_2, so dass $K_1\ s_1, K_2\ s_2$ keine gemeinsamen Variablen haben.*

2. *Es existiert eine Menge $\{L_1, \ldots, L_m\}$ mit $L_1, \ldots, L_m \in K_1\ s_1$ $(m \geq 1)$ und es existiert eine Menge von Literalen $L_1', \ldots, L_n \in K_2\ s_2$ $(n \geq 1)$ derart, dass*

$$\mathbb{L} = \{\overline{L}_1, \ldots, \overline{L}_m, L_1', \ldots, L_n'\}$$

   unifizierbar ist.

3. *Es existiert ein allgemeinster Unifikator sub derart, dass*

$$R = K_1\ s_1 \backslash \{L_1, \ldots, L_m\} \cup K_2\ s_2 \backslash \{L_1', \ldots, L_n'\})\ sub$$

   Für diesen Sachverhalt schreibe:

   Beispiel:

Definition 3.1.4
Es sei F eine prädikatenlogische Klauselmenge. Dann sei

1. $Res(F) = F \cup \{R : R$ *ist prädikatenlogischer Resolvent zweier Klauseln* $K_1, K_2 \in F\}$

2. $Res^0(F) = F$

   $Res1n + 1(F) = Res(Res^n(F))$

   $R^* := \bigcup\{Res^n(F) : n \in \mathbb{N}\}$

Aus der Definition folgt:

Lemma 3.1.1
Es ist $\square \in Res^*(F)$ genau dann, wenn eine endliche Folge $(K_1, \ldots, K_n)$ von Klauseln existiert, so dass gilt:

1. $K_n = \square$

2. *Entweder ist $K_i \in F$ oder es existieren Klauseln K_a, K_b mit $a, b \in \mathbb{N}$ und $a, b < i$, so dass K_i ein prädikatenlogischer Resolvent von K_a, K_b ist.*

Man sagt: $(K_1, \ldots, K_n)$ ist eine Resolutionsherleitung von $\square$.

Definition 3.1.5
Ist F eine Formel, so bezeichne $Sk(F)$ eine Skolemform von F mit einer Matrix F^* in KNF und $Kl(F)$ bezeichne die Klauselmenge von F^*

Definition 3.1.6
Ist Γ eine Menge von Formeln, so heißt F aus Γ ***herleitbar****, geschrieben $\Gamma| - F$, falls die Formeln $F_1, \ldots, F_n \in \Gamma$ existieren mit*

$$\square \in Res^*(Kl(F_1 \wedge \cdots \wedge F_n \wedge \neg F))$$

$F_1, \ldots, F_n| - F$

gdw. $F_1 \wedge \cdots \wedge F_n \rightarrow F$ allgemeingültig.

gdw. $F_1 \wedge \cdots \wedge F_n \wedge \neg F$ unerfüllbar

gdw $\square \in Res^*(Kl(F_1 \wedge \cdots \wedge F_n \wedge \neg))$

Beispiel $\Gamma^* = \{\forall x \forall y(P(x, y, c) \rightarrow R(y, g(f(x)))), \forall x \forall y P(f(x), y, c)\}$

$\Gamma = \{\forall x \forall y(\neg P(x, y, c) \vee R(y, g(f(x)))), \forall x \forall y P(f(x), y, c)\}$

Frage: $\Gamma| - \exists x \exists y R(f(x), g(y))$?

Setze

$F_1 := \forall x \forall y(\neg P(x, y, c) \vee R(y, g(f(x)))$

$F_2 := \forall x \forall y P(f(x), y, c)$

$\neg F := \forall x \forall y \neg R(f(x), g(y))$

$\square \in Res^*(Kl(F_1 \wedge F_2 \wedge \neg F))$?

$K_1 = \{\neg P(x, y, c), R(y, g(f(x)))\}$

$K_2 = \{P(f(x), y, c)\}$

$K_3 = \neg R(f(x), g(y))\}$

$\square \in Res^*(K_1, K_2, K_3)$?

