Formale Modellierung
Vorlesung 8 vom 08.06.15: Logik Höherer Stufe

Christoph Lüth

Universität Bremen

Sommersemester 2015

Fahrplan

- Alles über Logik höherer Stufe (Higher-order Logic, HOL):
  - Typen und Terme
  - Die Basis-Axiome
  - Definierte Operatoren

Fahrplan

- Teil I: Formale Logik
  - Einführung
  - Aussagenlogik (PL): Syntax und Semantik, Natürliches Schließen
  - Konsistenz & Vollständigkeit der Aussagenlogik
  - Prädikatenlogik (FOL): Syntax und Semantik
  - Konsistenz & Vollständigkeit von FOL
  - FOL mit induktiven Datentypen
  - FOL mit rekursiven Definitionen
  - Logik höherer Stufe (HOL): Syntax und Eigenschaften
  - Berechungsmodelle (Models of Computation)
  - Die Unvollständigkeitssätze von Gödel
- Teil II: Spezifikation und Verifikation

Logik höherer Stufe

- Ziel: Formalisierung von Mathematik
  - "Logik für Erwachsene"
- Problem: Mögliche Inkonsistenz (Russel's Paradox)
- Lösung: Restriktion vs. Ausdrucksstärke
- Alternative Grundlagen:
  - Andere Typtheorien (Martin-Löf, Calculus of Constructions)
  - Ungetypte Mengenlehre (ZFC)
- HOL: guter Kompromiss, weit verbreitet.
  - Klassische Logik höherer Stufe nach Church
  - Schwächer als ZFC, stärker als Typtheorien

Warum Logik höherer Stufe?

- Aussagenlogik: keine Quantoren
- Logik 1. Stufe: Quantoren über Terme

$$\forall x\, y.\, x = y \longrightarrow y = x$$

- Logik 2. Stufe: Quantoren über Prädikaten und Funktionen

$$\forall P.(P\,0 \wedge \forall x.P\,x \longrightarrow P\,(S\,x)) \longrightarrow \forall x.P\,x$$

- Logik 3. Stufe: Quantoren über Argumenten von Prädikaten
- Logik höherer Stufe (HOL): alle endlichen Quantoren
  - Keine wesentlichen Vorteile von Logik 2. Ordnung

Vermeidung von Inkonsistenzen

- Russell's Paradox
  - $R = \{X \mid X \notin X\}$
  - Abhilfe: Typen
- Gödel's 2. Unvollständigkeitssatz:
  - Jede Logik, die ihre eigene Konsistenz beweist, ist inkonsistent.
- Unterscheidung zwischen Termen und Aussagen
  - Dadurch **in** HOL keine Aussage **über** HOL

Typen

- Typen $\mathcal{T}ype$ gegeben durch
  - Typkonstanten: $c \in \mathcal{C}_{Type}$ (Menge $\mathcal{C}_{Type}$ durch Signatur gegeben)
    - $Prop, Bool \in \mathcal{C}_{Type}$: $Prop$ alle Terme, $Bool$ alle Aussagen
  - Typvariablen: $\alpha \in \mathcal{V}_{Type}$ (Menge $\mathcal{V}_{Type}$ fest)
  - Funktionen: $s, t \in \mathcal{T}ype$ dann $s \Rightarrow t$ in $\mathcal{T}ype$
- Konvention: Funktionsraum nach rechts geklammert
  $\alpha \Rightarrow \beta \Rightarrow \gamma$ für $\alpha \Rightarrow (\beta \Rightarrow \gamma)$

Terme

- Terme $\mathcal{T}erm$ gegeben durch
  - Konstanten: $c \in \mathcal{C}$ (Menge $\mathcal{C}$ durch Signatur gegeben)
  - Variablen: $v \in \mathcal{V}$
  - Applikation: $s, t \in \mathcal{T}erm$ dann $s\,t \in \mathcal{T}erm$
  - Abstraktion: $x \in \mathcal{V}, t \in \mathcal{T}erm$ dann $\lambda x.\, t \in \mathcal{T}erm$
- Konventionen: Applikation links geklammert, mehrfache Abstraktion
  $\lambda x\, y\, z.\, f\, x\, y\, z$ für $\lambda x.\, \lambda y.\, \lambda z.\, ((f\, x)\, y)\, z$

Basis-Syntax

$=\ :: \alpha \Rightarrow \alpha \Rightarrow Bool$
$\longrightarrow\ :: Bool \Rightarrow Bool \Rightarrow Bool$
$\iota\ :: (\alpha \Rightarrow Bool) \Rightarrow \alpha$

