051013 VO Theoretische Informatik

Aussagenlogik: Resolutionskalkül, Hornlogik, Einheitsresolution
Prädikatenlogik: Syntax & Semantik

Ekaterina Fokina

universität wien

Widerholung

Zur Erinnerung: In der formalen Logik müssen wir alle Bestandteile eines logischen Systems genau definieren.

Letzte Woche:

- **Formale Syntax**: Wir haben definiert, was genau eine Aussagenlogische Formel ist.
- **Formale Semantik**: Wir haben aussagenlogischen Formeln eine Bedeutung gegeben.
- Formalisieren in der Aussagenlogik.
- Grundbegriffe der Aussagenlogik.

Heute in der Vorlesung:

- Kalkül & Schlussregeln für die Aussagenlogik
- Hornlogik und Einheitsresolution
- Prädikatenlogik: Syntax, Semantik, Grundbegriffe

Grundlegende Begriffe - Zusammenfassung

α: Belegung für *Var*,
F,G: Formeln über *Var*
$\mathcal{F}$: Eine Menge von Formeln über *Var*

Definition

- α ist ein **Modell von** F wenn $\widehat{\alpha}(F) = 1$, wir schreiben auch $\alpha \models F$.
- F ist **erfüllbar** wenn F mind. ein Modell hat (sonst **unerfüllbar**).
- F ist **allgemein gültig** / eine **Tautologie** wenn jede Belegung auch ein Modell ist.
- F ist **semantisch äquivalent** zu G wenn $\widehat{\alpha}(F) = \widehat{\alpha}(G)$ für alle Belegungen α. Wir schreiben $F \equiv G$.
- G **folgt** aus $F/\mathcal{F}$ wenn jedes Modell von $F/\mathcal{F}$ auch Modell von G ist. Wir schreiben $F \models G$ bzw. $\mathcal{F} \models G$.

Grundlegende Begriffe - Zusammenfassung

α: Belegung für *Var*,
F,G: Formeln über *Var*
$\mathcal{F}$: Eine Menge von Formeln über *Var*

Definition

- α ist ein **Modell von** $\mathcal{F}$ wenn $\widehat{\alpha}(G) = 1$ für jede Formel $G \in \mathcal{F}$, wir schreiben auch $\alpha \models \mathcal{F}$.
- $\mathcal{F}$ ist **konsistent** (**widerspruchsfrei**) wenn es ein Modell für $\mathcal{F}$ gibt. $\mathcal{F}$ ist **inkonsistent**(**widersprüchlich**) wenn es kein Modell für $\mathcal{F}$ gibt.

Fundamentale Sätze

Satz

Ein Formel F ist genau dann eine Tautologie wenn $\neg F$ unerfüllbar ist.

Beispiel: Tautologie: $(c \vee \neg c)$
unerfüllbar: $\neg(c \vee \neg c) \equiv (\neg c \wedge c)$

Satz

Zwei Formeln F, G sind semantisch äquivalent genau dann wenn F aus G folgt und G aus F folgt.

- $F \equiv G \Leftrightarrow G \models F$ *und* $F \models G$

Satz

Zwei Formeln F, G sind semantisch äquivalent genau dann wenn $F \leftrightarrow G$ eine Tautologie ist

- $F \equiv G \Leftrightarrow \models F \leftrightarrow G$

Es gilt auch:

- $F \models G \Leftrightarrow \models F \rightarrow G$
- $\{F, G\} \models H \Leftrightarrow F \models G \rightarrow H$
- $\{F, G\} \models H \Leftrightarrow \models (F \wedge G) \rightarrow H$

Satz (Kompaktheitssatz)

Eine (unendliche) Menge von aussagenlogischen Formeln ist genau dann erfüllbar, wenn jede endliche Teilmenge erfüllbar ist.

Beobachtung:

- Jede Teilformel kann durch eine semantisch äquivalente Formel ersetzt werden.

Satz (Ersetzungssatz)

Betrachtet man eine Formel G mit einer Teilformel F_1 und bildet eine neue Formel $G[F_1/F_2]$ indem man (in G) die Teilformel F_1 durch eine semantisch äquivalente Formel F_2 ersetzt, dann gilt $G \equiv G[F_1/F_2]$.

Beispiel: $G = ((\neg a \vee b) \vee c)$

$F_1 = (\neg a \vee b)$ ist semantisch äquivalent zu $F_2 = (a \rightarrow b)$

$G[F_1/F_2] = ((a \rightarrow b) \vee c)$

Ersetzungssatz: $(\neg a \vee b \vee c) \equiv ((a \rightarrow b) \vee c)$

Äquivalenzen der Aussagenlogik

Mit dem Ersetzungssatz kann man äquivalente Formeln verwenden um eine Formel zu vereinfachen. Einige hilfreiche Äquivalenzen:

Idempotenz

- $(F \wedge F) \equiv F$
- $(F \vee F) \equiv F$

Kommutativität

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

Assoziativität

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

Absorption

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

Distributivität

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

Doppelnegation

- $\neg\neg F \equiv F$

de Morgansche Regeln

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

Subsection 10

Schlussregeln der Aussagenlogik

Der Resolutionskalkül der Aussagenlogik

- Resolution ist ein Verfahren der formalen Logik, um zu Testen ob eine Formel / eine Menge von Formeln widersprüchlich/unerfüllbar ist.
- Diese Herleitung geschieht mittels eines Algorithmus
  $\hookrightarrow$ Maschinengestütztes Beweisen.
- Der Resolutionskalkül arbeitet auf Formeln in KNF (Konjunktiver Normalform).
  $\hookrightarrow$ Der erste Schritt ist die gegebenen Formeln in KNF zu bringen.

Äquivalenzen der Aussagenlogik

Mit dem Ersetzungssatz kann man äquivalente Formeln verwenden um eine Formel zu vereinfachen. Einige hilfreiche Äquivalenzen:

- Idempotenz $(F \wedge F) \equiv F$ und $(F \vee F) \equiv F$
- Kommutativität $(F \wedge G) \equiv (G \wedge F)$ und $(F \vee G) \equiv (G \vee F)$
- Assoziativität $((F \wedge G) \wedge H) \equiv (F \wedge (G \wedge H))$ und
  $((F \vee G) \vee H) \equiv (F \vee (G \vee H))$
- Absorption $(F \wedge (F \vee G)) \equiv F$ und $(F \vee (F \wedge G)) \equiv F$
- Distributivität $(F \wedge (G \vee H)) \equiv ((F \wedge G) \vee (F \wedge H))$
  $(F \vee (G \wedge H)) \equiv ((F \vee G) \wedge (F \vee H))$
- Doppelnegation $\neg\neg F \equiv F$
- de Morgansche Regeln $\neg(F \wedge G) \equiv (\neg F \vee \neg G)$
  $\neg(F \vee G) \equiv (\neg F \wedge \neg G)$

Äquivalenzen der Aussagenlogik

Dank dieser Äquivalenzen können wir Formeln kompakter notieren:

- Klammerung zwischen aufeinander folgender $\wedge$ oder $\vee$ ist nicht notwendig:
  $\hookrightarrow$ Statt $((F \wedge G) \wedge H)$ können wir $(F \wedge G \wedge H)$ schreiben.
- Weiters werden wir Klammern um die ganze Formel nicht notieren:
  $\hookrightarrow$ Statt $(F \wedge G)$ können wir $F \wedge G$ schreiben.

Diese Äquivalenzen sind die Basis für **Normalformen**.

Normalform

Eine **Normalform** ist eine Einschränkung auf der Syntax sodass jede beliebige Formel in eine semantisch äquivalente Formel in Normalform umgewandelt werden kann.

- Vereinfacht die maschinelle Verarbeitung von logischen Formeln. (Viele Algorithmen verarbeiten nur eine bestimmte Normalform)

Konjunktive Normalform

Literal: Unter Literalen verstehen wir

- Atome $x \in \mathit{Var}$ und
- negierte Atome $\neg x$ für $x \in \mathit{Var}$.

