Prädikatenlogik: Normalformen

Normalformen basierend auf Äquivalenz

- Äquivalenz in der Prädikatenlogik
- Übertragung aussagenlogischer Äquivalenzen
- Formeln mit Quantoren: Äquivalenzen in der Prädikatenlogik
- Substitution (Überführungslemma)
  - Gebundene Umbenennung von Variablen
- Pränexform

Normalformen basierend auf Erfüllbarkeitsäquivalenz

- Erfüllbarkeitsäquivalenz vs. Äquivalenz
- Bindung freier Variablen
- Skolemisierung
- Skolemform und Klauselnormalform
  - Normalform, die Grundlage für Resolutionsbeweise ist [→ Kap. 11]

Äquivalenz in der Prädikatenlogik

zur Erinnerung: Definition 9.12 (ganz analog zur Aussagenlogik)

Zwei Formeln F und G sind *(logisch)* *äquivalent*, falls für jede Struktur $\mathcal{A}$ gilt:
$\mathcal{A}(F) = \mathcal{A}(G)$.

Satz 10.1

Wenn in zwei äquivalenten aussagenlogischen Formeln prädikatenlogische Formeln (uniform) für die Aussagensymbole substituiert werden, ergibt dies zwei äquivalente Formeln der Prädikatenlogik.
ohne Beweis, vgl. Satz 6.3
→ Die Äquivalenzen der Aussagenlogik gelten auch in der Prädikatenlogik.

Beispiel zu 10.1

Kommutativität von $\wedge$: $(\forall x\, P(x) \wedge \exists y\, Q(y)) \equiv (\exists y\, Q(y) \wedge \forall x\, P(x))$
Elimination von $\Rightarrow$: $(\forall x\, P(x) \Rightarrow \exists y\, Q(y)) \equiv (\neg\forall x\, P(x) \vee \exists y\, Q(y))$

→ Zusätzliche prädikatenlogische Äquivalenzen ergeben sich aus den semantischen Beziehungen zwischen Quantoren.

Wichtige Äquivalenzen der Prädikatenlogik

Satz 10.2: Seien F und G beliebige prädikatenlogische Formeln.

1. $\neg\forall x\, F \equiv \exists x\, \neg F$ — *Dualität* von $\forall$ und $\exists$
   $\neg\exists x\, F \equiv \forall x\, \neg F$
2. Falls x in G nicht frei vorkommt
   $G \equiv \forall x\, G \equiv \exists x\, G$ — *leere Quantifikation*
   $(\forall x\, F \wedge G) \equiv \forall x\, (F \wedge G)$ — *Skopuserweiterung* wenn damit keine neuen Variablen-Bindungen entstehen
   $(\forall x\, F \vee G) \equiv \forall x\, (F \vee G)$
   $(\exists x\, F \wedge G) \equiv \exists x\, (F \wedge G)$
   $(\exists x\, F \vee G) \equiv \exists x\, (F \vee G)$
   $(G \Rightarrow \forall x\, F) \equiv \forall x\, (G \Rightarrow F)$
   $(G \Rightarrow \exists x\, F) \equiv \exists x\, (G \Rightarrow F)$
   $(\forall x\, F \Rightarrow G) \equiv \exists x\, (F \Rightarrow G)$
   $(\exists x\, F \Rightarrow G) \equiv \forall x\, (F \Rightarrow G)$
3. $(\forall x\, F \wedge \forall x\, G) \equiv \forall x\, (F \wedge G)$ — *Distributivität* von Allquantor bzgl. Konjunktion
   $(\exists x\, F \vee \exists x\, G) \equiv \exists x\, (F \vee G)$ — Existenzquantor bzgl. Disjunktion
4. $\forall x\, \forall y\, F \equiv \forall y\, \forall x\, F$ — *Vertauschung* der *Quantorenreihenfolge* nur bei gleichen Quantoren
   $\exists x\, \exists y\, F \equiv \exists y\, \exists x\, F$

Beweis einiger Äquivalenzen

Satz 10.2.2.1: Wenn x nicht frei in G vorkommt, dann $G \equiv \forall x\, G$ und $G \equiv \exists x\, G$.

Beweis

Sei $\mathcal{A} = (U, I)$ eine Struktur.

$\mathcal{A}(G) = 1$ GDW. für alle $d \in U$ gilt: $\mathcal{A}_{[x/d]}(G) = 1$ — [da x nicht frei in G]

GDW. $\mathcal{A}(\forall x\, G) = 1$ — [Auswertung des Allquantors]

$\mathcal{A}(G) = 1$ GDW. für ein $d \in U$ gilt: $\mathcal{A}_{[x/d]}(G) = 1$ — [da U nicht leer und x nicht frei in G]

GDW. $\mathcal{A}(\exists x\, G) = 1$ — [Auswertung des Existenzquantors]

Satz 10.2.2.2: Wenn x nicht frei in G vorkommt, dann $(\forall x\, F \wedge G) \equiv \forall x\, (F \wedge G)$.

Beweis

Sei $\mathcal{A} = (U, I)$ eine Struktur.

