Theoretische Informatik und Logik

Prüfungsvorbereitung (Tutorium) – Logik-Teil

Chris(tian) Fermüller
chrisf@logic.at

Februar 2021

Formalisieren in PL (Aufgabe 5 — 7 Punkte)

Formalisieren Sie folgende Aussagen als prädikatenlogische Formeln. Wählen Sie dabei zunächst eine geeignete Signatur (gemeinsam für beide Sätze) und geben Sie die Kategorie und die intendierte Bedeutung aller Symbole vollständig an.

(1) Es gibt Architektinnen, die höchstens ein Bauvorhaben planen.
(*Some architects plan at most one building project.*)

(2) Peters Mutter ist eine Architektin, die mindestens zwei Bauvorhaben plant.
(*Peter's mother is an architect who plans at least two building projects.*)

(1) Es gibt Architektinnen, die höchstens ein Bauvorhaben planen.
(*Some architects plan at most one building project.*)

(2) Peters Mutter ist eine Architektin, die mindestens zwei Bauvorhaben plant.
(*Peter's mother is an architect who plans at least two building projects.*)

Signatur $\langle \{A, B, P\}, \{p\}, \{m\} \rangle$ mit folgender intendierter Bedeutung:

Prädikatensymbole (PS):

$A(x)$	...	x ist eine Architektin (einstellig)
$B(x)$	...	x ist ein Bauvorhaben (einstellig)
$P(x, y)$	...	x plant y (zweistellig)

Konstantensymbol (KS):

p	...	Peter

Funktionssymbol (FS):

$m(x)$	...	die Mutter von x (einstellig)

Formeln:

(1): $\exists x \{A(x) \wedge \exists y \forall z [(B(z) \wedge P(x, z)) \supset y = z)]\}$

(2): $A(m(p)) \wedge \exists x \exists y [x \neq y \wedge B(x) \wedge B(y) \wedge P(m(p), x) \wedge P(m(p), y)]$

Modell und Gegenbeispiel (Aufgabe 6 — 7 Punkte)

Geben Sie ein Modell und ein Gegenbeispiel zu folgender Formel an:
$\forall x[(P(x,x) \wedge \neg P(y, h(a,x))) \supset \neg\exists y P(z, h(y,x))]$

Beachten Sie dabei die in der Vorlesung eingeführten Schreibkonventionen. Spezifieren Sie beide Interpretationen vollständig und begründen Sie die Richtigkeit Ihrer Lösung informell.

Geben Sie auch an welche Variablen frei und welche gebunden vorkommen.

Beachte:

Eine vollständige Lösung besteht aus 6 Bestandteilen:

1. freie Variablenvorkommen
2. gebundene Variablenvorkommen
3. formale Spezifikation eines Modells $\mathcal{I} = \langle D, \Phi, \xi \rangle$
4. Begründung, dass $\mathcal{I}$ die Formel tatsächlich wahr macht
5. formale Spezifikation eines Gegenbeispiels $\mathcal{J} = \langle D', \Phi', \xi' \rangle$
6. Begründung, dass $\mathcal{J}$ die Formel tatsächlich falsch macht

$F = \forall x[(P(x,x) \wedge \neg P(y,h(a,x))) \supset \neg\exists y P(z,h(y,x))]$

1. freie Variablenvorkommen: y, z ($FV(F) = \{y, z\}$)
2. gebundene Variablenvorkommen: y, x
3. Spezifikation des Modells $\mathcal{I} = \langle D, \Phi, \xi \rangle$:
   Es genügt, für alle $c, d \in D$, $\Phi(P)(c,d) = \mathbf{f}$ zu setzen.
   (Alle anderen Komponenten sind beliebig.)
4. Begründung, dass $\mathcal{I}$ die Formel tatsächlich wahr macht:

   Es gilt $\mathrm{val}_{\mathcal{I}}(P(z,h(y,x))) = \mathbf{f}$ für beliebige Variablenbelegungen,
   daher auch $\mathrm{val}_{\mathcal{I}}(\exists y P(z,h(y,x))) = \mathbf{f}$
   und somit $\mathrm{val}_{\mathcal{I}}(\neg\exists y P(z,h(y,x))) = \mathbf{t}$.

   Ganz allgemein gilt: Wenn die rechte Seite einer Implikation wahr ist, dann ist die gesamte Formel wahr.

