A Review of Theorem Provers

Aleksey Nogin

February 11 & 18, 2002

Overview

- Higher–order interactive provers:
  - Constructive: ALF, Alfa, Coq, [MetaPRL, NuPRL]
  - Classical: HOL, PVS
- Logical Frameworks: Isabelle, LF, Twelf, [MetaPRL]
- Inductive provers: ACL2, Inka
- Automated:
  - Multi-logic: Gandalf, TPS
  - first–order classical logic: Otter, Setheo, SPASS
  - Equational reasoning: EQP, Maude
- Other: Omega, Mizar

ALF, Alfa

- ALF ("Another logical framework") is a structure editor for monomorphic Martin-Löf type theory.
- Proofs are done by refinement of an incomplete proof object.
- Alfa is ALF with display forms and hypertext navigation.
- Primarily used to formalize parts of intuitionistic mathematics.

Coq

- Coq is a Proof Assistant for a Calculus of Inductive Constructions Logical Framework.
- Decidable typechecker.
- Focus on extracting programs from proofs.
- Has a compilation mode and an interactive ("debug") mode.
- Written in OCaml, open source (GPL)

HOL

- HOL uses classical predicate calculus with terms from the typed λ–calculus — Church's higher–order logic.
- Its meta-language is ML.
- Used for hardware and software verification.
- Has derived rules that act like tactics (e.g have to be replayed).
- HOL98: Written in Moscow ML, open source (BSD)

PVS

- PVS (Prototype Verification System) is a specification language with support tools and theorem prover.
- Language of PVS is based on classical, strongly typed higher-order logic
- Provides some integration with model-checking.
- Applied to software and hardware specification and verification.
- License: free for non-commercial use, but without a right to modify.

Isabelle

- Isabelle is a Logical Framework.
- It implements first-order logic (constructive and classical)
- It implements a version of HOL's logic
- It implements ZF set theory
- It implements an version of Martin-Löf's Type Theory with universal equality.
- ...
- Has a higher–order unification algorithm at its "core"
- Used: I/O Automata verification, E-commerce verification, specification languages
- Free for non-commercial use

Elf, Twelf

- An implementation of LF logical framework.
- LF: Higher–order abstract syntax, Judgments-as-types
- Elf: combining LF style logic definition with λ–Prolog style logic programming.

ACL2

- Next generation of the Boyer-Moore theorem prover, Nqthm
- Logic is first–order quantifier–free
- Language is a subset of Common Lisp, ACL2 is also a programming language. ACL2 is written *in itself*
- Semiautomatic, lemmas as guidance
- Many decision procedures (propositional calculus, equality, arithmetic) and heuristics
- Many applications, primarily hardware verification
- Successful AMD K7 floating–point verification
- Open source (GPL)

Inka

- Inka is a first-order theorem prover with induction.
- Goal: verification

Gandalf

- Gandalf is a resolution automated prover
- Supports:
  - first–order classical logic with equality
  - first–order intuitionistic logic with equality
  - propositional linear logic
  - a fragment of Martin-Löf type theory
- Implemented in Scheme, open source (GPL)

TPS

- TPS (Theorem Proving System) is an automated theorem prover
- Supports classical first–order and higher–order logic
- Supports typed λ–calculus
- Supports automated, semi-automated and interactive modes

Otter

- Automated resolution prover
- Applications: abstract algebra and formal logic
- Implemented in C, open source

Setheo

- Setheo (SEquential THEOrem prover) is a high–performance automated prover
- Uses model elimination.

SPASS

- SPASS is an automated theorem prover for classical first-order logic with equality.

EQP

- EQP (EQuational Prover) is an automated prover for first–order classical equational logic
- Uses associative–commutative unification and matching, a variety of strategies for equational reasoning, and fast search
- Famous for Robbins Algebra proof that made NY Times.

Maude

- Maude is a high-performance equational and rewriting logic language and system
- Maude supports executable specification, prototyping, and programming areas.
- Fast - 100,000s rewrites per second.
- Case studies: software verification.
- License: free for non-commercial use, but without a right to modify.

Omega

- Omega proof planner.
- Goal: mainstream mathematics and mathematical education

Mizar

- Mizar project: a database of formal mathematics (2,000 definitions, 30,000 theorems).
- Mizar helps one to verify a proof, but not to build a proof (???)