$\mathcal{A}(\forall x\, F \wedge G) = 1$ GDW. $\mathcal{A}(\forall x\, F) = 1$ und $\mathcal{A}(G) = 1$ — [Auswertung der Konjunktion]

GDW. für alle $d \in U$ gilt: $\mathcal{A}_{[x/d]}(F) = 1$ und $\mathcal{A}(G) = 1$ — [Auswertung des Allquantors]

GDW. für alle $d \in U$ gilt: $\mathcal{A}_{[x/d]}(F) = 1$ und $\mathcal{A}_{[x/d]}(G) = 1$ — [da x nicht frei in G]

GDW. für alle $d \in U$ gilt: $\mathcal{A}_{[x/d]}(F \wedge G) = 1$ — [Auswertung der Konjunktion]

GDW. $\mathcal{A}(\forall x\, (F \wedge G)) = 1$ — [Auswertung des Allquantors]

Einige Nicht-Äquivalenzen

Satz 10.3

a) $(\forall x\ F \vee \forall x\ G) \not\equiv \forall x\ (F \vee G)$

$(\exists x\ F \wedge \exists x\ G) \not\equiv \exists x\ (F \wedge G)$

b) $\forall x\ \exists y\ F \not\equiv \exists y\ \forall x\ F$

Beweisidee:

Durch Gegenbeispiele
zu a) Domäne: natürliche Zahlen: $F \approx x$ ist ungerade, $G \approx x$ ist gerade
zu b) Domäne: natürliche Zahlen: $F \approx y$ ist direkter Nachfolger von x

Zum Selbststudium: Äquivalenzen – Nicht-Äquivalenzen

- Um Äquivalenz von Formeln F und G zu beweisen, ist es notwendig, beliebige Strukturen $\mathcal{A}$ (d.h. insbesondere über beliebigen Domänen) zu berücksichtigen, genauer (vgl. Def. 9.12), zu zeigen, dass für jede Struktur $\mathcal{A}$ gilt:
  $\mathcal{A}(F) = \mathcal{A}(G)$.
  [Dies geschieht in den Beweisen zu Satz 10.2].
- Um die Nicht-Äquivalenz der Formeln F und G zu beweisen, reicht es aus, eine Struktur $\mathcal{A}$ zu finden, für die $\mathcal{A}(F) \neq \mathcal{A}(G)$.
  [Für Satz 10.3 gibt die Beweisidee den Hinweis auf entsprechende Strukturen; ein ausgearbeiteter Beweis für 10.3 beinhaltet sowohl die explizite Spezifikation der Strukturen als auch den Nachweis der Verschiedenheit der Auswertung, d.h. die Begründung für $\mathcal{A}(F) \neq \mathcal{A}(G)$.]

- Entsprechendes gilt für Beweise von
  - Unerfüllbarkeit – Nicht-Unerfüllbarkeit
  - Folgerbarkeit – Nicht-Folgerbarkeit

Ersetzbarkeitstheorem (Prädikatenlogik)

Satz 10.4

Seien F und G äquivalente Formeln und sei H eine Formel mit Teilformel F.
Sei H' eine Formel, die aus H durch Ersetzung von F durch G hervorgeht.

Dann gilt: $H' \equiv H$.

Beweis: s. Satz 4.9 (folgt dem Prinzip der strukturellen Induktion):
Beim Induktionsschritt sind zusätzlich Formeln mit Quantoren zu berücksichtigen.
Der entscheidende Punkt ist, zu zeigen:

Hilfssatz 10.5

Wenn H_1 und H_1' äquivalente Formeln sind, dann $\exists x\, H_1 \equiv \exists x\, H_1'$ und $\forall x\, H_1 \equiv \forall x\, H_1'$.

Beweis

Sei $H_1 \equiv H_1'$ und $\mathcal{A}$ ein Modell von $\exists x\, H_1$. Dann gibt es ein $d \in U_{\mathcal{A}}$, so dass $\mathcal{A}_{[x/d]}(H_1) = \mathbf{1}$. Da $H_1 \equiv H_1'$, ist $\mathcal{A}_{[x/d]}(H_1') = \mathbf{1}$ und damit ist $\mathcal{A}$ ein Modell von $\exists x\, H_1'$.
Der zweite Teil des Beweises funktioniert analog.

Beispiel: Äquivalenz-Umformungen

	$(\neg(\exists x\, P(x, y) \vee \forall z\, Q(z)) \wedge \exists w\, P(f(a), w))$	
$\equiv$	$((\neg\exists x\, P(x, y) \wedge \neg\forall z\, Q(z)) \wedge \exists w\, P(f(a), w))$	de Morgan
$\equiv$	$((\forall x\, \neg P(x, y) \wedge \exists z\, \neg Q(z)) \wedge \exists w\, P(f(a) ,w))$	[1] Dualität
$\equiv$	$(\exists w\, P(f(a), w) \wedge (\forall x\, \neg P(x, y) \wedge \exists z\, \neg Q(z)))$	Kommutativität $\wedge$
$\equiv$	$\exists w\, (P(f(a), w) \wedge (\forall x\, \neg P(x, y) \wedge \exists z\, \neg Q(z)))$	[2] Skopus von $\exists w$
$\equiv$	$\exists w\, (P(f(a), w) \wedge \forall x\, (\neg P(x, y) \wedge \exists z\, \neg Q(z)))$	[2] Skopus von $\forall x$
$\equiv$	$\exists w\, (\forall x\, (\neg P(x, y) \wedge \exists z\, \neg Q(z)) \wedge P(f(a), w))$	Kommutativität $\wedge$
$\equiv$	$\exists w\, (\forall x\, (\exists z\, \neg Q(z) \wedge \neg P(x, y)) \wedge P(f(a), w))$	Kommutativität $\wedge$
$\equiv$	$\exists w\, (\forall x\, \exists z\, (\neg Q(z) \wedge \neg P(x, y)) \wedge P(f(a), w))$	[2] Skopus von $\exists z$
$\equiv$	$\exists w\, \forall x\, \exists z\, (\neg Q(z) \wedge \neg P(x, y) \wedge P(f(a), w))$	[2] Skopus von $\forall x$ und $\exists z$

➔ Quantorenreihenfolge nach der Umformung hängt von der Reihenfolge der Umformungen ab

➔ Skopuserweiterung nach [2] ist nur möglich, wenn in der neu eingeschlossenen Formel nicht die Variable auftritt, die durch den betroffenen Quantor gebunden ist.

