Theorem-Proving Fundamentals

Specification and Verification with Higher-Order Logic

Arnd Poetzsch-Heffter
(Slides by Jens Brandt)

Software Technology Group
Fachbereich Informatik
Technische Universität Kaiserslautern

Sommersemester 2008

Outline

Overview

Motivation

- How does a theorem prover work?
- What does a theorem prover?
- What is a proof?

Goals

- recapitulate elementary proof theory
- introduce English terms

Outline

Syntax

Language

- used to designate things and express facts
- terms and formulas are formed from variables and function symbols
- function symbols map a tupel of terms to another term
- constant symbols: no arguments
- constant can be seen as functions with zero arguments
- predicate symbols are considered as boolean functions
- set of variables

Example (Natural Numbers)

- constant symbol: 0
- function symbol suc : $\mathbb{N} \rightarrow \mathbb{N}$

Syntax of Propositional Logic

Example (Symbols)

- $\mathscr{V} = \{a, b, c, \ldots\}$ is a set of propositional variables
- two function symbols: $\neg$ and $\rightarrow$

Example (Language)

- each $P \in \mathscr{V}$ is a formula
- if ϕ is a formula, then $\neg\phi$ is a formula
- if ϕ and ψ are formulas, then $\phi \rightarrow \psi$ is a formula

Syntactic Sugar

Purpose

- additions to the language that do not affect its expressiveness
- more practical way of description

Example

Abbreviations in Propositional Logic

- true denotes $\phi \longrightarrow \phi$
- false denotes $\neg$true
- $\phi \vee \psi$ denotes $(\neg \phi) \longrightarrow \psi$
- $\phi \wedge \psi$ denotes $\neg((\neg \phi) \vee (\neg \psi))$
- $\phi \leftrightarrow \psi$ denotes $((\phi \longrightarrow \psi) \wedge (\psi \longrightarrow \phi))$

Semantics

Purpose

- syntax only specifies the structure of terms and formulas
- symbols and terms are assigned a meaning
- variables are assigned a value
- in particular, propositional variables are assigned a truth value

Bottom-Up Approach

- assignments give variables a value
- terms/formualas are evaluated based on the meaning of the function symbols

Structure

Definition (Structure)

Let $\mathscr{L}$ be a language formed with the function symbols $f_0, f_1, \ldots, f_n$. An untyped structure $\mathscr{M}$ for $\mathscr{L}$ is an $(n+2)$-tuple:

- non-empty set M, called the *universe*
- a function $\hat{f}_0 : M^{\mathrm{arity}(\mathrm{f}_0)} \longrightarrow M$
- ...
- a function $\hat{f}_n : M^{\mathrm{arity}(\mathrm{f}_n)} \longrightarrow M$

Interpretation

Definition (Variable assignment)

- a function $I : \mathscr{V} \rightarrow M$, maps variables to values of M

Definition (Denotation V of a term)

- if $\phi \in \mathscr{V}$: $V(\phi) = I(\phi)$
- if $f_i(\phi_1, \ldots, \phi_n) = \hat{f}_i(V(\phi_1), \ldots V(\phi_n))$

Interpretation

Example (Assignment in Propositional Logic)

- $I : \mathscr{V} \longrightarrow \{\text{true}, \text{false}\}$

Example (Denotation of Propositional Logic)

- if $\phi \in \mathscr{V}$: $V(\phi) = I(\phi)$
- $V(\neg\phi) = f_\neg(V(\phi))$
- $V(\phi \rightarrow \psi) = f_\rightarrow(V(\phi), V(\psi))$

$f_\neg$	
false	true
true	false

$f_\rightarrow$	false	true
false	true	true
true	false	true

Validity

Definition (Validity of formulas)

- a formula ϕ is valid in $\mathscr{M}$ if ϕ evaluates to true for all assigments I
- notation: $\mathscr{M} \models \phi$
- a proposition ϕ is valid if it is valid in $\mathscr{M} = (\{\text{true}, \text{false}\}, f_\neg, f_\rightarrow)$

Example (Propositional Logic Tautology)

- $\phi = a \vee \neg a$ (where $a \in \mathscr{V}$) is valid
  - $I(a) = \text{false}$: $V(a \vee \neg a) = \text{true}$
  - $I(a) = \text{true}$: $V(a \vee \neg a) = \text{true}$

Outline

Introduction

General Concept

- purely syntactical manipulations based on designated transformation rules
- starting point: set of formulas, often a given set of axioms
- deriving new formulas by deduction rules from given formulas Γ
- ϕ is *provable* from Γ if ϕ can be obtained by a finite number of derivation steps assuming the formulas in Γ
- notation: $\Gamma \vdash \phi$ means ϕ is *provable* from Γ
- notation: $\vdash \phi$ means ϕ is *provable* from a given set of axioms

Proof System Styles

Hilbert Style

- easy to understand
- hard to use

Natural Deduction

- easy to use
- hard to understand

- ...