Definition

Eine Formel ist in **konjunktiver Normalform (KNF)**, wenn sie eine Konjunktion von Disjunktionen von Literalen ist. F ist also von folgender Form ($L_{i,j}$ Literale):

$$F = (L_{1,1} \vee \cdots \vee L_{1,m_1}) \wedge \cdots \wedge (L_{n,1} \vee \cdots \vee L_{n,m_n})$$

Beispiele:

- $(\neg a \vee b) \wedge (a \vee \neg c \vee d) \wedge \neg b$
- $(\neg a \vee x \vee \neg c \vee d) \wedge (\neg b \vee \neg y)$

Disjunktive Normalform

Definition

Eine Formel ist in **disjunktiver Normalform (DNF)**, wenn sie eine Disjunktion von Konjunktionen von Literalen ist. F ist also von folgender Form ($L_{i,j}$ Literale):

$$F = (L_{1,1} \wedge \cdots \wedge L_{1,m_1}) \vee \cdots \vee (L_{n,1} \wedge \cdots \wedge L_{n,m_n})$$

Beispiele:

- $(\neg a \vee b) \wedge (a \vee \neg c \vee d) \wedge \neg b$ (KNF)
- $(\neg a \wedge b \wedge c) \vee (a \wedge \neg c \wedge d)$ (DNF)
- $\neg a \wedge b \wedge c$ (DNF, KNF)
- $\neg a \vee b \vee c$ (DNF, KNF)
- $(\neg a \vee b \vee c) \wedge (a \vee (b \wedge d))$ ()

Jede Formel kann mit Hilfe der präsentierten Äquivalenzumformungen in eine semantisch äquivalente KNF (DNF) transformiert werden.

Konjunktive und Disjunktive Normalform - Beispiele

Zwei Beispiele wie man eine Formel in KNF umwandeln kann:

$$\begin{aligned}
&(A \land B) \to C && \\
&\neg(A \land B) \lor C && \text{(Definition von } \to\text{)} \\
&(\neg A \lor \neg B) \lor C && \text{(de Morgan)} \\
&\neg A \lor \neg B \lor C && \text{(Assoziativität)}
\end{aligned}$$

$$\begin{aligned}
&C \leftrightarrow (A \land B) && \\
&(\neg C \lor (A \land B)) \land (C \lor \neg(A \land B)) && \text{(Definition von } \leftrightarrow\text{)} \\
&(\neg C \lor (A \land B)) \land (C \lor (\neg A \lor \neg B)) && \text{(de Morgan)} \\
&(\neg C \lor (A \land B)) \land (C \lor \neg A \lor \neg B) && \text{(Assoziativität)} \\
&((\neg C \lor A) \land (\neg C \lor B)) \land (C \lor \neg A \lor \neg B) && \text{(Distributivität)} \\
&(\neg C \lor A) \land (\neg C \lor B) \land (C \lor \neg A \lor \neg B) && \text{(Assoziativität)}
\end{aligned}$$

Konjunktive und Disjunktive Normalform - Algorithmus

Diese Schritte lassen sich auch in einen Algorithmus fassen der eine Formel F in eine semantisch äquivalente KNF umzuwandeln.

Algorithmus

1. Ersetze $G \leftrightarrow H$ durch $(G \rightarrow H) \wedge (H \rightarrow G)$
2. Ersetze $G \rightarrow H$ durch $(\neg G \vee H)$
3. Iteriere das Folgende solange wie möglich
   1. Ersetze $\neg\neg G$ durch G
   2. Ersetze $\neg(G \wedge H)$ durch $(\neg G \vee \neg H)$
   3. Ersetze $\neg(G \vee H)$ durch $(\neg G \wedge \neg H)$
4. Iteriere das Folgende solange wie möglich
   1. Ersetze $(G \vee (H \wedge I))$ und $((H \wedge I) \vee G)$ durch $((G \vee H) \wedge (G \vee I))$

„Ersetze“ liest sich als „Ersetze alle Teilformeln der Form“

Klauseln von Formeln

Gegeben sei eine Formel F in KNF, also

$$F = (L_{1,1} \vee \cdots \vee L_{1,m_1}) \wedge \cdots \wedge (L_{n,1} \vee \cdots \vee L_{n,m_n})$$

wobei die $L_{i,j}$ Literale sind.

Konjunktion und Disjunktion sind idempotent, kommutativ und assoziativ. Daher können wir eine Formel in KNF auch als Menge von Mengen anschreiben.

Die F zugeordnete **Klauselmenge** $K(F)$ ist

$$K(F) = \{\{L_{1,1}, \ldots, L_{1,m_1}\}, \ldots, \{L_{n,1}, \ldots, L_{n,m_N}\}\}$$

Eine einzelne Menge $\{L_{i,1}, \ldots, L_{i,m_i}\}$ heißt **Klausel** von F.

- Eine Klausel entspricht einer Disjunktion von Literalen

Klauseln von Formeln

- Verschiedene Formeln können dieselbe Klauselmenge besitzen

Beispiel

Die Formeln

$F_1 = (A \vee B) \wedge C$,
$F_2 = (C \vee C) \wedge (A \vee B) \wedge C$ und
$F_3 = C \wedge (B \vee A)$

besitzen alle die Klauselmenge $\{\{A, B\}, \{C\}\}$.

- Eine leere Klausel entspricht dem Wahrheitswert 0 (falsch).
  (Eine leere Disjunktion ist falsch)
- Eine Klauselmenge die eine leere Klausel enthält hat den Wahrheitswert 0.
  (Eine Konjunktion ist falsch wenn ein Eintrag falsch ist)
- Eine leere Klauselmenge entspricht dem Wahrheitswert 1.
  (Eine leere Konjunktion ist wahr)

Resolvente von Klauseln

Idee: Wenn wir Klauseln $\{A, B\}$ und $\{\neg A, C\}$ haben, dann kann nur entweder A oder $\neg A$ wahr sein. Es folgt, dass B oder C wahr sein muss, d.h. die Klausel {B,C}.

Definition

Eine Klausel K_3 heißt **Resolvente** der Klauseln K_1, K_2 wenn es ein Literal L gibt sodass

- $L \in K_1$, $\neg L \in K_2$
- $K_3 = (K_1 \setminus \{L\}) \cup (K_2 \setminus \{\neg L\})$

Wenn K_3 Resolvente von K_1 und K_2 ist, verwenden wir auch folgende graphische Darstellung.

$$\begin{tikzcd} K_1 \arrow[dr, dash] & & K_2 \arrow[dl, dash] \\ & K_3 & \end{tikzcd}$$

Resolvente von Klauseln - Beispiel

Gegeben sei die Klauselmenge $\{\{A, B, \neg C\}, \{A, \neg B, C\}\}$. Zu den Klauseln dieser Klauselmenge gibt es die folgenden Resolventen:

$$
\begin{tikzcd}[column sep=small]
\{A, B, \neg C\} \arrow[dr, dash] & & \{A, \neg B, C\} \arrow[dl, dash] \\
& \{A, C, \neg C\} &
\end{tikzcd}
\qquad
\begin{tikzcd}[column sep=small]
\{A, B, \neg C\} \arrow[dr, dash] & & \{A, \neg B, C\} \arrow[dl, dash] \\
& \{A, B, \neg B\} &
\end{tikzcd}
$$

Folgendes ist aber FALSCH:

$$
\begin{tikzcd}[column sep=small]
\{A, B, \neg C\} \arrow[dr, dash] & & \{A, \neg B, C\} \arrow[dl, dash] \\
& \{A\} &
\end{tikzcd}
$$

Resolutionslemma

Satz (Resolutionslemma)

Für jede Formel F mit Klauselmenge $K(F)$: Ist K_3 eine Resolvente zweier Klauseln $K_1, K_2 \in K(F)$ so ist F semantisch äquivalent zu allen Formeln mit Klauselmenge $K(F) \cup \{K_3\}$.

- Wir können also neue Klauseln hinzufügen ohne die Semantik zu verändern.
- Weiters können wir Klauseln die Tautologien entsprechen (Klauseln die x und $\neg x$ enthalten) weglassen ohne die Semantik zu verändern.

Idee des Resolutionskalküls: Solange Resolventen bilden bis man entweder die leere Klausel enthält oder alle möglichen Resolventen gebildet wurden.

Sei $K = K(F)$ die Klauselmenge einer Formel:

- $Res(K) = K \cup \{R \mid R \text{ ist Resolvente zweier Klauseln in } K\}$
- $Res^0(K) = K$ und für $n \geq 1$ sei

$$Res^n(K) = Res(Res^{n-1}(K))$$

- $Res^\infty(K) = \bigcup_{n \geq 0} Res^n(K)$

Satz

Eine Formel F ist genau dann unerfüllbar wenn $\emptyset \in Res^\infty(K(F))$.