Normalform für Prädikatenlogik

Definition 10.6

- Eine Formel F heißt *bereinigt*, falls es keine Variable in F gibt, die in der Formel sowohl gebunden als auch frei vorkommt und falls alle Quantoren in F unterschiedliche Quantorenvariablen aufweisen.
- Eine Formel F heißt pränex (oder in Pränexform), falls sie die folgende Form aufweist: $F = \mathcal{Q}_1 y_1 \mathcal{Q}_2 y_2 \ldots \mathcal{Q}_n y_n\ G$, wobei $\mathcal{Q}_i \in \{\exists, \forall\}$, $n \geq 0$, die y_i Variablen sind und in G keine Quantoren vorkommen. $\mathcal{Q}_1 y_1 \mathcal{Q}_2 y_2 \ldots \mathcal{Q}_n y_n$ heißt (Quantoren-) *Präfix*, G ist die *Matrix* von F.
- Ist eine Formel G bereinigt, in Pränexform und äquivalent zur Formel F, dann nennen wir G eine BPF zu F.

Das nächste Ziel

- Satz 10.10: Zu jeder Formel gibt es eine BPF

Dafür erforderlich

- Möglichkeit der Umbenennung von Variablen
  - Definition Variablensubstitution
  - Überführungslemma (semantische Effekte der Variablensubstitution)
  - Satz zur gebundenen Umbenennung

Variablen-Substitution (1)

Definition 10.7 (einfache (Variablen-) Substitution)

Seien F eine Formel, x eine Variable und t ein Term. $F_{[x/t]}$ ist die Formel, die sich aus F ergibt, wenn jedes freie Vorkommen der Variablen x in F durch den Term t ersetzt wird. $[x/t]$ wird als *Substitution* bezeichnet.

Explizit als rekursive Funktion definiert:

Eine einfache *(Variablen-) Substitution* ist eine rekursiv definierte Funktion $[x/t]$.

- Die Variable x wird auf den Term t abgebildet.
- Alle anderen Variablen und Konstanten werden auf sich selbst abgebildet.
- Für komplexe Terme gilt: $[x/t](f(t_1, \ldots, t_k)) = f([x/t](t_1), \ldots, [x/t](t_k))$
- Für atomare Formeln gilt: $[x/t](P(t_1, \ldots, t_k)) = P([x/t](t_1), \ldots, [x/t](t_k))$
- Für komplexe Formeln gilt:
  $[x/t](\neg F) = \neg[x/t](F)$
  $[x/t]((F \wedge G)) = ([x/t](F) \wedge [x/t](G))$ $\quad$ $[x/t]((F \vee G)) = ([x/t](F) \vee [x/t](G))$
  $[x/t]((F \Rightarrow G)) = ([x/t](F) \Rightarrow [x/t](G))$ $\quad$ $[x/t]((F \Leftrightarrow G)) = ([x/t](F) \Leftrightarrow [x/t](G))$
  $[x/t](\forall x\ F) = \forall x\ F$ $\quad$ $[x/t](\exists x\ F) = \exists x\ F$
  $[x/t](\forall y\ F) = \forall y\ [x/t](F)$ $\quad$ $[x/t](\exists y\ F) = \exists y\ [x/t](F)$, wobei $x \neq y$.
- Statt $[x/t](F)$ wird auch $F_{[x/t]}$ geschrieben.

~~Zum Selbststudium:~~ (Variablen-)Substitution (2)

Definition 10.7x ((Variablen-) Substitution)

Eine *(Variablen-) Substitution* ist eine rekursiv definierte Funktion σ.

- Variablen x_i werden auf Terme $\sigma(x_i)$ abgebildet.
- Konstanten a_i werden auf sich selbst abgebildet.
- Für komplexe Terme gilt: $\sigma(f(t_1, ..., t_k)) = f(\sigma(t_1), ..., \sigma(t_k))$
- Für atomare Formeln gilt: $\sigma(P(t_1, ..., t_k)) = P(\sigma(t_1), ..., \sigma(t_k))$
- Für komplexe Formeln gilt:

  $\sigma(\neg F) = \neg\sigma(F)$

  $\sigma((F \wedge G)) = (\sigma(F) \wedge \sigma(G))$ $\qquad$ $\sigma((F \vee G)) = (\sigma(F) \vee \sigma(G))$

  $\sigma((F \Rightarrow G)) = (\sigma(F) \Rightarrow \sigma(G))$ $\qquad$ $\sigma((F \Leftrightarrow G)) = (\sigma(F) \Leftrightarrow \sigma(G))$

  $\sigma(\forall x\, F) = \forall x\, \sigma_x(F)$ $\qquad$ $\sigma(\exists x\, F) = \exists x\, \sigma_x(F)$
- Dabei ist σ_x diejenige Substitution, die sich von σ nur dadurch unterscheidet, dass die Variable x auf sich selbst abgebildet wird.

➔ Durch eine (Variablen-)Substitution wird jede Formel F auf eine Formel $\sigma(F)$ abgebildet, wobei *jedes* Vorkommen einer freien Variablen x_i in F durch den entsprechenden Term $\sigma(x_i)$ ersetzt wird.