Hilbert-Style Deduction Rules

Definition (Deduction Rule)

- deduction rule d is a $n+1$-tuple

$$\frac{\phi_1 \quad \cdots \quad \phi_n}{\psi}$$

- formulas $\phi_1 \ldots \phi_n$, called premises of rule
- formula ψ, called conclusion of rule

Hilbert-Style Proofs

Definition (Proof)

- let D be a set of deduction rules, including the axioms as rules without premisses
- *proofs* in D are (natural) trees such that
  - axioms are proofs
  - if $P_1, \ldots, P_n$ are proofs with roots $\phi_1 \ldots \phi_n$ and $\dfrac{\phi_1 \cdots \phi_n}{\psi}$ is in D, then $\dfrac{P_1 \cdots P_n}{\psi}$ is a proof in D
- can also be written in a line-oriented style

Hilbert-Style Deduction Rules

Axioms

- let Γ be a set of axioms, $\psi \in \Gamma$, then $\overline{\psi}$ is a proof
- axioms allow to construct trivial proofs

Modus Ponens

- Rule example: $\dfrac{\phi \rightarrow \psi \quad \phi}{\psi}$
- if $\phi \rightarrow \psi$ and ϕ have already been proven, ψ can be deduced

Proof Example

Example (Hilbert Proof)

- language formed with the four proposition symbols P, Q, R, S
- axioms: P, Q, $Q \longrightarrow R$, $P \longrightarrow (R \longrightarrow S)$

$$\dfrac{\dfrac{\dfrac{}{P \longrightarrow (R \longrightarrow S)} \quad \dfrac{}{P}}{R \longrightarrow S} \qquad \dfrac{\dfrac{}{Q \longrightarrow R} \quad \dfrac{}{Q}}{R}}{S}$$

Hilbert Calculus for Propositional Logic

Example (Axioms of Propositional Logic)

All instantiations of the following schemas:

- true
- $a \longrightarrow (b \longrightarrow a)$
- $(a \longrightarrow (b \longrightarrow c)) \longrightarrow ((a \longrightarrow b) \longrightarrow (a \longrightarrow c))$
- $(\neg b \longrightarrow \neg a) \longrightarrow ((\neg b \longrightarrow a) \longrightarrow b)$
- where a, b, c are arbitrary propositions

Natural Deduction

Motivation

- introducing a hypothesis is a natural step in a proof
- Hilbert proofs do not permit this directly
- can be only encoded by using $\longrightarrow$
- proofs are much longer and not very natural

Natural Deduction

- alternative definition where introduction of a hypothesis is a deduction rule
- deduction step can modify not only the proven propositions but also the theory Γ

Natural Deduction Rules

Definition (Deduction Rule)

- deduction rule d is a $n+1$-tuple

$$\frac{\Gamma_1 \vdash \phi_1 \quad \cdots \quad \Gamma_n \vdash \phi_n}{\Gamma \vdash \psi}$$

- pairs of Γ (set of formulas) and ϕ (formulas): sequents
- proof: tree of sequents

Natural Deduction Rules

Definition (Deduction Rule)

- rich set of rules
- elimination rules eliminate a logical symbol from a premise
- introduction rules introduce a logical symbol into the conclusion
- reasoning from assumptions formalised as the elimination rule for the implication $\longrightarrow$

Natural Deduction Rules

Example (Natural Deduction Rules)

- $\vee$-introduction

$$\frac{\Gamma \vdash \phi}{\Gamma \vdash \phi \vee \psi} \qquad \frac{\Gamma \vdash \psi}{\Gamma \vdash \phi \vee \psi}$$

- $\vee$-elimination

$$\frac{\Gamma \vdash \phi \vee \psi \quad \Gamma, \phi \vdash \xi \quad \Gamma, \psi \vdash \xi}{\Gamma \vdash \xi}$$

- $\longrightarrow$-introduction

$$\frac{\Gamma, \phi \vdash \psi}{\Gamma \vdash \phi \longrightarrow \psi}$$

- $\longrightarrow$-elimination

$$\frac{\Gamma \vdash \phi \longrightarrow \psi \quad \Gamma \vdash \phi}{\Gamma \vdash \psi}$$

Outline

Summary

Theorem-Proving Fundamentals

- syntax: symbols, language
- semantics: structure, assigment, denotation
- proof system: theory, axioms, deduction rules

Outlook

- theorem-prover principles and architecture