Hat F nur n Variablen dann hat $Res^\infty(K)$ höchsten 4^n Klauseln.

$\hookrightarrow$ $Res^\infty(K)$ kann mit (maximal) 4^n Iterationen berechnet werden.

Resolutionskalkül - Erfüllbarkeitstest

Gegeben sei eine Formel in KNF:

1. Bilde Klauselmenge $K(F)$
2. Berechne $Res^1(K), Res^2(K), \ldots, Res^n(K)$ bis

   $$\emptyset \in Res^n(K) \text{ oder } Res^{n-1}(K) = Res^n(K).$$

3. Wenn $\emptyset \in Res^n(K)$ „F unerfüllbar“

   anderenfalls „F erfüllbar“.

Formalisieren in Aussagenlogik - Beispiel 2 VO1

Aussage 1: Wenn der Geiger das Konzert gibt, werden viele kommen, wenn die Preise nicht zu hoch sind.
Aussage 2: Wenn der Geiger das Konzert gibt, werden die Preise nicht zu hoch sein.
Schluss: Daher werden, falls der Geiger das Konzert bestreitet, viele kommen.

Wir wollen wissen ob der Schluss korrekt ist.

Atomare Aussagen:
P: „der Geiger gibt das Konzert"
C: „viele werden kommen"
H: „die Preise sind zu hoch"

Aussage 1: $P \rightarrow (\neg H \rightarrow C)$
Aussage 2: $P \rightarrow \neg H$
Schluss: $P \rightarrow C$

Wir wollen wissen ob $P \rightarrow C$ aus $P \rightarrow (\neg H \rightarrow C)$ und $P \rightarrow \neg H$ folgt.

Resolutionskalkül - Beispiel

Wir betrachten die Formel aus dem Geiger-Beispiel:

$$(P \to (\neg H \to C)) \land (P \to \neg H) \land \neg(P \to C)$$

1.Schritt: Wir bringen die Formel in KNF.

$$\begin{aligned} (\neg P \lor (H \lor C)) \land (\neg P \lor \neg H) \land \neg(\neg P \lor C) & \qquad \text{Definition von } \to \\ (\neg P \lor H \lor C) \land (\neg P \lor \neg H) \land P \land \neg C & \qquad \text{de Morgan, Assoziativität} \end{aligned}$$

2.Schritt: Bilden der Klauselmengen

$$K(F) = \{\{\neg P, H, C\}, \{\neg P, \neg H\}, \{P\}, \{\neg C\}\}$$

Resolutionskalkül - Beispiel

3.Schritt: Resolventen berechnen

$$Res^0(K) = \{\{\neg P, H, C\}, \{\neg P, \neg H\}, \{P\}, \{\neg C\}\}$$
$$Res^1(K) = Res^0(K) \cup \{\{\neg P, C\}, \{H, C\}, \{\neg P, H\}, \{\neg H\}\}$$
$$Res^2(K) = Res^1(K) \cup \{\{C\}, \{\neg P\}, \{H\}\}$$
$$Res^3(K) = Res^2(K) \cup \{\{\}\}$$

Da wir die leere Klausel erreicht haben ist die Formel unerfüllbar.

Resolutionskalkül - Beispiel

Oft müssen wir nicht alle Resolventen berechnen um einen Widerspruch (die leere Klausel) zu finden:

$$
\begin{tikzcd}[column sep=small]
\{\neg P, H, C\} \arrow[dr, dash] & & \{\neg P, \neg H\} \arrow[dl, dash] & & \{P\} \arrow[ddl, dash] & & \{\neg C\} \arrow[dddl, dash] \\
& \{\neg P, C\} \arrow[drr, dash] & & & & & \\
& & & \{C\} \arrow[drr, dash] & & & \\
& & & & & \emptyset &
\end{tikzcd}
$$

Aber: Um zu zeigen dass eine Formel widerspruchsfrei ist muss man alle Resolventen berechnen!!!

Subsection 12

Einheitsresolution

Einheitsresolution

Eine **Einheitsklausel** ist eine Klausel die nur aus einem (positiven oder negativen) Literal besteht.

Beobachtung: Enthält die Klauselmenge

- eine positive Einheitsklausel $\{x\}$ so muss x jedenfalls auf 1 gesetzt werden
- eine negative Einheitsklausel $\{\neg x\}$ so muss x jedenfalls auf 0 gesetzt werden

um die Formel zu erfüllen.

Einheitsresolution (unit propagation)

Formel F in KNF, $K = K(F)$

- Solange die Klauselmenge K eine Einheitsklausel enthält wähle eine Einheitsklausel $\{L\}$
  - Entferne alle Klauseln in K die L enthalten
  - Lösche $\neg L$ aus allen Klauseln in K
- Wenn $\emptyset \in K$ dann ist die Formel F unerfüllbar.

Einheitsresolution

Sei K eine Klauselmenge dann schreiben wir $ERes(K)$ für die vereinfachte Klauselmenge die der Einheitsresolution-Algorithmus erzeugt.

Satz (Erfüllbarkeitsäquivalenz)

Eine Klauselmenge K ist genau dann erfüllbar wenn ihre Vereinfachung $ERes(K)$ durch Einheitsresolution erfüllbar ist.

- Wenn $\emptyset \in ERes(K)$ dann ist die Formel F unerfüllbar.

Einheitsresolution und allgemeine Klauselmengen

Gegeben: Klauselmenge $K(F)$.

Ziel: Testen ob die Klauselmenge $K(F)$ erfüllbar ist.

Idee: Kombiniere Resolution mit Einheitsresolution

Verfahren: Wandle den Erfüllbarkeitstest mittels Resolution wie folgt ab

- Zu Beginn starte Einheitsresolution um die Klauselmenge zu verkleinern.
- Wurde die leere Klausel berechnet, wird ist $K(F)$ unerfüllbar; sonst starte Resolution auf der verkleinerten Klauselmenge.
- Wann immer Resolution eine Einheitsklausel hinzufügt, starte Einheitsresolution um die Klauselmenge zu verkleinern bevor mit Resolution fortgesetzt wird.

Einheitsresolution und allgemeine Klauselmengen - Beispiel

Beispiel

Formel $F = a \wedge (\neg a \vee b \vee c) \wedge (b \leftrightarrow c) \wedge (\neg a \vee \neg b \vee \neg c) \wedge (a \vee b \vee c)$
Klauselmenge
$K(F) = \{\{a\}, \{\neg a, b, c\}, \{b, \neg c\}, \{\neg b, c\}, \{\neg a, \neg b, \neg c\}, \{a, b, c\}\}$
mit Einheitsklausel $\{a\}$

Einheits-Resolution mit $\{a\}$ ergibt: $\{\{b, c\}, \{b, \neg c\}, \{\neg b, c\}, \{\neg b, \neg c\}\}$

Ein Resolution-Schritt ergibt: $\{\{b, c\}, \{b, \neg c\}, \{\neg b, c\}, \{\neg b, \neg c\}, \{b\}\}$

Einheits-Resolution mit $\{b\}$ ergibt: $\{\{c\}, \{\neg c\}\}$

Einheits-Resolution mit $\{\neg c\}$ ergibt: $\{\{\}\}$

Die Formel F ist also unerfüllbar.

Hornlogik

Die Hornlogik ist eine Einschränkung der Aussagenlogik, d.h. es sind nur Formeln einer speziellen Form erlaubt.

- Benannt nach dem Logiker Alfred Horn.
- Kann mit speziellen Resolutions-Varianten sehr effizient ausgewertet werden.
  $\hookrightarrow$ Einheitsresolution, SLD-Resolution
- Dient als Grundlage der logischen Programmierung (Prolog).
- Bedeutung in der Komplexitätstheorie.

Hornlogik

Definition
Eine Klausel heißt **Hornklausel** wenn sie höchstens ein positives Literal enthält. Eine Formel F in KNF heißt **Hornformel** wenn $K(F)$ nur aus Hornklauseln besteht.

Beispiel für eine Hornformel:

- $(a \vee \neg b \vee \neg c) \wedge (\neg b \vee \neg c) \wedge c$

Unterschiedliche Arten von Hornklauseln:

- Tatsachenklauseln/Fakten: nur ein positives Literal, z.B. c
- Regeln: ein positives Literal, z.B. $(a \vee \neg b \vee \neg c)$ $\big(\equiv (b \wedge c) \rightarrow a\big)$
- Zielklausel: nur negative Literale, z.B. $(\neg b \vee \neg c)$

Einheitsresolution

Sei K eine Klauselmenge dann schreiben wir $ERes(K)$ für die vereinfachte Klauselmenge die der Einheitsresolution-Algorithmus erzeugt.