➔ Quantorenvariablen und durch sie gebundene Vorkommen von Variablen werden durch die Variablen-Substitution nicht ersetzt.

Überführungslemma

Satz 10.8 (Überführungslemma)

Sei F eine Formel, x eine Variable und t ein Term, der keine in F gebundene Variable enthält. Dann gilt für jede Struktur $\mathcal{A}$:

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

Beweis: siehe Schöning: Übung 58.

- Das Überführungslemma stellt den Zusammenhang zwischen Substitutionen und Modellen (bzw. Auswertungen) her:

$\mathcal{A}(F_{[x/t]})$ Auswertung der Formel, die aus F durch Substitution von x durch t hervorgeht

$\mathcal{A}_{[x/\mathcal{A}(t)]}(F)$ Auswertung von F bezüglich der x-Variante, in der x zu $\mathcal{A}(t)$ ausgewertet wird.

$$\begin{array}{ccc}
x & \xrightarrow{[x/t]} & t \\
\downarrow \mathcal{A}_{[x/\mathcal{A}(t)]} & & \downarrow \mathcal{A} \\
\mathcal{A}_{[x/\mathcal{A}(t)]}(x) & = & \mathcal{A}(t)
\end{array}
\qquad
\begin{array}{ccc}
F & \xrightarrow{[x/t]} & F_{[x/t]} \\
\downarrow \mathcal{A}_{[x/\mathcal{A}(t)]} & & \downarrow \mathcal{A} \\
\mathcal{A}_{[x/\mathcal{A}(t)]}(F) & = & \mathcal{A}(F_{[x/t]})
\end{array}$$

Gebundene Umbenennung

Satz 10.9 (gebundene Umbenennung)

Sei F eine Formel und y eine Variable, die in F nicht vorkommt.

Dann gilt: $\exists x\ F \equiv \exists y\ F_{[x/y]}$ und $\forall x\ F \equiv \forall y\ F_{[x/y]}$

Beweis

Sei $\mathcal{A}$ eine Struktur.

$\mathcal{A}(\exists y\ F_{[x/y]}) = 1$

GDW. es ein $d \in U$ gibt, so dass $\mathcal{A}_{[y/d]}(F_{[x/y]}) = 1$ [Quantor-Interpretation]

GDW. es ein $d \in U$ gibt, so dass $\mathcal{A}_{[y/d][x/d]}(F) = 1$ [Überführungslemma]

GDW. $\mathcal{A}_{[y/d]}(\exists x\ F) = 1$ [Quantor-Interpretation]

GDW. $\mathcal{A}(\exists x\ F) = 1$ [da y nicht frei in F vorkommt]

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

GDW. für alle $d \in U$ gilt $\mathcal{A}_{[y/d]}(F_{[x/y]}) = 1$ [Quantor-Interpretation]

GDW. für alle $d \in U$ gilt $\mathcal{A}_{[y/d][x/d]}(F) = 1$ [Überführungslemma]

GDW. für alle $d \in U$ gilt $\mathcal{A}_{[x/d]}(F) = 1$ [da y nicht frei in F vorkommt]

GDW. $\mathcal{A}(\forall x\ F) = 1$ [Quantor-Interpretation]

Achtung: Umbenennung *freier* Variablen erhält die Äquivalenz nicht !

Existenz einer äquivalenten bereinigten Pränexform

Satz 10.10 (Existenz einer äquivalenten Formel in BPF)

Für jede Formel F gibt es eine äquivalente Formel G in bereinigter Pränexform.

Beweis

Vorbemerkung:

Gemäß Satz 4.10 gibt es zu jeder Formel eine äquivalente Formel, in der $\Leftrightarrow$ nicht vorkommt. Der Beweis setzt entsprechend voraus, dass $\Leftrightarrow$ in F nicht vorkommt.

Induktionsanfang

Wenn F eine atomare Formel ist, dann hat F Pränexform mit leerem Präfix (n = 0).

Induktionsannahme

Es seien F_1 und F_2 Formeln, zu denen es äquivalente bereinigte Pränexformen $\mathcal{Q}_1 x_1 \mathcal{Q}_2 x_2 \ldots \mathcal{Q}_n x_n\ G_1$ bzw. $\mathcal{Q}'_1 y_1 \mathcal{Q}'_2 y_2 \ldots \mathcal{Q}'_m y_m\ G_2$ gibt.

Aufgrund von Satz 10.9 können wir weiterhin voraussetzen, dass $x_i \neq y_j$ für alle i und j sind und dass keine in F_1 bzw. F_2 freien Variable zu den x_i bzw. y_j gehört.

Induktionsschritt: nächste Folie.

Pränexform: Induktionsschritt

Zur Erinnerung: $F_1 \equiv \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\, G_1$ und $F_2 \equiv \mathcal{Q}'_1y_1\mathcal{Q}'_2y_2 \ldots \mathcal{Q}'_my_m\, G_2$

Sei $\overline{\mathcal{Q}} = \begin{cases} \exists, \text{ falls } \mathcal{Q} = \forall \\ \forall, \text{ falls } \mathcal{Q} = \exists \end{cases}$.