$\{\neg P(x, y, c), R(y, g(f(x)))\}$ $\{\neg R(f(x), g(y))\}$

$K_1' = K_1[x/u][y/v]$

$\{\neg P(u, v, c), R(v, g(f(u))\}$ $\{\neg R(f(x), g(y))\}$

Allgemeiner Unifikator

$sub_1 = [v/f(x)][y/f(u)]$

$K_4 = \{\neg P(u, f(x), c)\}$ Resolvente

$\{P(f(x), y, c)\}$ $\{\neg P(u, f(x), c)\}$

$K_2' = K_2[x/z]$

$\{P(f(z), y, c)\}$ $\{\neg P(u, f(x), c)\}$

Allgemeinster Unifikator

$sub_2 = [u/f(z)][y/f(x)]$

$\{P(f(z), f(x), c)\}$ $\{\neg P(f(z), f(x), c)\}$

Setze $sub := sub_1 sub_2$. Dann ist $R(f(x), g(y))sub = R(f(x), g(f(f(z))))$

Man kann zeigen: $\Gamma| - R(f(x), g(f(f(z))))$

Für das Existenzproblem $\Gamma| - \exists x \exists y R(f(x), g(y))$ hat man x *sub* und y *sub* als Lösungsschar, d.h. „x und $f(f(z))$“ Ist z.B. $x = z = c$ so gilt $\Gamma| - R(f(c), g(f(f(c))))$

Lemma 3.1.2 (Lifting-Lemma)
Es seien K_1, K_2 zwei prädikatenlogische Klauseln und K_1', K_2' Grundinstanzen von K_1, K_2, so dass K_1', K_2' resolvierbar im aussagenlogischen Sinne sind. Nun sei R' ein aussagenlogischer Resolvent von K_1' und K_2'. Dann sind K_1, K_2 prädikatenlogisch resolvierbar und es existiert ein prädikatenlogischer Resolvent R von K_1 und K_2 derart, dass R' eine Grundinstanz von R ist.

Beweis

Es seien s_1, s_2 Variablenumbenennungen, so dass $Var(K_1 s_1) \cap Var(K_2 s_2) = \emptyset$. Es sind nach Voraussetzung K_1', K_2' Grundinstanzen von K_1, K_2. Somit sind auch K_1', K_2' Grundinstanzen von $K_1 s_1, K_2 s_2$. Es seien sub_1, sub_2 Grundsubstitution mit

$$K_1' = K_1 s_1 sub_1, K_2' = K_2 s_2 sub_2.$$

Da sub_1, sub_2 Grundsubstitutionen sind, ist auch

$$sub := sub_1 sub_2$$

eine Grundsubstitution.

Es gilt

1. $K_1' = K_1 s_1 sub, K_2' = K_2 s_2 sub$

   Es sei R' ein aussagenlogischer Resolvent von K_1', K_2'. Dann existiert ein Literal $L \in K_1'$ mit $\overline{L} \in K_2'$ und $R' = (K_1 \backslash \{L\}) \cup (K_2' \backslash \{\overline{L}\})$.

   Das Literal L gehe aus den Literalen $L_1, \ldots, L_m \in K_1 s_1$ unter Anwendung von *sub* hervor. Dann existieren Literale $L_1', \ldots, L_n' \in K_2 s_2$, so dass das Literal L durch Anwendung von *sub* auf $L_1', \ldots, L_n'$ hervorgeht, d.h.

   $$\overline{L} = L_1' \ sub = \cdots = L_n' \ sub.$$

   Daher unifiziert *sub* die Menge

   $$\mathbb{L} := \{L_1, \ldots, L_m, \overline{L}_1', \ldots, \overline{L}_m'\}$$

   Ist sub_0 der allgemeinste Unifikator von $\mathbb{L}$, so sei

   $$R = ((K_1 s_1 \backslash \{L_1, \ldots, L_m\}) \cup (K_2 s_2 \backslash \{L_1', \ldots, L_m'\})) sub_0$$