Satz (Erfüllbarkeitsäquivalenz)

Eine Klauselmenge K ist genau dann erfüllbar wenn ihre Vereinfachung $ERes(K)$ durch Einheitsresolution erfüllbar ist.

- Wenn $\emptyset \in ERes(K)$ dann ist die Formel F unerfüllbar.
- Wenn K eine **Horn-Formeln** ist gilt auch:
    - Wenn $\emptyset \notin ERes(K)$ dann ist die Formel erfüllbar.

Satz

Eine Horn-Formel ist genau dann erfüllbar wenn in $\emptyset \notin ERes(K)$.

Hornlogik/Einheitsresolution - Beispiel 1

Beispiel

Formel $F = a \wedge b \wedge ((a \wedge b) \rightarrow c) \wedge (d \rightarrow e)$
Klauselmenge $K(F) = \{\{a\}, \{b\}, \{\neg a, \neg b, c\}, \{\neg d, e\}\}$
mit zwei Einheitsklausel $\{a\}, \{b\}$

$$\begin{array}{rl} \{\{a\}, \{b\}, \{\neg a, \neg b, c\}, \{\neg d, e\}\} & \text{Resolviere } \{a\} \\ \{\{b\}, \{\neg b, c\}, \{\neg d, e\}\} & \text{Resolviere } \{b\} \\ \{\{c\}, \{\neg d, e\}\} & \text{Resolviere } \{c\} \\ \{\{\neg d, e\}\} & \end{array}$$

Die Formel F ist erfüllbar.

Hornlogik/Einheitsresolution - Beispiel 2

Beispiel

Formel $F = a \wedge b \wedge ((a \wedge b) \rightarrow c) \wedge (\neg a \vee \neg b) \wedge (d \rightarrow e)$
Klauselmenge $K(F) = \{\{a\}, \{b\}, \{\neg a, \neg b, c\}, \{\neg a, \neg b\}, \{\neg d, e\}\}$
Zwei Einheitsklausel $\{a\}, \{b\}$

$$\begin{aligned} \{\{a\}, \{b\}, \{\neg a, \neg b, c\}, \{\neg a, \neg b\}, \{\neg d, e\}\} &\quad \text{Resolviere } \{a\} \\ \{\{b\}, \{\neg b, c\}, \{\neg b\}, \{\neg d, e\}\} &\quad \text{Resolviere } \{b\} \\ \{\{c\}, \{\}, \{\neg d, e\}\} & \end{aligned}$$

Da wir die leere Klausel erhalten haben ist die Formel F unerfüllbar.

Hornlogik/Einheitsresolution - Beispiel 3

(zum Selbststudium)

Betrachte die Formel

$$F = (\neg T \vee \neg W) \wedge (R \rightarrow T) \wedge W \wedge R$$

Durch Auflösen der Inklusion $(R \rightarrow T)$ zu $(\neg R \vee T)$ erhalten wir

$$K(F) = \{\{\neg T, \neg W\}, \{\neg R, T\}, \{W\}, \{R\}\}$$

$$\begin{array}{rl}
\{\{\neg T, \neg W\}, \{\neg R, T\}, \{W\}, \{R\}\} & \text{Resolviere } \{W\} \\
\{\{\neg T\}, \{\neg R, T\}, \{R\}\} & \text{Resolviere } \{\neg T\} \\
\{\{\neg R\}, \{R\}\} & \text{Resolviere } \{\neg R\} \\
\{\{\}\} &
\end{array}$$

Da wir die leere Klausel erhalten haben ist die Formel F unerfüllbar.

Subsection 14

Limitierungen von Aussagenlogik

Limitierungen von Aussagenlogik

Es gibt logische Zusammenhänge die sich mit Aussagenlogik nicht abbilden lassen.

Beispiel

Aussage 1: Fritz fährt Ski
Aussage 2: Uli fährt Ski

Obwohl beide Aussagen augenscheinlich von gleicher Natur sind, kann Aussagenlogik die gemeinsame Struktur nicht modellieren.

Beispiel

Aussage 1: Jeder Mensch ist wertvoll.
Aussage 2: Hans ist ein Mensch.

Wir würden also gerne schließen „Hans ist wertvoll“. Das ist aber in Aussagenlogik nicht möglich.

Limitierungen von Aussagenlogik

Limitierungen von Aussagenlogik

- Aussagen werden als Atome genutzt und nicht weiter analysiert
  $\hookrightarrow$ oft stecken mehrere Informationen in einer atomaren Aussage
- Die innere Struktur einer Aussage geht verloren.
- Es ist unmöglich/schwer auszudrücken, dass
  - gewisse Beziehungen zwischen Objekten gelten;
  - etwas für alle Objekte gilt; oder
  - es ein Objekt mit einer Eigenschaft geben muss.

Wir benötigen also eine **ausdrucksstärkere Logik**.

Prädikatenlogik

Section 4

Prädikatenlogik

Prädikatenlogik – Prädikate

Wir wollen Objekte von ihren Eigenschaften und Beziehungen zwischen einander trennen. Dazu verwenden wir **Prädikate**.

Beispiel

Um die Aussage „Hans ist ein Mensch“ zu formalisieren verwenden wir ein (einstelliges) Prädikat *Mensch*(.) und ein Objekt *hans* und bekommen

$$\mathit{Mensch}(\mathit{hans})$$

Mit dem gleichen Prädikat können wir auch ausdrücken dass Barbara ein Mensch ist

$$\mathit{Mensch}(\mathit{barbara})$$

Wir können auch ausdrücken das der Kater Findus kein Mensch ist

$$\neg\mathit{Mensch}(\mathit{findus})$$

Ein Prädikat kann für manche Objekte wahr aber für andere falsch sein.

Prädikatenlogik – Quantoren

Prädikate erlauben es mittels Quantoren Aussagen über die Gesamtheit der Objekte die eine Eigenschaft erfüllen zu Treffen.

Wir betrachten zwei **Quantoren**

- **Allquantor** $\forall$: etwas gilt für alle Objekte
- **Existenzquantor** $\exists$: etwas gilt für mindestens ein Objekt

Beispiel

Die Aussage „Jeder Mensch ist wertvoll" können wir jetzt wie folgt formalisieren:

$$\forall x\ (\mathit{Mensch}(x) \rightarrow \mathit{Wertvoll}(x))$$

Gemeinsam mit $\mathit{Mensch}(\mathit{hans})$ schließen wir

$$\mathit{Wertvoll}(\mathit{hans})$$

Prädikatenlogik – 2-stellige Prädikate

Mit Mehrstelligen Prädikaten können wir Beziehungen zwischen Objekten modellieren.

Beispiel

Aussage 1: Fritz fährt Ski
Aussage 2: Uli fährt Ski
Aussage 3: Uli fährt Snowboard

1.Versuch: $\mathit{FahrtSki}(\mathit{fritz}) \wedge \mathit{FahrtSki}(\mathit{uli}) \wedge \mathit{FahrtSnowboard}(\mathit{uli})$

Wir verlieren aber etwas von der gemeinsamen Struktur der Aussagen.

Besser: 2-stelliges Prädikat *Fahrt*

$$\mathit{Fahrt}(\mathit{fritz}, \mathit{ski}) \wedge \mathit{Fahrt}(\mathit{uli}, \mathit{ski}) \wedge \mathit{Fahrt}(\mathit{uli}, \mathit{snowboard})$$

Jetzt können wir auch sagen das jeder Wintersportler mindestens ein Sportgerät fährt (das muss aber nicht Ski oder Snowboard sein)

$$\forall x \ (\mathit{Wintersportler}(x) \rightarrow \exists y \ \mathit{Fahrt}(x, y))$$

Prädikatenlogik – Funktionen

Manche Beziehungen sind von der Form, dass einem Objekt ein anderes eindeutig zugeordnet wird, z.B. jeder hat nur eine (biologische) Mutter. In solchen Fällen verwendet man statt einem Prädikat besser eine Funktion.

Beispiel

Aussage: „Jo Anns Mutter liebt Musik"

Ohne Funktionen: $\exists x(Mutter(x, joAnn) \wedge Liebt(x, musik))$.