Dann gilt: $\neg F_1 \equiv \overline{\mathcal{Q}}_1x_1\overline{\mathcal{Q}}_2x_2 \ldots \overline{\mathcal{Q}}_nx_n\, \neg G_1$ [Satz 10.2.1]

$F_1 \wedge F_2 \equiv \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\, G_1 \wedge \mathcal{Q}'_1y_1\mathcal{Q}'_2y_2 \ldots \mathcal{Q}'_my_m\, G_2$

$\equiv \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\mathcal{Q}'_1y_1\mathcal{Q}'_2y_2 \ldots \mathcal{Q}'_my_m\, (G_1 \wedge G_2)$ [Satz 10.2.2]

$F_1 \vee F_2 \equiv \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\, G_1 \vee \mathcal{Q}'_1y_1\mathcal{Q}'_2y_2 \ldots \mathcal{Q}'_my_m\, G_2$

$\equiv \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\mathcal{Q}'_1y_1\mathcal{Q}'_2y_2 \ldots \mathcal{Q}'_my_m\, (G_1 \vee G_2)$ [Satz 10.2.2]

$F_1 \Rightarrow F_2 \equiv \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\, G_1 \Rightarrow \mathcal{Q}'_1y_1\mathcal{Q}'_2y_2 \ldots \mathcal{Q}'_my_m\, G_2$

$\equiv \overline{\mathcal{Q}}_1x_1\overline{\mathcal{Q}}_2x_2 \ldots \overline{\mathcal{Q}}_nx_n\, \mathcal{Q}'_1y_1\mathcal{Q}'_2y_2 \ldots \mathcal{Q}'_my_m\, (G_1 \Rightarrow G_2)$ [Satz 10.2.2]

$\exists x\, F_1 \equiv \exists x\, \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\, G_1$ [BPN, wenn $x \neq x_i$ für alle i]

$\forall x\, F_1 \equiv \forall x\, \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\, G_1$ [BPN, wenn $x \neq x_i$ für alle i]

wenn $x = x_i$ für ein i, dann kommt x nicht frei in $\mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\, G_1$ vor und

$\exists x\, F_1 \equiv \forall x\, F_1 \equiv \mathcal{Q}_1x_1\mathcal{Q}_2x_2 \ldots \mathcal{Q}_nx_n\, G_1$ [BPN]

Normalformen: Zwischenstand

Äquivalenzumformungen führen zu bereinigten Pränexformen

- die Matrix lässt sich in KNF oder DNF bringen.
- im Präfix können aber Allquantoren und Existenzquantoren gemischt auftreten.
- In einer Formel können sowohl freie als auch gebundene Variablen auftreten.
- Durch Äquivalenzumformungen läßt sich dieses nicht aufheben.

Die nächsten Schritte

- Bindung freier Variablen durch Existenzquantoren.
- Elimination der Existenzquantoren im Präfix durch Skolemisierung.
- Beide Umformungsschritte gewährleisten zwar nicht Äquivalenz aber (wenigstens) Erfüllbarkeitsäquivalenz.

Erfüllbarkeitsäquivalenz

Definition 10.11 (Erfüllbarkeitsäquivalenz)

Zwei Formeln F und G sind genau dann *erfüllbarkeitsäquivalent*, wenn gilt:
F ist erfüllbar GDW. G erfüllbar ist.

Beobachtung zu Definition 10.11

- F und G sind erfüllbarkeitsäquivalent: F ist unerfüllbar GDW. G unerfüllbar ist.
- F und G sind erfüllbarkeitsäquivalent:
  Es gibt eine erfüllende Struktur $\mathcal{A}$ zu F GDW. es eine erfüllende Struktur $\mathcal{A}'$ zu G gibt.
- Erfüllbarkeitsäquivalenz ist schwächer als Äquivalenz,
  d.h. Äquivalenz schließt Erfüllbarkeitsäquivalenz ein.
- Äquivalenz betrifft gleiche Bewertung durch alle Strukturen. Erfüllbarkeitsäquivalenz betrifft die Existenz von erfüllenden Strukturen. (Diese können verschieden sein.)

Umformungen, die Erfüllbarkeitsäquivalenz garantieren

- sind als Vorstufe von Widerlegungsverfahren nützlich ➔ Resolution (Kapitel 12)

Zur Erinnerung: Reduktion semantischer Fragen auf (Un)Erfüllbarkeit

Sätze

(gelten in der Prädikatenlogik ebenso wie in der Aussagenlogik)

- Formel F ist genau dann eine Tautologie ($\vDash$ F), wenn $\neg$F unerfüllbar ist.
- Formel F folgt genau dann aus Formel G (G $\vDash$ F), wenn (G $\land$ $\neg$F) unerfüllbar ist.
- Formeln F und G sind genau dann äquivalent (G $\equiv$ F), wenn $\neg$(F $\Leftrightarrow$ G) unerfüllbar ist.

Konsequenz

- Jedes Verfahren, das der Feststellung der Erfüllbarkeit / Unerfüllbarkeit von Formeln dient, ist auch verwendbar, um Gültigkeit, Folgerung und Äquivalenz festzustellen.
- Allerdings muss das Verfahren dazu auf geeignet gebildete Formeln angewendet werden.
- Entsprechend muss auch die Vorverarbeitung auf die geeignete (und das ist nicht immer der ursprüngliche) Formel angewendet werden.
- Nachdem die ursprüngliche Frage in eine (Un)Erfüllbarkeitsfrage bzgl. einer Formel umgewandelt wurde, können auch (unbesorgt) Umformungen beruhend auf Erfüllbarkeitsäquivalenz vorgenommen werden.

Beispiel: Erfüllbarkeitsäquivalenz

Satz 10.12

Sei F eine Formel und x eine Variable. F ist genau dann erfüllbar, wenn $\exists x\, F$ erfüllbar ist.
➔ maW. F und $\exists x\, F$ sind erfüllbarkeitsäquivalent.

Beweis

Sei $\mathcal{A}$ ein Modell von F.