   Da sub_0 allgemeinster Unifikator ist, existiert eine Substitution s mit $sub = sub_0 \ s$. Da *sub* eine Grundsubstitution ist, ist s eine Grundsubstitution. Es gilt

   $$\begin{aligned} R' &= (K_1' \backslash \{L\}) \cup (K_2' \backslash \{\overline{L}\}) \\ &= (K_1 s_1 sub \backslash \{L\}) \cup (K_2 s_2 sub \backslash \{\overline{L}\}) \\ &= ((K_1 s_1 \backslash \{L_1, \ldots, L_m\}) \cup (K_2 s_2 \backslash \{L_1', \ldots, L_m'\})) sub \\ &= ((K_1 s_1 \backslash \{L_1, \ldots, L_m\}) \cup (K_2 s_2 \backslash \{L_1', \ldots, L_m'\})) sub_0 s \\ &= Rs \end{aligned}$$

   Da s Grundsubtitution ist R' Grundinstanz.

Satz 3.1.3 (Resolutionssatz der Prädikatenlogik)
Es sei F eine Aussage in Skolemform ohne Gleichheitszeichen, mit Matrix F^* in KNF. Dazu gilt: F ist unerfüllbar genau dann, wenn $\Box \in Res^*(Kl(F))$.

Beweis:

„⇐“ Korrektheit

Es gelte: $\Box \in Res^*(Kl(F))$.

Ist H irgendeine Formel mit freien Variablen $x_1, \ldots, x_n$, so heißt $\forall x_1 \ldots \forall x_n H$ der **Allabschluss** von H. Es gilt nach Satz 2.2.2

$$F \equiv \bigwedge_{K \in F^*} \forall K$$

Behauptung 1: Sind K_1, K_2 irgendwelche Klauseln und ist R ein Resolvent von K_1 und K_2, so folgt $\forall R$ aus $\forall K_1, \forall K_2$; d.h.

$$\forall K_1 \wedge \forall K_2 \rightarrow \forall R$$

ist allgemeingültig. Es sei $\mathcal{A}$ eine Struktur mit

$$\mathcal{A}(\forall K_1) = 1 = \mathcal{A}(\forall K_2)$$

Zu Zeigen: $\mathcal{A}(\forall R) = 1$

Annahme: $\mathcal{A}(\forall R) = 0$

Durch geeignete Interpretation der freien Variablen von R existiert eine Struktur $\mathcal{A}'(R) = 0$. Es ist

$$R = (K_1 s_1 \backslash \{L_1, \ldots, L_m\}) \cup \left(K_2 s_2 \backslash \{L'_1, \ldots, L'_n\}\right) \ sub,$$

wobei *sub* der allgemeinste Unifikator von $\mathbb{L} = \{L_1, \ldots, L_m, \overline{L'}_1, \ldots, \overline{L'}_n\}$ ist. Es ist

$$L := L_1 sub = \cdots = L_m sub = \overline{L'}_1 sub = \cdots = \overline{L'}_n sub$$

Ferner ist

$$(1)\ \mathcal{A}'(K_1 s_1 sub \backslash \{L\}) = 0 = \mathcal{A}'(K_2 s_2 sub \backslash \{\overline{L}\})$$

wegen $\mathcal{A}(\forall K_1) = 1 = \mathcal{A}(\forall K_2)$ ist

$$(2)\ \mathcal{A}'(K_1 s_1 sub) = 1 = \mathcal{A}'(K_2 s_2 sub)$$

Folglich ist $\mathcal{A}'(L) = 1 = \mathcal{A}'(\overline{L})$

Widerspruch!

Behauptung 2: F ist unerfüllbar

Annahme: F ist erfüllbar.

Es sei $\mathcal{A}$ ein Modell von F. Wegen

$$F \equiv \bigwedge_{K \in F^*} \forall K$$

gilt

$$\mathcal{A}(\forall K) = 1$$

für alle $K \in F^*$. Nach Vorraussetzung $\square \in Res^*(Kl(F))$. Dann gilt: $Res(R, \overline{R}) = \square$. Nach Behauptung 1 gilt: $\mathcal{A}(\forall R) = 1 = \mathcal{A}(\forall \overline{R})$. Sind irgendwelche Belegungen von R gegeben, so erhält man Modell $\mathcal{A}'$ mit $\mathcal{A}'(R) = 1 A'(\overline{R})$

Widerspruch!