Das liest sich als: „Jo Ann hat mindestens eine Mutter die Musik liebt"

Mit Funktionen
Wir verwenden: Objekte *joAnn*, *musik*,
das 2-stellige Prädikat *Liebt*,
die 1-stellige Funktion *mutter*

$$Liebt(mutter(joAnn), musik)$$

Prädikatenlogik – Funktionen

Unterschied **Prädikat – Funktion** (in der Prädikatenlogik):

- Prädikate ordnen Objekten Wahrheitswerte zu.
- Funktionen ordnen Objekten wieder Objekte zu.

Prädikatenlogik – Funktionen

Beispiel

Aussage: Wenn a größer als b ist dann ist für alle x auch $a + x$ größer als $b + x$

$$Großer(a, b) \rightarrow \forall x\, Großer(plus(a, x), plus(b, x))$$

Beispiel

Die Multiplikation kann mittels der Addition wie folgt definiert werden:

- x mal 0 ist 0
- x mal $(y + 1)$ ist x mal y plus x

In der Prädikatenlogik erhalten wir:

$$\forall x \forall y \big(IstGleich(mal(x, 0), 0) \wedge \\ IstGleich(mal(x, plus(y, 1)), plus(mal(x, y), x)) \big)$$

Das entspricht $\forall x \forall y \big(x \cdot 0 = 0 \wedge x \cdot (y + 1) = (x \cdot y) + x \big)$.

Formale Definition der Prädikatenlogik

Nun kennen wir die zentralen Bauteile der Prädikatenlogik und können diese formal definieren.

2 Schritte:

- Zuerst definieren wir die **formale Syntax**, d.h., wie Prädikatenlogische Formeln aufgebaut sind.
- Dann geben wir diesen Formeln eine **formale Semantik**, d.h., wir geben den Formeln eine Bedeutung.

Subsection 3

Formale Syntax

Formale Syntax der Prädikatenlogik

Nomenklatur

Die elementaren Bestandteile prädikatenlogischer Formeln:

- Eine **Variable** ist ein Symbol der Form $x, y, z, \ldots$
- Ein **Funktionssymbol** fängt mit einem Kleinbuchstaben an und hat eine gewisse Stelligkeit, z.B. mutter(.) ist eine 1-stellige Funktion
- Nullstellige Funktionssymbole heißen auch **Konstanten**, z.B. joAnn
- Ein **Prädikatensymbol** fängt mit einem Großbuchstaben an und hat immer die gleiche Stelligkeit, z.B. Liebt(.,.) ist ein 2-stelliges Prädikat.

Ausnahmen von der obigen Nomenklatur-Regel sind die üblichen arithmetischen Operationen $+, *, \ldots$ und Relationen $=, \leq, \geq, \ldots$

Formale Syntax der Prädikatenlogik – Terme

Definition

Ein **Term** wird induktiv nach den folgenden Regeln gebildet

- Jede Variable ist ein Term
- Ist f ein k-stelliges Funktionssymbol, und sind $t_1, \ldots, t_k$ Terme dann ist $f(t_1, \ldots, t_k)$ ein Term. Insbesondere sind alle Konstanten Terme.

Terme werden also aus Variablen und Funktionssymbolen (insbesondere Konstanten) gebildet und repräsentieren Objekte.

Beispiel

- x, y (Variablen)
- *joAnn*, *musik* (Konstanten)
- $mutter(x)$, $mutter(mutter(joAnn))$ (Terme mit Funktionssymbolen)

Syntax der Prädikatenlogik – Atomare Formeln

Atomare Formeln repräsentieren simple Aussagen.

Definition

Wenn P ein k-stelliges Prädikatensymbol ist und $t_1, \ldots, t_k$ Terme sind dann ist $P(t_1, \ldots, t_k)$ eine **atomare Formel**.

Atomare Formeln werden gebildet indem man Terme in Prädikate (entsprechend der Stelligkeit) „einsetzt“.

Beispiel

- $Liebt(x, y)$
- $Liebt(mutter(joAnn), musik)$
- $Prim(zwei)$
- $Staffel(bernhard, elfi, julia, y)$

Syntax der Prädikatenlogik – Formeln

Nun können wir prädikatenlogische Formeln definieren:

Definition

Eine **Formel** wird induktiv nach den folgenden Regeln gebildet

- Jede atomare Formel ist eine Formel
- Sind F und G Formeln dann sind auch

  $$\neg F, \quad (F \wedge G), \quad \text{und} \quad (F \vee G)$$

  Formeln.
- Ist x eine Variable und F eine Formel dann sind auch

  $$\forall x\, F \qquad \text{und} \qquad \exists x\, F$$

  Formeln.

Die Implikation $F \rightarrow G$ verwenden wir als Abkürzung für $\neg F \vee G$.

Syntax der Prädikatenlogik – Beispiel

Beispiel

Prädikate: $P(.), R(.,.), S(.,.,.,.)$
Funktionen: $f(.), g(.,.), h(.,.,.)$
Variablen: x, y, z

- **Terme:** x, $f(x)$, $g(f(x), y)$, $h(y, f(z), x)$, ...
- **Atomare Formeln:** $P(x)$, $R\big(g(f(x), y), x\big)$, $S\big(z, x, f(x), y\big)$ ...
- **Formeln:**
  $(P(x) \wedge \neg R(g(f(x), y), x))$,
  $\exists x\, P(x)$,
  $\exists z\big(\exists x\, P(x) \vee \forall x\, S(z, x, f(x), y)\big)$, ...

Prädikatenlogik – Gebundene / Freie Variablen

Eine Variable, die bei jedem Auftreten einem Quantor zugeordnet ist, bezeichnen wir als **gebunden**. Anderenfalls nennen wir die Variable **frei**.

Definition

- In atomaren Formeln sind alle Variablen frei.
- In Formeln der Form $\forall x\, F$, $\exists x\, F$ kommt x gebunden vor. Alle anderen Variablen kommen gebunden vor, wenn sie in F gebunden vorkommen.
- In Formeln der Form $\neg F$ kommt x gebunden vor, wenn es in F gebunden vorkommt.
- In Formeln der Form $(F \wedge G)$, $(F \vee G)$ kommt x gebunden/frei vor wenn es in F und G gebunden/frei vorkommt.

Eine Formel F ist eine **geschlossene Formel** wenn F keine freien Variablen hat.

Prädikatenlogik – Gebundene / Freie Variablen

Beispiele

- $\forall x \big(Mensch(x) \land Spielt(x,y) \big)$

  x kommt gebunden vor, y kommt frei vor

- $\forall x \big(Mensch(x) \land \exists y \, Spielt(x,y) \big)$

  x, y kommen gebunden vor
  geschlossene Formel

- $\big(\forall x \; Mensch(x) \land \exists y \, Spielt(x,y) \big)$

  x kommt frei und gebunden vor, y kommt gebunden vor

- $\big(\forall x \, Mensch(y) \land \exists y \, Spielt(x,y) \big)$

  x kommt frei for, y kommt frei und gebunden vor

Subsection 4

Formale Semantik

Semantik der Prädikatenlogik - Strukturen

Idee: Die Semantik wird mit Hilfe sogenannter Strukturen definiert.

Eine Struktur besteht aus

- eine Grundmenge von Objekten,
- konkrete Funktionen und Prädikate über diese Objekte
- eine Zuordnung zwischen diesen Funktionen/Prädikaten und den Funktionssymbolen/Prädikatensymbolen in der Formel
- Eine Funktion die jeder (freien) Variablen ein Objekt zuweist

- Atomare Formeln können in einer Struktur durch Einsetzen überprüft werden.
- Zusammengesetzte Formeln werden mit einem rekursiven Schema ausgewertet (ähnlich wie in der Aussagenlogik).
- Die Semantik einer Formel ergibt sich durch die Strukturen die Modell der Formel sind.

Semantik der Prädikatenlogik - Strukturen

Zuerst müssen wir Strukturen definieren:

Definition

Eine zu einer Formel F passende **Struktur** ist ein Tupel $\alpha = (U, \varphi, \psi, \xi)$

- U ist eine nicht leere Menge, das **Universum** oder die Grundmenge.
- φ eine Abbildung die jedem k-stelligen Funktionssymbol f in F eine Funktion $f^\varphi : U^k \to U$ zuordnet.
- ψ eine Abbildung die jedem k-stelligen Prädikatensymbol P in F ein Prädikat $P^\psi \subseteq U^k$ zuordnet.
- ξ eine Abbildung die jeder Variablen x ein Element $x^\xi \in U$ zuordnet.