Dann gilt $\mathcal{A}_{[x/\mathcal{A}(x)]}(F) = \mathcal{A}(F) = \mathbf{1}$ und damit $\mathcal{A}(\exists x\, F) = \mathbf{1}$

Sei $\mathcal{A}$ ein Modell von $\exists x\, F$.

Dann gibt es ein $d \in U$, so dass $\mathcal{A}_{[x/d]}(F) = \mathbf{1}$. Also ist auch F erfüllbar.

Satz 10.13

Zu jeder prädikatenlogischen Formel gibt es eine erfüllbarkeitsäquivalente geschlossene Formel.

Beweisidee

Sei F eine Formel und seien $x_1,\ldots, x_n$ die in F frei vorkommenden Variablen.

F und $\exists x_1 \ldots \exists x_n\, F$ sind erfüllbarkeitsäquivalent.

(Dies ist durch vollständige Induktion über n zu zeigen.)

Beispiel: Erfüllbarkeitsäquivalenz

Satz 10.14

Sei F eine Formel, x eine Variable und a eine Konstante, die nicht in F vorkommt.
Dann sind $F_{[x/a]}$ und $\exists x\, F$ erfüllbarkeitsäquivalent.

Beweis

Sei $\mathcal{A}$ ein Modell von $F_{[x/a]}$.

Dann gilt $\mathcal{A}_{[x/\mathcal{A}(a)]}(F) = \mathcal{A}(F_{[x/a]}) = \mathbf{1}$ und damit $\mathcal{A}(\exists x\, F) = \mathbf{1}$

Sei $\mathcal{B}$ ein Modell von $\exists x\, F$.

Dann gibt es ein $d \in U$, so dass $\mathcal{B}_{[x/d]}(F) = \mathbf{1}$. Die Struktur $\mathcal{A}$ sei so definiert, dass sie nur in der Interpretation von a von $\mathcal{B}$ abweicht. Hier gilt: $\mathcal{A}(a) = d$. Da a nicht in F vorkommt, spielt $\mathcal{B}(a) = \mathcal{B}_{[x/d]}(a)$ für die Bestimmung von $\mathcal{B}_{[x/d]}(F)$ keine Rolle, also ist $\mathcal{B}_{[x/d]}(F) = \mathcal{A}_{[x/d]}(F) = \mathcal{A}_{[x/\mathcal{A}(a)]}(F)$.

Wegen des Überführungslemmas gilt weiter: $\mathcal{A}_{[x/\mathcal{A}(a)]}(F) = \mathcal{A}(F_{[x/a]})$.

Also ist auch $F_{[x/a]}$ erfüllbar.

Beobachtung zu den Sätzen 10.12, 10.13, 10.14

- Die Sätze machen nur Aussagen über Existenzquantoren mit maximalem Skopus.

➔ Skolemisierung: Verallgemeinerung von Satz 10.14 für Existenzquantoren an beliebiger Position im Quantorenpräfix

Skolemisierung

Definition 10.15 (Skolemisierung, Skolemsymbol)

Es sei $F = \forall y_1 \forall y_2 \ldots \forall y_k \exists z\ G$, $k \geq 0$ und f ein k-stelliges Funktionssymbol, das nicht in F vorkommt.

Wir bilden die Formel:

$$F' = \forall y_1 \forall y_2 \ldots \forall y_k\ G_{[z/f(y1,\ \ldots,\ yk)]}$$

- f heißt ***Skolemsymbol***,
  wenn $k > 0$ auch ***Skolemfunktion***, im Fall $k = 0$ auch ***Skolemkonstante***.
  [Toralf Skolem]
- Skolemsymbole sind von allen bisher verwendeten Funktionssymbolen und Konstanten verschieden.
- Die Bildung von F' heißt **Skolemisierung** von F.

Beobachtungen zu Definition 10.15

- F' enthält einen Existenzquantor weniger als F.
- Die durch diesen Existenzquantor gebundene Variable z tritt nicht mehr in F' auf.
- z wurde in G durch $f(y_1, \ldots, y_k)$ substituiert.
- Ist F eine Formel in BPF und F' durch Skolemisierung aus F hervorgegangen, dann ist auch F' in BPF.

Beispiele: Skolemisierung

k=0	$\exists x\ \forall y\ (vorf(x,y) \wedge weibl(x))$ $\forall y\ (vorf(e, y) \wedge weibl(e))$	Skolemkonstante e
k=1	$\forall y\ \exists x\ (elt(x, y) \wedge weibl(x))$ $\forall y\ (elt(m(y), y) \wedge weibl(m(y)))$	Skolemfunktion $m(y)$
	$\forall y\ \exists x\ (vorf(x, y) \wedge weibl(x))$ $\forall y\ (vorf(f(y), y) \wedge weibl(f(y)))$	Skolemfunktion $f(y)$
k=2	$\forall x\ \forall y\ \exists z\ (gr\_elt(x, y) \Rightarrow (elt(x, z) \wedge elt(z, y)))$ $\forall x\ \forall y\ (gr\_elt(x, y) \Rightarrow (elt(x, g(x, y)) \wedge elt(g(x, y),y)))$	Skolemfunktion $g(x, y)$

➔ Vorsicht: Die Einführung der Skolemsymbole besagt noch nichts über die Interpretation der Formeln, die Skolemsymbol enthalten.

Skolemisierung – Erfüllbarkeitsäquivalenz

Satz 10.16
Sei F eine Formel und F' durch Skolemisierung aus F hervorgegangen.
Dann sind F und F' erfüllbarkeitsäquivalent.

Beweis
Der Fall der Einführung von Skolemkonstanten wurde in Satz 10.14 behandelt. Für den allgemeinen Fall:
➔ siehe Schöning (s. 67f.)