   Da $(P(x,x) \wedge \neg P(y,h(a,x))) \supset \neg\exists y P(z,h(y,x)))$ für beliebige Variablenbelegungen wahr ist, gilt schließlich $\mathrm{val}_{\mathcal{I}}(F) = \mathbf{t}$.

   Es gibt natürlich viele alternative Modelle bzw. auch Begründungen

$F = \forall x[(P(x,x) \wedge \neg P(y, h(a,x))) \supset \neg \exists y P(z, h(y,x))]$

5. formale Spezifikation eines Gegenbeispiels $\mathcal{J} = \langle D', \Phi', \xi' \rangle$:

   $D' = \omega$ — zur Erinnerung: ω bezeichnet die Menge (im Unterschied zur Struktur) der natürlichen Zahlen.
   $\Phi'(P)(m,n) = \mathbf{t} \Leftrightarrow m \geq n$,
   $\Phi'(h)(m,n) = m + n$,
   $\Phi'(a) = 1$,
   $\xi'(y) = \xi'(z) = 0$, beliebig sonst.

6. Begründung, dass $\mathcal{J}$ die Formel tatsächlich falsch macht:

   Die Formel wird als eine ($\forall$-quantifizierte) Wenn-Dann-Aussage über alle natürlichen Zahlen n interpretiert.
   Der Wenn-Teil besagt: $n \geq n$ und $0 < 1 + n$, was immer wahr ist.
   Das Dann-Teil besagt: Es gibt keine natürliche Zahl m, sodass $0 \geq m + n$.
   Diese Aussage ist falsch, falls $n = 0$.
   Ganz allgemein gilt: Wenn die linke Seite einer Implikation wahr ist, aber die rechte Seite falsch ist, dann ist die gesamte Formel falsch.
   Somit ist die gesamte Formel unter dieser Interpretation falsch.

Tableau-Beweis (Aufgabe 7 — 8 Punkte)

(als letztes in diesem Tutorium, da mit Vorbereitungen verbunden)

Programmkorrektheit [und PL-Eigenschaften] (Aufgabe 8 — 8 Punkte)

Beurteilen Sie die Richtigkeit folgender Aussagen und begründen Sie Ihre Antworten.

(Punkte gibt es nur für hinreichend begründete und korrekte Antworten.)

Hinweis: Sie müssen nicht auf den Hoare-Kalkül verweisen, aber in jedem Fall möglichst genau und vollständig für die Richtigkeit Ihrer Antwort argumentieren.

(Es sind immer zwei Aussagen zu analysieren)

Programmkorrektheit (Forts.)

(a) $\{ x > 3 \}$ `while` $y \leq 2x$ `do begin` $y \leftarrow y + 2x$ `end` $\{ y > 7 \}$ ist bezüglich der angegebenen Spezifikation über dem Datentyp $\mathbb{Z}$ partiell und total korrekt.

Begründung: ☒ richtig ☐ falsch

Lösung:

In jeder Umgebung I, die die Vorbedingung erfüllt, gilt $I(x) > 3$.
Da x in der Schleife nicht verändert wird, garantiert die Schleifenbedingung, dass für die Umgebung I' nach Termination $I'(y) > 2 \cdot I(x) > 7$ gilt.
Somit ist das Programm partiell korrekt.

Das Programm terminiert in jeder Umgebung I, die die Vorbedingung erfüllt, da der Wert von y bei jedem Schleifendurchlauf mindestens um 8 erhöht wird und somit $y \leq 2x$ irgendwann falsch werden muss.
Das Programm ist also auch total korrekt.

Programmkorrekheit (Forts.)

(b) Folgende Aussage gilt für alle P, Q und α bezüglich partieller, aber nicht bezüglich totaler Korrektheit: $\{\, P \,\}$ `while` Q `do` α $\{\, Q \supset \neg P \,\}$.
Begründung: ☒ richtig ☐ falsch

Lösung:
Das Programm terminiert im Allgemeinen nicht.
Daher ist die Aussage bezüglich totaler Korrektheit falsch.

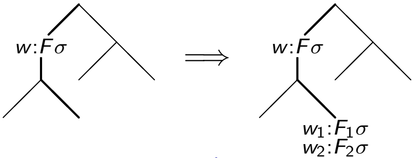
Allerdings garantiert die Schleifenbedingung Q, dass bei Verlassen der Schleife Q falsch ist und somit $Q \supset \neg P$ wahr ist. Daher stimmt die Aussage bezüglich partieller Korrektheit.

Tableau-Expansion

Tableau-Regeln legen fest, wie ein Baum expandiert werden kann.

