

Formale Modellierung

Vorlesung 13 vom 13.07.2015: Zusammenfassung, Rückblick,
Ausblick

Christoph Lüth

Universität Bremen

Sommersemester 2015

Fahrplan

- ▶ Teil I: Formale Logik
- ▶ Teil II: Spezifikation und Verifikation
 - ▶ Formale Modellierung von Software
 - ▶ Temporale Logik und Modellprüfung
 - ▶ Zusammenfassung, Rückblick, Ausblick

Heute in diesem Theater

- ▶ Zusammenfassung und Rückblick
- ▶ Formale Modellierung und Formale Methoden in der Praxis
- ▶ ... und jetzt?

Unsere Reise durch die Logik

	Entscheidbar?	Vollständig?	Konsistent?	Werkzeuge (Beweiser)
Aussagenlogik	J	J	J	SAT-Solver
Presburger	J	J	J	SMT-Beweiser: Z3, CVC
Peano-Ar.	N	J	J	
FOL	N	J	J	ATPs: SPASS, Vampire
FOL + Induktion	N	N	J	KIV, KeY, Inka
HOL	N	N	J	ITPs: Isabelle, Coq, PVS

Aussagenlogik

- ▶ Formeln und Bedeutung
- ▶ Beweisprinzipien:
 - ▶ Wahrheitstabelle, natürliches Schließen, Äquivalenzumformung, Resolution
- ▶ $\models P$ vs. $\vdash P$
- ▶ Warum ist Aussagenlogik entscheidbar?

Prädikatenlogik

- ▶ Formeln und Bedeutung
- ▶ Welche Beweisprinzipien?
- ▶ Besonderheit beim natürlichen Schließen?
- ▶ Warum ist Prädikatenlogik vollständig?
- ▶ ... und warum nicht mehr entscheidbar?

Induktion und Logik höherer Stufe

- ▶ Wie axiomatisieren wir die natürlichen Zahlen?
- ▶ Wie sehen Modelle der natürlichen Zahlen aus (und was ist ein Nichtstandardmodell)?
- ▶ Was ist der Unterschied zwischen natürlicher Induktion und wohlfundierter Induktion?
- ▶ Wie funktioniert der Beweis für die Unvollständigkeitssätze?
- ▶ Warum ist Logik höherer Stufe nicht mehr vollständig?
- ▶ Was ist eine konservative Erweiterung?

Die Gödelschen Unvollständigkeitssätze

- ▶ Kerntechnik: Gödelkodierung

Die Gödelschen Unvollständigkeitssätze

- ▶ Kerntechnik: Gödelkodierung
- ▶ Kodierung von Termen, Formeln, Ableitung in Peano-Arithmetik (**PA**)
 - ▶ Warum Peano?
- ▶ Beweisidee: Logik, in der **PA** formalisierbar ist, kann potenziell über sich selbst reden.
- ▶ Technisch:
 - ▶ $\text{Thm}(f)$ gdw. $\lfloor f \rfloor$ ist ein Theorem
 - ▶ Fixpunktsatz: zu Formel $\varphi(x)$ gibt es ψ so dass $\vdash \varphi(\lfloor \psi \rfloor) \leftrightarrow \psi$
 - ▶ *Gödel-sentence*: $\varphi(x) \stackrel{\text{def}}{=} \neg \text{Thm}(x)$

UML

- ▶ Was ist formal an der UML?

UML

- ▶ Was ist formal an der UML?
 - ▶ Klassendiagramme, Zustands- und Sequenzdiagramme
- ▶ Was ist OCL?
 - ▶ Eine Sprache zur Einschränkung der Modellklasse
 - ▶ Woraus besteht die OCL?
 - ▶ Welche Logik benutzt die OCL?
 - ▶ Welche Typen kennt die OCL?

Temporallogik

- ▶ Was sind temporale Logiken?
- ▶ Welche Operatoren haben LTL und CTL? Was ist der Unterschied?
- ▶ Wie ist Gültigkeit für LTL/CTL definiert?
- ▶ Ist LTL/CTL entscheidbar? ... vollständig?
- ▶ Was ist das Modelchecking-Problem?
- ▶ Was ist das Problem beim Modelchecking?