Definition 10.17
Eine Formel F ist in Skolemform (… ist eine Skolemformel), wenn sie
- in BPF (bereinigter Pränexform) ist
- geschlossen ist (keine freien Variablen aufweist) und
- keinen Existenzquantor enthält.

Eine Formel F ist in Klauselnormalform, wenn sie
- in Skolemform ist und
- ihre Matrix in KNF ist.

➔ In Skolemformen und Klauselnormalformen sind alle Variablen allquantifiziert.
➔ Die Matrix enthält alle relevante Information.

Skolemform

Satz 10.18 (Existenz einer erfüllbarkeitsäquivalenten Skolemformel)
Für jede Formel in BPF existiert eine erfüllbarkeitsäquivalente Skolemformel.
Beweisidee
Nach Satz 10.13 können die freien Variablen durch Existenzquantoren gebunden werden und auf die entstehende Formel kann mehrfach Skolemisierung angewendet werden, bis kein Existenzquantor mehr auftritt. Alle Schritte gewährleisten Erfüllbarkeitsäquivalenz (und Erfüllbarkeitsäquivalenz ist transitiv.)

Verfahren zur Erstellung der Skolemform
Sei F in BPF.

$F_0 := F$
i := 0
while F_i enthält einen Existenzquantor **do**
begin
Es ist $F_i = \forall y_1 \forall y_2 \ldots \forall y_k \exists z\, G_i$
Wähle ein neues k-stelliges Funktionssymbol und setze:
$F_{i+1} = \forall y_1 \forall y_2 \ldots \forall y_k\, G_i[z/f(y1, \ldots, yk)]$
i := i + 1
end

Klauselnormalform für die Prädikatenlogik

Satz 10.18 (Ex. einer erfüllbarkeitsäquivalenten Klauselnormalform)
Für jede Formel existiert eine erfüllbarkeitsäquivalente Formel in Klauselnormalform.
Beweisidee

	Umformungsschritte		
F	beliebige Form		
	Bereinigung: Umbenennen der gebundenen Variablen		
F_1	bereinigt	$F \equiv F_1$	Satz 10.9
	Bindung aller freien Variablen durch Existenzquantoren		
F_2	bereinigt, geschlossen	F_1 erfüllbarkeitsäquivalent F_2	Satz 10.13
	Erstellung einer Pränexform		
F_3	BPF, geschlossen	$F_2 \equiv F_3$	Satz 10.10
	Skolemisierung		
F_4	Skolemform	F_3 erfüllbarkeitsäquivalent F_4	Satz 10.18
	Umformung der Matrix in KNF		
F_5	Klauselnormalform	$F_4 \equiv F_5$	Satz 4.23

Also sind auch F und F_5 sind erfüllbarkeitsäquivalent.

NEU

Ziel: Mengendarstellung der Klauselnormalform (1)

Klauselnormalform
$F = \forall y_1 \forall y_2 \ldots \forall y_p \, (\bigwedge_{i=1}^{n} (\bigvee_{k=1}^{mi} L_{i,k}))$

- Keine Existenzquantoren, keine freien Variablen.
- kein Allquantor ist einem Junktor untergeordnet.

Distributivität
- $(\forall x \, G \wedge \forall x \, H) \equiv \forall x \, (G \wedge H)$

Also
$F \equiv (\bigwedge_{i=1}^{n} \forall y_1 \forall y_2 \ldots \forall y_p \, (\bigvee_{k=1}^{mi} L_{i,k}))$

- In dieser Darstellung ist keine Konjunktion einen Allquantor untergeordnet und kein Allquantor einer Disjunktion.
- Durch Mengenbildung können wir die Konjunktion implizit darstellen:

$\{ \forall y_1 \forall y_2 \ldots \forall y_p \, (\bigvee_{k=1}^{mi} L_{i,k}) \mid i \in \{1, \ldots, n\} \}$

NEU

Ziel: Mengendarstellung der Klauselnormalform (2)

Klauselnormalform: $F = \forall y_1 \forall y_2 \ldots \forall y_p \, (\bigwedge_{i=1}^{n} (\bigvee_{k=1}^{mi} L_{i,k}))$

F **ist repräsentierbar durch:** $\{ \forall y_1 \forall y_2 \ldots \forall y_p \, (\bigvee_{k=1}^{mi} L_{i,k}) \mid i \in \{1, \ldots, n\} \}$

- Keine Existenzquantoren, keine freien Variablen.
- Alle Variablen sind allquantifiziert.
- Alle Quantoren haben Klausel (und ggf. All-Quantoren-Präfix) als Skopus.

Reihenfolge der Allquantoren ist egal, leere Quantifikation ist unnötig

- $\forall x \, \forall y \, G \equiv \forall y \, \forall x \, G$
- Falls x in G nicht frei vorkommt gilt: $G \equiv \forall x \, G$

➔ In der Mengendarstellung kann man die Quantoren weglassen, da man alles weiß, was man zu ihrer Rekonstruktion benötigen könnte.

➔ Namensübereinstimmung von Variablen in verschiedenen Klausel ist irrelevant.

F **ist repräsentierbar durch:** $\{ (\bigvee_{k=1}^{mi} L_{i,k}) \mid i \in \{1, \ldots, n\} \}$

Mengendarstellung von Klauselnormalformen

Definition 11.2

Auch in der Prädikatenlogik werden

- atomare Formeln und ihre Negationen als Literale
- und Disjunktionen von Literalen als Klauseln bezeichnet.