ξ gibt freien Variablen eine Bedeutung und hat keinen Einfluss auf die Semantik von geschlossenen Formeln.

Semantik der Prädikatenlogik - Strukturen

Beispiel

Betrachten wir die Formel:

$$\exists x \, Liebt(mutter(x), musik)$$

Eine passende Struktur wäre:

- $U = \{JoAnn, Mum, STS\}$
- $musik^{\varphi} = STS$
  $$\begin{aligned} mutter^{\varphi} : \; & mutter^{\varphi}(JoAnn) = Mum, \\ & mutter^{\varphi}(Mum) = JoAnn, \\ & mutter^{\varphi}(STS) = STS \end{aligned}$$
- $Liebt^{\psi} = \{(Mum, STS), (JoAnn, Mum)\}$
- $x^{\xi} = Mum$

Semantik der Prädikatenlogik - Terme

Wir definieren zunächst den Wert eines Terms in einer Struktur α

Definition

Der **Wert** $\alpha(t)$ eines Terms t in der Struktur α ist wie folgt gegeben:

- Ist t eine Variable dann $\alpha(t) = t^{\xi}$
- Ist t von der Form $f(t_1, \ldots, t_k)$ dann $\alpha(t) = f^{\varphi}(\alpha(t_1), \ldots, \alpha(t_k))$

Beispiel

Gegeben α mit $joAnn^{\varphi} = Susi$, $mutter^{\varphi}(Susi) = Mum$

- $\alpha(joAnn) = joAnn^{\varphi} = Susi$
- $\alpha(mutter(joAnn)) = mutter^{\varphi}(\alpha(joAnn)) = mutter^{\varphi}(Susi) = Mum$

Beispiel

Gegeben α mit $mutter^{\varphi}(Mum) = JoAnn$, $x^{\xi} = Mum$

- $\alpha(x) = x^{\xi} = Mum$
- $\alpha(mutter(x)) = mutter^{\varphi}(\alpha(x)) = mutter^{\varphi}(Mum) = JoAnn$

Semantik der Prädikatenlogik - Atome

Definition

Der **Wahrheitswert** $\alpha(F)$ einer atomaren Formel $F = P(t_1, \ldots, t_k)$ in einer Struktur α ist gegeben durch

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

Beispiel

Gegeben Struktur α mit

- $\alpha(joAnn) = joAnn^\varphi = Susi$, $\alpha(musik) = musik^\varphi = STS$
- $mutter^\varphi(Susi) = Mum$, $mutter^\varphi(Mum) = Susi$, $mutter^\varphi(STS) = STS$
- $Liebt^\psi = \{(Mum, STS), (Susi, Mum)\}$
- $x^\xi = Mum$

dann gilt

1. $\alpha(Liebt(mutter(joAnn), musik)) = Liebt^\psi(Mum, STS) = 1$
2. $\alpha(Liebt(mutter(x), musik)) = Liebt^\psi(Susi, STS) = 0$

Zwei **Spezialfälle:**

- Konstanten, als 0-stelligen Funktionen, wird von α immer ein Objekt $u \in U$ zugewiesen.
- 0-stelligen Prädikatensymbolen wird von α ein Wahrheitswert zugewiesen.

Semantik der Prädikatenlogik – Boolesche Operatoren

Die Semantik der Operatoren $\neg, \wedge,$ und $\vee$ ist analog zur Aussagenlogik:

$$\alpha(F \wedge G) = \begin{cases} 1 & \text{Wenn } \alpha(F) = 1 \text{ und } \alpha(G) = 1 \\ 0 & \text{sonst} \end{cases}$$

$$\alpha(F \vee G) = \begin{cases} 1 & \text{Wenn } \alpha(F) = 1 \text{ oder } \alpha(G) = 1 \\ 0 & \text{sonst} \end{cases}$$

$$\alpha(\neg F) = \begin{cases} 1 & \text{Wenn } \alpha(F) = 0 \\ 0 & \text{sonst} \end{cases}$$

$$\alpha(F \rightarrow G) = \alpha(\neg F \vee G)$$

Semantik der Prädikatenlogik – Boolesche Operatoren

Beispiel

Gegeben Struktur α mit

- $\alpha(joAnn) = Susi$, $\alpha(musik) = STS$
- $\alpha(mutter(Susi)) = Mum$, $\alpha(mutter(Mum)) = Susi$, $\alpha(mutter(STS)) = STS$
- $Liebt^{\psi} = \{(Mum, STS), (Susi, Mum)\}$
- $x^{\xi} = Mum$

wir wissen schon

1. $\alpha(Liebt(mutter(joAnn), musik)) = Liebt^{\psi}(Mum, STS) = 1$
2. $\alpha(Liebt(mutter(x), musik)) = Liebt^{\psi}(Susi, STS) = 0$

Wir betrachten nun

$$\alpha\,(Liebt(mutter(joAnn), musik) \wedge Liebt(mutter(x), musik)) =$$
$$\alpha\,(Liebt(mutter(joAnn), musik)) \wedge \alpha\,(Liebt(mutter(x), musik)) =$$
$$1 \wedge 0 = 0$$

Semantik der Prädikatenlogik – Quantoren

Für die Semantik der **Quantoren** $\forall$,$\exists$ benötigen wir zusätzliche Notation.

Definition

Für gegebene Struktur $\alpha = (U, \varphi, \psi, \xi)$, Variable x und $u \in U$ definieren wir die Struktur $\tilde{\alpha}_x^u = (U, \varphi, \psi, \tilde{\xi})$, sodass $x^{\tilde{\xi}} = u$ und $y^{\tilde{\xi}} = y^{\xi}$ für alle anderen Variablen y.

$\tilde{\alpha}_x^u$ setzt also $\tilde{\alpha}_x^u(x) = u$ und lässt α sonst unverändert.

Beispiel

Gegeben: $\alpha = (\{a, b, c\}, \varphi, \psi, \xi)$ mit $x^{\xi} = a, y^{\xi} = b$

- $\alpha(x) = a$, $\alpha(y) = b$
- $\tilde{\alpha}_x^c(x) = c$, $\tilde{\alpha}_x^c(y) = b$
- $\tilde{\alpha}_y^c(x) = a$, $\tilde{\alpha}_y^c(y) = c$
- $\tilde{\alpha}_{x,y}^{c,a}(x) = c$, $\tilde{\alpha}_{x,y}^{c,a}(y) = a$

Semantik der Prädikatenlogik – Quantoren

Definition

Die **Wahrheitswerte** $\alpha(\forall x\, F)$, $\alpha(\exists x\, F)$ sind wie folgt definiert:

$$\alpha\,(\forall x\, F) = \begin{cases} 1 & \text{falls für alle } u \in U \text{ gilt } \tilde{\alpha}_x^u(F) = 1 \\ 0 & \text{sonst} \end{cases}$$

$$\alpha\,(\exists x\, F) = \begin{cases} 1 & \text{falls es ein } u \in U \text{ gibt sodass } \tilde{\alpha}_x^u(F) = 1 \\ 0 & \text{sonst} \end{cases}$$

Semantik der Prädikatenlogik – Quantoren

Beispiel

Betrachte wieder die Struktur $\alpha = (U, \varphi, \psi, \xi)$:

- $U = \{JoAnn, Mum, STS\}$
- $musik^{\varphi} = STS$
  $mutter^{\varphi}: mutter^{\varphi}(JoAnn) = Mum,$
  $mutter^{\varphi}(Mum) = JoAnn,$
  $mutter^{\varphi}(STS) = STS$
- $Liebt^{\psi} = \{(Mum, STS), (JoAnn, Mum)\}$
- $x^{\xi} = Mum$

Uns interessiert:

$$\alpha\,(\exists x\, Liebt(mutter(x), musik))$$

und

$$\alpha\,(\forall x\, Liebt(mutter(x), musik))$$

Semantik der Prädikatenlogik – Quantoren

Beispiel (cont.)