Modellierung, formale Modellierung, Programme und formale Methoden

- ▶ Formale Logik — Mathematik
- ▶ Programme und Berechenbarkeit
- ▶ Formale Methoden: Anwendung der Methoden der Logik auf Programme
- ▶ Automatisierte Beweisverfahren: Anwendung von Programmen auf die Logik

Formale Modellierung: Geschichtlicher Rückblick

- ▶ Gottlob Frege (1848– 1942)
 - ▶ 'Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens' (1879)
- ▶ Georg Cantor (1845– 1918), Bertrand Russel (1872– 1970), Ernst Zermelo (1871– 1953)
 - ▶ Einfache Mengenlehre: inkonsistent (Russel's Paradox)
 - ▶ Axiomatische Mengenlehre: Zermelo-Fränkel
- ▶ David Hilbert (1862– 1943)
 - ▶ Hilbert's Programm: 'mechanisierte' Beweistheorie
- ▶ Kurt Gödel (1906– 1978)
 - ▶ Vollständigkeitssatz, Unvollständigkeitssätze

Formale Methoden: Geschichtlicher Rückblick

- ▶ Ziel: Methoden, um die **Korrektheit** von Programmen sicherzustellen
- ▶ Erste Ansätze: Alan Turing (1949)
- ▶ Robert Floyd und CAR Hoare: Floyd-Hoare-Kalkül (1969/1971)
- ▶ Korrektheit durch Konstruktion: Dijkstra, Gries und andere (1972 ff)
- ▶ Problem: **sehr viele**, größtenteils **triviale** Beweise

Automatisches Theorembeweisen

- ▶ Automatisches Beweisen: Wurzeln in der Mathematik, ursprünglich Teil der KI:
 - ▶ Termersetzung (Thue, Semi-Thue-Systeme: 1910; Schönfinkel, Kombinatorlogik: 1930)
 - ▶ SAT (Davis-Putnam, 1960; Davis, Logemann and Loveland, 1962)
 - ▶ Resolution (Robinson, 1965: Unifikation)
- ▶ Früher Enthusiasmus, dann Ernüchterung; durch leistungsfähigere Rechner und Algorithmen späte Blüte.

Formale Modellierung und Formale Methoden

- ▶ Das LCF System (Robin Milner: Stanford LCF, Cambridge LCF, Edinburgh LCF, ab 1972)
 - ▶ Entwickelt als “Programmbeweissystem”
 - ▶ “Stammvater” vieler moderner Beweise: Isabelle, Coq, HOL4, HOL light
- ▶ NQTHM (Boyer-Moore, ab 1971)
 - ▶ Heute: ACL-2
- ▶ Zwei Schulen: getypt vs. ungetypt, expansiv vs. Beweisobjekte

Formale Methoden

- ▶ Stetiger Fortschritt auf vielen Ebenen
- ▶ Statische Programmanalyse (eg. WCET, AbsInt)
- ▶ Modellbasierte Entwicklung (insbes. SCADE und andere)
- ▶ Beweisbasierte Verfahren (Microsoft's SLAM, B-Methode)
- ▶ Hardwareverifikation: Intel, AMD, Infineon, ...
- ▶ L4.verified

Formale Modellierung in der Mathematik

- ▶ Offene mathematische Probleme und ihre Lösung:
 - ▶ Perelmann und Poincaré; Andrew Wiles und Fermat's letztes Theorem; die Riemannsche Vermutung
 - ▶ Beweise werden zunehmend komplexer
- ▶ Rechnergestützte Beweisverfahren:
 - ▶ Vierfarbenproblem (Appel-Haken, 1970)
- ▶ Vollständig formalisierte Beweisverfahren:
 - ▶ Vierfarbenproblem in Coq (Gonthier, 2005)
 - ▶ Die Keplersche Vermutung und Flyspeck (Hales, 2002– 2014)

Stand der Kunst

- ▶ Formale Modellierung Stand der Kunst
 - ▶ Luft- und Raumfahrt
 - ▶ Automotive
 - ▶ ... **nicht** im Finanzbereich!
- ▶ In den kommenden Jahren: weitere Anwendungsgebiete
 - ▶ Spezialisierte Techniken für bestimmte Anwendungsfälle (DSLs)

... und jetzt?

- ▶ Besuchen Sie auch: Formale Methoden der Softwaretechnik (Master-Wahlveranstaltung)
- ▶ Bachelor/Diplomarbeiten am DFKI/AGRA
- ▶ Andere Gruppen an der Uni Bremen

Tschüß!