$\neg\ :: Bool \Rightarrow Bool$
$\top\ :: Bool$
$\bot\ :: Bool$
$if\ :: Bool \Rightarrow \alpha \Rightarrow \alpha \Rightarrow \alpha$
$\forall\ :: (\alpha \Rightarrow Bool) \Rightarrow Bool$
$\exists\ :: (\alpha \Rightarrow Bool) \Rightarrow Bool$
$\wedge\ :: Bool \Rightarrow Bool \Rightarrow Bool$
$\vee\ :: Bool \Rightarrow Bool \Rightarrow Bool$

- Einbettung (wird weggelassen) $trueprop :: Bool \Rightarrow Prop$
- Basis-Operatoren: $=, \longrightarrow, \iota$
- Syntaktische Konventionen:
  - Bindende Operatoren: $\forall$, $\exists$, ι $\forall x.P \equiv \forall(\lambda x.P)$
  - Infix-Operatoren: $\wedge, \vee, \longrightarrow, =$
  - Mixfix-Operator: $if\ b\ then\ p\ else\ q \equiv if\ b\ p\ q$

Basis-Axiome I: Gleichheit

- Reflexivität:

$$\frac{}{t = t}\ \mathrm{refl}$$

- Substitutivität:

$$\frac{s = t \quad P\ s}{P\ t}\ \mathrm{subst}$$

- Extensionalität:

$$\frac{f\ x = g\ x}{(\lambda x.\ f\ x) = (\lambda x.\ g\ x)}\ \mathrm{ext}\quad (*)$$

$(*)$ Eigenvariablenbedingung: x nicht frei in offenen Vorbedingungen

- Einführungsregel:

$$\frac{}{(P \longrightarrow Q) \longrightarrow (Q \longrightarrow P) \longrightarrow (P = Q)}\ \mathrm{iff}$$

Abgeleitete Operatoren

$$\begin{aligned}
\top &\equiv (\lambda x.\ x) = (\lambda x.\ x)\\
\forall P &\equiv (P = \lambda x.\ \top)\\
\exists P &\equiv \forall Q.(\forall x.P\ x \longrightarrow Q) \longrightarrow Q\\
\bot &\equiv \forall P.\ P\\
\neg P &\equiv P \longrightarrow \bot\\
P \wedge Q &\equiv \forall R.\ (P \longrightarrow Q \longrightarrow R) \longrightarrow R\\
P \vee Q &\equiv \forall R.\ (P \longrightarrow R) \longrightarrow (Q \longrightarrow R) \longrightarrow R\\
if\ P\ then\ x\ else\ y &\equiv \iota z.(P = \top \longrightarrow z = x) \wedge (P = \bot \longrightarrow z = y)
\end{aligned}$$

Basis-Axiome II: Implikation und Auswahl

- Einführungsregel Implikation:

$$\frac{\begin{array}{c}[P]\\ \vdots\\ Q\end{array}}{P \longrightarrow Q}\ \mathrm{impl}$$

- Eliminationsregel Implikation:

$$\frac{P \longrightarrow Q \quad P}{Q}\ \mathrm{mp}$$

- Eliminationsregel Auswahloperator:

$$\frac{}{(\iota x.x = a) = a}\ \mathrm{the\_eq}$$

- HOL ist klassisch:

$$\frac{}{(P = \top) \vee (P = \bot)}\ \mathrm{true\_or\_false}$$

Die Basis-Axiome (Isabelle-Syntax)

$$\begin{aligned}
\mathrm{refl}: &\quad t = t\\
\mathrm{subst}: &\quad [\![s = t; P(s)]\!] \Longrightarrow P(t)\\
\mathrm{ext}: &\quad [\![\textstyle\bigwedge x.fx = gx]\!] \Longrightarrow (\lambda x.fx) = (\lambda x.gx)\\
\mathrm{impl}: &\quad [\![P \Longrightarrow Q]\!] \Longrightarrow P \longrightarrow Q\\
\mathrm{mp}: &\quad [\![P \longrightarrow Q; P]\!] \Longrightarrow Q\\
\mathrm{iff}: &\quad (P \longrightarrow Q) \longrightarrow (Q \longrightarrow P) \longrightarrow (P = Q)\\
\mathrm{the_eq}: &\quad (\iota x.x = a) = a\\
\mathrm{true_or_false}: &\quad (P = \top) \vee (P = \bot)
\end{aligned}$$

Eigenschaften der Logik höherer Stufe

- Konsistent (soweit wir wissen)
- Unvollständig
  - ... und damit unentscheidbar
  - Beweis: folgt aus den Gödelschen Unvollständigkeitssätzen

Erweiterungen

- Weitere Operatoren
- Weitere Typen: natürliche Zahlen, Datentypen
- Alle Erweiterungen sind konservativ und damit konsistenzbewahrend

Zusammenfassung

Logik höherer Stufe (HOL):

- Syntax basiert auf dem einfach getypten λ-Kalkül
- Drei Basis-Operatoren, acht Basis-Axiome
- Rest folgt durch konservative Erweiterung — Donnerstag