„⇒“ Vollständigkeit

Es sei F unerfüllbar. Nach Grundresolutionssatz existiert eine Folge $(K'_1, \ldots, K'_n)$ mit folgenden Eigenschaften:

1. $K'_n = \square$
2. Entweder ist K'_i eine Grundinstanz einer Klausel aus F^* oder es existieren die Grundinstanzen K'_a, K'_b mit $a, b \in \mathbb{N}; a, b < i$, so dass K'_i ein aussagenlogischer Resolvent von K'_a, K'_b. Mit Hilfe des Liftinglemmas hebe diese aussagenlogische Resolutionsherleitung in die Prädikatenlogik zurück.

4 Hornklauselprogramme

Definition 4.0.7

1. *Eine* ***Tatsachenklausel*** *hat die Form* $\{P\}$*, wobei* P *ein Prädikatssymbol ist.*

2. *Eine* ***Prozedurklausel*** *hat die Form* $\{P, \neg Q_1, \ldots, \neg Q_k\}$*, wobei* $P, Q_1, \ldots, Q_k$ *Prädikatensymbole sind*

3. *Ein* ***Hornklauselprogramm*** *oder* ***Logik-Programm*** *ist eine Menge von Tatsachen- und Prozedurklauseln.*

4. *Ein Logik-Programm wird durch eine* ***Zielklausel*** *aufgerufen. Sie hat die Form* $\{\neg Q_1, \neg Q_2, \ldots, \neg Q_k\}$.

Eine ***SLD-Resolutionsherleitung***[1] *startet mit der Zielklausel, wählt eine Tatsachen- oder Prozedurklausel an und resolviert. Der Resolvent ist die neue Zielklausel. Die für die Resolvierung notwendigen Substitutionen werden bei den Programmklauseln und* ***nicht*** *bei den Zielklauseln vorgenommen.*

Beispiel: (Addition)

$$\begin{aligned} x + 0 &= x \\ x + y' &= (x+y)' \end{aligned}$$

s (=successor) steht für die Nachfolgefunktion, d.h. statt x' schreibe $s(x)$

$$\begin{aligned} & x + 0 = x \\ & x + y = z \to x + s(y) = s(z) \end{aligned}$$

Leider = nicht zugelassen. A sei 3-stelliges Prädikat $A(x, 0, x)$. $A(x, y, z) \to A(x, s(y), s(z))$.

Klauseln

$$\left.\begin{array}{l} \{A(x, 0, x)\} \\ \{A(x, s(y), s(z), \neg A(x, y, z)\} \end{array}\right\}\text{Logikprogramm}$$

$\{\neg A(s(s(s(0))), s(s(s(0)), u)\}$ Zielklauseln

$K_1 =$ $\{\neg A(s(s(s(0))), s(s(s(0)), u)\}$

$K_2 =$ $\{A(x, 0, x)\}$

$K_3 =$ $\{A(x, s(y), s(z)), \neg A(x, y, z)\}$

[1]**l**inear resolution with **s**election function for **d**efinite clauses

$K_4 = Res(K_1, K_3) = \{\neg A(s(s(s(0))), s(0), z)\}$ mit $sub_1 = [x/s(s(s(0)))][y/s(0)][u/s(z)]$

$K_5 = Res(K_4, K_2[z/z']) = \{\neg A(s(s(s(0))), 0, z')$ mit $sub_2 = [x/s(s(s(0)))][y/0][z/s(z')]$

$K_6 = Res(K_5, K_1) = \square$ mit $sub_3 = [x/s(s(s(0)))][z'/s(s(s(0)))]$

Antwort: $u\ sub_1\ sub_2\ sub_3 = s(z)\ sub_2\ sub_3 = s(s(z'))\ sub_3 = s(s(s(s(s(0))))) = 5$