Ist K eine (prädikatenlogische) Klausel, mit

$K = (\bigvee_{k=1}^{m} L_k)$, dann nennen wir

$K = \{L_1, \ldots, L_m\}$ die *Mengendarstellung* von K.

Ist F eine Klauselnormalform mit der Matrix F^*, wobei

$F^* = (\bigwedge_{i=1}^{n} (\bigvee_{k=1}^{mi} L_{i,k}))$, dann nennen wir

$F = \{\{L_{1,1}, \ldots, L_{1,m1}\}, \ldots, \{L_{n,1}, \ldots, L_{n,mn}\}\}$ die *Mengendarstellung* von F.

➔ In der Mengendarstellung einer Klauselnormalform sind die Quantoren nicht mehr explizit.

Mengendarstellung von Klauselnormalformen (2)

Da in Klauselnormalformen

- keine freien Variablen und
- keine Existenzquantoren vorkommen,
- alle Allquantoren im Präfix stehen und
- die Reihenfolge von Allquantoren im Präfix semantisch unwesentlich ist,

werden alle Variablen in der Mengendarstellung behandelt, als wenn sie durch Allquantoren mit maximalem Skopus gebunden sind.

➔ das ist gleichwertig mit: Der Skopus der impliziten Allquantoren ist jeweils die Klausel

Wahrheitswertberechnung

für die Mengendarstellung von Klauselnormalformen

Klauseln

$\mathcal{A}(K) =$ Maximum($\{\mathcal{A}(L) \mid L \in K\}$)

Klauselnormalformen

$\mathcal{A}(F) =$ Minimum($\{\mathcal{A}'(K) \mid K \in F$ und $\mathcal{A}'$ ist eine Struktur, die sich von $\mathcal{A}$ nur durch die Interpretation der Variablen unterscheidet$\}$)

Wichtige Konzepte dieser Vorlesung

- Äquivalenz in der Prädikatenlogik, wichtige Äquivalenzen (Dualität der Quantoren, Skopuserweiterung, Distributivität)
- Äquivalenzumformungen
- bereinigte Formeln, Pränixform, (Quantoren-)Präfix, Matrix
- Variablen-Substitution, Gebundene Umbenennung
- Erfüllbarkeitsäquivalenz (Unterschied zur Äquivalenz)
- Skolemfunktion, Skolemkonstante, Skolemisierung (→ Erfüllbarkeitsäquivalalenz)
- Skolemform, Klauselnormalform, Mengendarstellung der Klauselnormalform

NEU

Beispiel: einfache Variablensubstitution

$F = \forall y\,(\exists z\, P(x, y, z) \vee \forall x\, P(x, y, z))$

$t = g(a)$

$\sigma_1 = [x / g(a)]$

$$\begin{aligned}
\sigma_1(F) &= \sigma_1(\forall y\,(\exists z\, P(x, y, z) \vee \forall x\, P(x, y, z)))\\
&= \forall y\, \sigma_1((\exists z\, P(x, y, z) \vee \forall x\, P(x, y, z)))\\
&= \forall y\,(\sigma_1(\exists z\, P(x, y, z)) \vee \sigma_1(\forall x\, P(x, y, z)))\\
&= \forall y(\exists z\, \sigma_1(P(x, y, z)) \vee \forall x\, P(x, y, z))\\
&= \forall y(\exists z\, P(\sigma_1(x), \sigma_1(y), \sigma_1(z)) \vee \forall x\, P(x, y, z))\\
&= \forall y\,(\exists z\, P(g(a), y, z) \vee \forall x\, P(x, y, z))
\end{aligned}$$

$\sigma_2 = [y / g(a)]$

$\sigma_2(F) = \forall y\,(\exists z\, P(x, y, z) \vee \forall x\, P(x, y, z))$

$\sigma_3 = [z / g(b)]$

$\sigma_3(F) = \forall y\,(\exists z\, P(x, y, z) \vee \forall x\, P(x, y, g(b)))$

$\sigma_1(\sigma_3(F)) = \forall y\,(\exists z\, P(g(a), y, z) \vee \forall x\, P(x, y, g(b)))$

Einfügen hinter 10-11

Beispiel: Substitution

$F = \forall y\,(\exists z\, P(x, y, z) \vee \forall x\, P(x, y, z))$

$\sigma_4 = [x / g(a), z / g(a)]$: die beiden Variablen werden parallel substituiert !

$\sigma_{4y} = \sigma_4$

$\sigma_{4x} = [z / g(a)]$

$\sigma_{4z} = [x / g(a)]$

$$\begin{aligned}
\sigma_4(F) &= \sigma_4(\forall y\,(\exists z\, P(x, y, z) \vee \forall x\, P(x, y, z)))\\
&= \forall y\, \sigma_{4y}((\exists z\, P(x, y, z) \vee \forall x\, P(x, y, z)))\\
&= \forall y\, \sigma_4((\exists z\, P(x, y, z) \vee \forall x\, P(x, y, z)))\\
&= \forall y\,(\sigma_4(\exists z\, P(x, y, z)) \vee \sigma_4(\forall x\, P(x, y, z)))\\
&= \forall y(\exists z\, \sigma_{4z}(P(x, y, z)) \vee \forall x\, \sigma_{4x}(P(x, y, z)\,))\\
&= \forall y(\exists z\, P(\sigma_{4z}(x), \sigma_{4z}(y), \sigma_{4z}(z)) \vee \forall x\, P(\sigma_{4x}(x), \sigma_{4x}(y), \sigma_{4x}(z)))\\
&= \forall y\,(\exists z\, P(g(a), y, z) \vee \forall x\, P(x, y, g(a)))
\end{aligned}$$