In beiden Fällen betrachte:

- $\tilde{\alpha}_x^{JoAnn}(Liebt(mutter(x), musik))=$
  $Liebt^{\psi}(\tilde{\alpha}_x^{JoAnn}(mutter(x)), \tilde{\alpha}_x^{JoAnn}(musik))=$
  $Liebt^{\psi}(mutter^{\varphi}(\tilde{\alpha}_x^{JoAnn}(x)), musik^{\varphi})=$
  $Liebt^{\psi}(mutter^{\varphi}(JoAnn), STS)=Liebt^{\psi}(Mum, STS) = 1$
- $\tilde{\alpha}_x^{Mum}(Liebt(mutter(x), musik))= \ldots =$
  $Liebt^{\psi}(mutter^{\varphi}(Mum), STS)=Liebt^{\psi}(JoAnn, STS) = 0$
- $\tilde{\alpha}_x^{STS}(Liebt(mutter(x), musik))= \ldots =$
  $Liebt^{\psi}(mutter^{\varphi}(STS), STS)=Liebt^{\psi}(STS, STS) = 0$

Wir erhalten

$$\alpha(\exists x\, Liebt(mutter(x), musik)) = 1$$

$$\alpha(\forall x\, Liebt(mutter(x), musik)) = 0$$

Subsection 5

Fundamentale Begriffe & Sätze

Prädikatenlogik – Begriffe

Elementare Begriffe:

- Eine Struktur α heißt **Modell** für F, wenn $\alpha(F) = 1$. Wir schreiben $\alpha \models F$.
- Eine Formel F heißt **allgemein gültig** oder **Tautologie** wenn alle zu F passenden Strukturen α auch Modelle von F sind.
- Eine Formel F heißt **erfüllbar** wenn es ein Modell für F gibt, anderenfalls **unerfüllbar**.

Anmerkungen

- F ist unerfüllbar genau dann wenn $\neg F$ eine Tautologie ist.
- Die Aussagenlogik ist ein Spezialfall der Prädikatenlogik: Aussagenlogik entspricht der Prädikatenlogik mit ausschließlich 0-stelligen Prädikaten und ohne Quantoren.

Prädikatenlogik – Begriffe

- Eine Struktur α heißt **Modell** für eine Menge von Formeln $\mathcal{F}$, wenn α eine Modell jeder Formel $G \in \mathcal{F}$ ist. Wir schreiben $\alpha \models \mathcal{F}$.
- Zwei (Mengen von) Formeln F, G sind **semantisch äquivalent** wenn Sie die gleichen Modelle haben. Wir schreiben $F \equiv G$.
- Eine Formel G **folgt** aus einer Formel / einer Menge von Formeln $F/\mathcal{F}$ wenn jedes Modell von $F/\mathcal{F}$ auch Modell von G ist. Wir schreiben $F \models G/\mathcal{F} \models G$.

Satz

Zwei Formeln F,G sind semantisch äquivalent ($F \equiv G$) genau dann wenn $F \models G$ und $G \models F$.

Satz (Ersetzungssatz)

Seien F_1 und F_2 zwei semantisch äquivalente Formeln und sei G eine Formel die F_1 als Teilformel enthält. Dann gilt $G \equiv G[F_1/F_2]$.

Erinnerung: $G[F_1/F_2]$ erhält man aus G indem man F_1 durch F_2 ersetzt.

Prädikatenlogik – Fundamentale Äquivalenzen

Es gelten die semantischen Äquivalenzen aus der Aussagenlogik
und weiters gelten folgende Äquivalenzen für Quantoren:

Vertauschen von Negation und Quantoren

- $\neg\forall x\, F \equiv \exists x\, \neg F$
- $\neg\exists x\, F \equiv \forall x\, \neg F$

Reihenfolge von Quantoren

- $\forall x\, \forall y\, F \equiv \forall y\, \forall x\, F$
- $\exists x\, \exists y\, F \equiv \exists y\, \exists x\, F$

Vorsicht: $\forall x\, \exists y\, F \not\equiv \exists y\, \forall x\, F$

- "Zu jedem Schloss gibt es einen passenden Schlüssel." vs.
- "Es gibt einen Schlüssel der zu jedem Schloss passt."

Prädikatenlogik – Fundamentale Äquivalenzen

Quantoren und $\wedge$, $\vee$

- $\forall x\, F \wedge \forall x\, G \equiv \forall x\, (F \wedge G)$
- $\exists x\, F \vee \exists x\, G \equiv \exists x\, (F \vee G)$

Vorsicht:

- $\forall x\, F \vee \forall x\, G \not\equiv \forall x\, (F \vee G)$:
  - "Alle Hörer sind Frauen oder Alle Hörer sind Männer" vs.
  - "Alle Hörer sind Frauen oder Männer"
- $\exists x\, F \wedge \exists x\, G \not\equiv \exists x\, (F \wedge G)$
  - "Es gibt ein Auto mit Diesel-Motor und es gibt ein Auto mit Elektro-Motor" vs.
  - "Es gibt ein Auto mit Diesel-Motor und Elektro-Motor"

Prädikatenlogik – Fundamentale Äquivalenzen

Quantoren und $\wedge$, $\vee$

Wenn die Formel G die Variable x nicht enthält gilt auch:

- $\forall x\, F \wedge G \equiv \forall x\,(F \wedge G)$
- $\forall x\, F \vee G \equiv \forall x\,(F \vee G)$
- $\exists x\, F \wedge G \equiv \exists x\,(F \wedge G)$
- $\exists x\, F \vee G \equiv \exists x\,(F \vee G)$

Prädikatenlogik – Beispiel

Gegeben: Formel: $F = \forall x \exists y\ \mathit{Invers}(x, y)$
Struktur: $\alpha = (\{2, 3\}, \varphi, \psi, \xi)$ mit $\mathit{Invers}^{\psi}(x, y) = \{(2, 2), (3, 3)\}$.

Frage: Ist α Modell von F?

Um $\alpha(\forall x \exists y\ \mathit{Invers}(x, y))$ zu berechnen betrachte:

- $\alpha_x^2(\exists y\ \mathit{Invers}(x, y))$:
  - $\alpha_{x,y}^{2,2}(\mathit{Invers}(x, y))$=$\mathit{Invers}^{\psi}(2, 2) = 1$
  - $\alpha_{x,y}^{2,3}(\mathit{Invers}(x, y))$=$\mathit{Invers}^{\psi}(2, 3) = 0$

  $\alpha_x^2(\exists y\ \mathit{Invers}(x, y)) = 1$
- $\alpha_x^3(\exists y\ \mathit{Invers}(x, y))$:
  - $\alpha_{x,y}^{3,2}(\mathit{Invers}(x, y))$=$\mathit{Invers}^{\psi}(3, 2) = 0$
  - $\alpha_{x,y}^{3,3}(\mathit{Invers}(x, y))$=$\mathit{Invers}^{\psi}(3, 3) = 1$

  $\alpha_x^3(\exists y\ \mathit{Invers}(x, y)) = 1$

$\alpha(\forall x \exists y\ \mathit{Invers}(x, y)) = 1 \Rightarrow \alpha$ ist Modell von F.

Prädikatenlogik – Beispiel
(zum Selbststudium)

Gegeben:
Formel $F = \forall x \forall y \; \mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))$
Struktur $\alpha = (\{0, 1\}, \varphi, \psi, \xi)$ mit
$\mathit{Gleich}^{\psi}(x, y) = \{(0, 0), (1, 1)\}$ und
plus^{φ} wie in der folgenden Tabelle gegeben

x	y	$\mathit{plus}^{\varphi}(x, y)$
0	0	0
0	1	0
1	0	1
1	1	0

Frage: Ist α Modell von F?

Prädikatenlogik – Beispiel

(zum Selbststudium)

Um $\alpha(\forall x \forall y\ \mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x)))$ zu berechnen betrachte:

- $\alpha_x^0(\forall y\ \mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x)))$:
  - $\alpha_{x,y}^{0,0}(\mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))) = \mathit{Gleich}^\psi(\mathit{plus}^\varphi(0, 0), \mathit{plus}^\varphi(0, 0)) = \mathit{Gleich}^\psi(0, 0) = 1$
  - $\alpha_{x,y}^{0,1}(\mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))) = \mathit{Gleich}^\psi(\mathit{plus}^\varphi(0, 1), \mathit{plus}^\varphi(1, 0)) = \mathit{Gleich}^\psi(0, 1) = 0$

  $\alpha_x^0(\forall y\ \mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))) = 0$ [1]
- $\alpha_x^1(\forall y\ \mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x)))$:
  - $\alpha_{x,y}^{1,0}(\mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))) = \mathit{Gleich}^\psi(1, 0) = 0$
  - $\alpha_{x,y}^{1,1}(\mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))) = \mathit{Gleich}^\psi(0, 0) = 1$

  $\alpha_x^0(\forall y\ \mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))) = 0$

$\alpha(\forall x \forall y\ \mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))) = 0 \Rightarrow \alpha$ ist kein Modell von F.