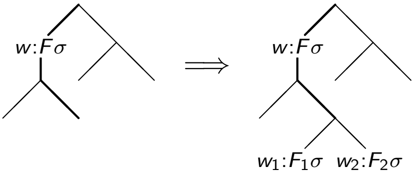
α-Regeln (Konjunktion auf der Meta-Ebene):

$$\frac{w : F}{\begin{array}{c} w_1 : F_1 \\ w_2 : F_2 \end{array}}$$



β-Regeln (Disjunktion auf der Meta-Ebene):

$$\frac{w : F}{w_1 : F_1 \;\Big|\; w_2 : F_2}$$



w, w_1, w_2	...	Wahrheitswerte
F, F_1, F_2	...	Formeln
$F\sigma$	...	F nach Anwendung der Substitution σ
$w : F$	...	bewertete Formeln (*signed formulas*)

Regeln des aussagenlogischen Tableau-Kalküls

$$\frac{\mathbf{f} : A \vee B}{\begin{array}{c}\mathbf{f} : A \\ \mathbf{f} : B\end{array}} \qquad \frac{\mathbf{t} : A \vee B}{\mathbf{t} : A \;\mid\; \mathbf{t} : B} \qquad \frac{\mathbf{f} : A \wedge B}{\mathbf{f} : A \;\mid\; \mathbf{f} : B} \qquad \frac{\mathbf{t} : A \wedge B}{\begin{array}{c}\mathbf{t} : A \\ \mathbf{t} : B\end{array}}$$

$$\frac{\mathbf{f} : A \supset B}{\begin{array}{c}\mathbf{t} : A \\ \mathbf{f} : B\end{array}} \qquad \frac{\mathbf{t} : A \supset B}{\mathbf{f} : A \;\mid\; \mathbf{t} : B} \qquad \frac{\mathbf{f} : \neg A}{\mathbf{t} : A} \qquad \frac{\mathbf{t} : \neg A}{\mathbf{f} : A}$$

Beachte:

- Je zwei Regeln: eine für $\mathbf{f} : A * B$, eine für $\mathbf{t} : A * B$ ($\mathbf{f} : \neg A$ / $\mathbf{t} : \neg A$).
- Regeln aus den Wahrheitsfunktionen (*or*, *and*, *implies*, *not*) ablesbar.
- Beweise sind auf den Kopf gestellte Bäume (*downward trees*).
- Jede dieser Expansionsregeln muss auf jedes Vorkommen einer bewerteten Formel nur einmal angewendet werden.
  $\Longrightarrow$ Jeder Beweisversuch terminiert.

Quantoren-Regeln des Tableau-Kalküls

Σ^{par} ... Signatur Σ erweitert um Parameter $c, d, e, \ldots$
$\mathbf{t} : \forall x F$ und $\mathbf{f} : \exists x F$ heißen γ-Formeln
$\mathbf{f} : \forall x F$ und $\mathbf{t} : \exists x F$ heißen δ-Formeln

γ-Regeln: $\qquad \dfrac{\mathbf{t} : \forall x F}{\mathbf{t} : F(x/t)} \qquad \dfrac{\mathbf{f} : \exists x F}{\mathbf{f} : F(x/t)}$

für beliebige geschlossene (=variablenfreie) Terme t über Σ^{par}

δ-Regeln: $\qquad \dfrac{\mathbf{f} : \forall x F}{\mathbf{f} : F(x/c)} \qquad \dfrac{\mathbf{t} : \exists x F}{\mathbf{t} : F(x/c)}$

für einen neuen Parameter c in Σ^{par}

Die α- und β-Regeln (AL) bleiben unverändert!

Achtung: γ-Regeln müssen i.A. auf die selbe γ-Formel öfters, mit verschiedenen t angewendet werden. (Im Gegensatz zur δ-Regel.)
Überlege: Warum ist das so?

Tableau-Kalkül – Teil 3: Gleichheitsregeln für PL-Tableaux

Erinnerung: Wir wenden die Regeln nur auf geschlossene Formeln an.
Daher gibt es in einem PL-Tableau niemals freie Variablen!

Daher können atomare Formeln nur geschlossene (=variablenfreie) Terme enthalten. Für vorkommende "$\mathbf{t}[\mathbf{f}] : s = t$" gilt also $V(s) = V(t) = \{\}$!

Abschlussregel für Gleichheitsatome ($AB^=$):

Enthält ein Ast eine negative Formel $\mathbf{f} : t = t$ so wird er geschlossen.

Notation:

$A[s]$ . . . A ist ein Atom, s ein bestimmtes Term-Vorkommen darin

$A[s/t]$ . . . in $A[s]$ ausgewiesenes Vorkommen von s wurde durch t ersetzt

Substitutionsregeln für Gleichheitsatome ($S^=$):

$$\frac{\begin{array}{c}\mathbf{t} : s = t\\ \mathbf{t} : A[s]\end{array}}{\mathbf{t} : A[s/t]} \qquad \frac{\begin{array}{c}\mathbf{t} : s = t\\ \mathbf{f} : A[s]\end{array}}{\mathbf{f} : A[s/t]} \qquad \frac{\begin{array}{c}\mathbf{t} : s = t\\ \mathbf{t} : A[t]\end{array}}{\mathbf{t} : A[t/s]} \qquad \frac{\begin{array}{c}\mathbf{t} : s = t\\ \mathbf{f} : A[t]\end{array}}{\mathbf{f} : A[t/s]}$$

Achtung: A muss atomar sein, kann aber auch ein Gleichheitsatom sein.
s bzw. t kommen beliebig tief vor (nicht nur als Argument eines Prädikats).

PL-Tableaux mit Gleichheit – Beispiele (1)

Reflexivität, Symmetrie und Transitivität der Gleichheit

$$\begin{array}{lll}
(1) & \mathbf{f} : \forall x\, x = x & \text{Ann. } (\delta) \\
\hline
(2) & \mathbf{f} : a = a & \text{von } 1 \\
 & \times & AB^{=}
\end{array}$$

$$\begin{array}{lll}
(1) & \mathbf{t} : a = b & \text{Ann.} \\
(2) & \mathbf{f} : b = a & \text{Ann.} \\
\hline
(3) & \mathbf{f} : a = a & S^{=}\text{: } 1{\to}2 \\
 & \times & AB^{=}
\end{array}$$

$$\begin{array}{lll}
(1) & \mathbf{t} : a = b & \text{Ann.} \\
(2) & \mathbf{t} : b = c & \text{Ann.} \\
(3) & \mathbf{f} : a = c & \text{Ann.} \\
\hline
(4) & \mathbf{f} : b = c & S^{=}\text{: } 1{\to}3 \\
 & \times & \text{Wid. } 2/4
\end{array}$$

Beachte: In den Tableaux für Symmetrie und Transitivität könnten statt den Konstanten a, b, c beliebige Terme stehen.
Alternativ könnte man mit $\mathbf{f} : \forall x \forall y (x = y \supset y = x)$ bzw. mit
$\mathbf{f} : \forall x \forall y \forall z [(x = y \land y = z) \supset x = z]$ beginnen.

Tableau-Beweis (Aufgabe 7 — 8 Punkte)

Zeigen Sie mit dem Tableau-Kalkül:
Aus $\forall y[\exists z Q(y,z) \supset P(f(y))]$, $\forall y\, y = f(f(y))$ und $\forall y Q(y,y)$ folgt $P(b)$.
Kennzeichnen Sie γ- und δ-Formeln als solche und nummerieren Sie alle auftretenden Formeln.

Folgendes geschlossene Tableau zeigt, dass die Konsequenzbehauptung richtig ist:

(1)	$\mathbf{t} : \forall y[\exists z Q(y,z) \supset P(f(y))]$				Ann., γ-Formel
(2)	$\mathbf{t} : \forall y\, y = f(f(y))$				Ann., γ-Formel
(3)	$\mathbf{t} : \forall y Q(y,y)$				Ann., γ-Formel
(4)	$\mathbf{f} : P(b)$				Ann.
(5)	$\mathbf{t} : \exists z Q(f(b),z) \supset P(f(f(b)))$				von 1 [nicht δ/γ!]
(6)	$\mathbf{f} : \exists z Q(f(b),z)$	von 5 γ-Formel	(7)	$\mathbf{t} : P(f(f(b)))$	von 5
(8)	$\mathbf{f} : Q(f(b),f(b))$	von 6	(9)	$\mathbf{t} : b = f(f(b))$	von 2
(10)	$\mathbf{t} : Q(f(b),f(b))$	von 3	(11)	$\mathbf{t} : P(b)$	$S^{=}$:9$\to$7
	$\times$	Wid.: 8/10		$\times$	Wid.: 4/11

Häufiger Fehler:
γ/δ markiert nicht entsprechende Formeln, sondern Regel-Ergebnisse