[1] Wir könnten schon hier auf $\alpha(\forall x \forall y\ \mathit{Gleich}(\mathit{plus}(x, y), \mathit{plus}(y, x))) = 0$ schließen. Der Vollständigkeit halber betrachten wir aber auch die zweite Substitution.

Subsection 6

Formalisieren in Prädikatenlogik

Formalisieren in Prädikatenlogik - Faustregeln

- Zeitwörter bzw. Eigenschaften werden Prädikatensymbole
- Hauptwörter zu ihren Argumenten
- Konkrete Personen, Objekte werden Konstanten

Beispiel

- Sokrates ist sterblich $\qquad Sterblich(sokrates)$
- Thomas läuft $\qquad Laeuft(thomas)$
- Alice spielt Ball $\qquad SpieltBall(alice)$
  $Spielt(alice, ball)$
- Alice spielt Schach $\qquad Spielt(alice, schach)$
- Thomas Mutter spielt Schach $\qquad Spielt(mutter(thomas), schach)$

Formalisieren in Prädikatenlogik - Faustregeln

Regeln: Für alle Objekte mit einer Eigenschaft P gilt, dass ...

$$\forall x(P(x) \rightarrow \dots)$$

Stichworte: *Alle, Jede, ...* manchmal aber auch nur *„wenn ... dann“*

Existenzaussagen: Es gibt ein Objekt mit der Eigenschaft P, dass ...

$$\exists x(P(x) \wedge \dots)$$

Stichworte: *existiert, es gibt, (mindestens/hat/...) einen, ...*

Beispiel

Aussage: Jeder Tennisspieler besitzt einen Tennisschläger.

$$\forall x \left(TSpieler(x) \rightarrow (\exists y(Schlaeger(y) \wedge Besitzt(x,y)))\right)$$

oder auch $\forall x \, \exists y \left(TSpieler(x) \rightarrow (Schlaeger(y) \wedge Besitzt(x,y))\right)$.

Prädikatenlogik – Beispiel 1

Aussage: „Wenn zwei Zahlen negativ sind dann ist ihr Produkt positiv“

$$\forall x \forall y \left((\mathit{Negativ}(x) \wedge \mathit{Negativ}(y)) \to \mathit{Positiv}(\mathit{produkt}(x, y)) \right)$$

Ein Modell:

- $U = \{-1, 1\}$
- $\mathit{produkt}^\varphi$: $\mathit{produkt}^\varphi(1, 1) = \mathit{produkt}^\varphi(-1, -1) = 1$, $\mathit{produkt}^\varphi(-1, 1) = \mathit{produkt}^\varphi(1, -1) = -1$
- $\mathit{Negativ}^\psi = \{-1\}$, $\mathit{Positiv}^\psi = \{1\}$
- $x^\xi = 1$, $y^\xi = 1$

Ein anderes Modell wären die ganzen Zahlen $\mathbb{Z}$, wobei $\mathit{produkt}^\varphi(.,.)$ die übliche Multiplikation ist und $\mathit{Negativ}^\psi = \mathbb{Z}^-$, $\mathit{Positiv}^\psi = \mathbb{Z}^+$.

Prädikatenlogik – Beispiel 2

Aussage: „Wenn es einen Weg von A nach B gibt und einen Weg von B nach C gibt dann gibt es auch einen Weg von A nach C.“

$$\forall x \forall y \forall z \big((Weg(x, y) \wedge Weg(y, z)) \rightarrow Weg(x, z) \big)$$

oder semantisch äquivalent

$$\forall x \forall y \forall z \big(\neg Weg(x, y) \vee \neg Weg(y, z) \vee Weg(x, z) \big)$$

Ein **Modell**:

- $U = \{Wien, Berlin, Dresden, Eisenstadt\}$
- keine Funktionen/Konstanten
- $Weg^{\psi} = \{(Wien, Eisenstadt)\}$,
- $x^{\xi} = Wien$, $y^{\xi} = Berlin$, $z^{\xi} = Dresden$

Beispiel 3

Wir wollen mehrere Aussagen in Prädikatenlogik formalisieren:

1. Nicht alle Musiker sind berühmt.
2. Es gibt berühmte Personen die keine Musiker sind.
3. Ein Musiker ist genau dann berühmt wenn er gut ist.
4. Es existieren sowohl schlechte als auch gute Musiker.

Die „Objekte“ sind in diesem Fall Personen.

Wir erkennen mehrere Eigenschaften: ist Musiker, ist berühmt, ist gut, ist schlecht.

$\hookrightarrow$ wir nutzen die folgenden Prädikate:

- $Musiker(x) \ldots x$ ist Musiker
- $Beruehmt(x) \ldots x$ ist berühmt
- $Gut(x) \ldots x$ ist gut
- $Schlecht(x) \ldots x$ ist schlecht

Beispiel 3

Wir wollen mehrere Aussagen in Prädikatenlogik formalisieren:

1. Nicht alle Musiker sind berühmt.
2. Es gibt berühmte Personen die keine Musiker sind.
3. Ein Musiker ist genau dann berühmt wenn er gut ist.
4. Es existieren sowohl schlechte als auch gute Musiker.

1. $\neg\forall x(Musiker(x) \to Beruehmt(x))$ oder
   $\exists x(Musiker(x) \land \neg Beruehmt(x))$
2. $\exists x(Beruehmt(x) \land \neg Musiker(x))$
3. $\forall x(Musiker(x) \to (Beruehmt(x) \leftrightarrow Gut(x)))$
4. $\exists x(Musiker(x) \land Gut(x)) \land \exists y(Musiker(y) \land Schlecht(y))$

Beispiel 3

Wir wollen zeigen dass die Aussagen miteinander konsistent sind.

1. $F_1 = \neg\forall x(\mathit{Musiker}(x) \rightarrow \mathit{Beruehmt}(x))$
2. $F_2 = \exists x(\neg\mathit{Musiker}(x) \wedge \mathit{Beruehmt}(x))$
3. $F_3 = \forall x(\mathit{Musiker}(x) \rightarrow (\mathit{Beruehmt}(x) \leftrightarrow \mathit{Gut}(x)))$
4. $F_4 = \exists x(\mathit{Musiker}(x) \wedge \mathit{Gut}(x)) \wedge \exists y(\mathit{Musiker}(y) \wedge \mathit{Schlecht}(y))$

Um zu zeigen, dass $F_1 \wedge F_2 \wedge F_3 \wedge F_4$ erfüllbar ist geben wir ein Modell an.

Ein Modell:

$\alpha = (\{\mathit{Uli}, \mathit{Tom}, \mathit{Bob}\}, \varphi, \psi, \xi)$

- $\mathit{Musiker}^{\psi}(x) = \{\mathit{Uli}, \mathit{Tom}\}$
- $\mathit{Beruehmt}^{\psi}(x) = \{\mathit{Uli}, \mathit{Bob}\}$
- $\mathit{Gut}^{\psi}(x) = \{\mathit{Uli}\}$
- $\mathit{Schlecht}\psi(x) = \{\mathit{Tom}\}$
- $x^{\xi} = \mathit{Tom}$, $y^{\xi} = \mathit{Tom}$

Ein anderes Modell:

$\alpha' = (\{0, 1, 2\}, \varphi, \psi, \xi)$

- $\mathit{Musiker}^{\psi}(x) = \{0, 1\}$
- $\mathit{Beruehmt}^{\psi}(x) = \{0, 2\}$
- $\mathit{Gut}^{\psi}(x) = \{0\}$
- $\mathit{Schlecht}^{\psi}(x) = \{1\}$
- $x^{\xi} = 0$, $y^{\xi} = 0$

Zusammenfassung & Ausblick

Bis jetzt haben wir Folgendes behandelt:

- Resolutionsverfahren der Aussagenlogik
- Hornlogik und Einheitsresolution
- Formale Definition der Prädikatenlogik: Syntax
- Formale Definition der Prädikatenlogik: Semantik
- Fundamentale Begriffe und Sätze
- Formalisieren in Prädikatenlogik

Weiter geht es mit:

- Normalformen in der Prädikatenlogik
- Unentscheidbarkeit der Prädikatenlogik
- Resolutionskalkül der Prädikatenlogik