Hoops, Loops and Groups
You can do it too!

Bob Veroff
Department of Computer Science
University of New Mexico

AWARD 2005
Argonne National Laboratory
August 11–13, 2005

Objectives of Automated Reasoning

- Mechanizing mathematics
  - repositories for accumulated knowledge
  - enforce formality and rigor
- Automatic theorem proving
  - plug and chug
  - consistently and reliably prove “easy” problems easily
- Automated reasoning assistant
  - mathematically challenging problems
  - “interesting” (to someone)
  - new knowledge (e.g., open questions)
- Real-world applications

Proof Sketches (Motivation)

- Example: $x^3 = x$ problem (ring theory)
  1. $6x = 0$
  2. $3xy + 3yx = 0$
  3. $4xy + 2yx = 0$
  4. $xy = yx$
- Provide guidance without constraining search

As in approaches bases on analogy, proof planning, and abstraction, the point is to have high-level control of the proof search based on some notion of mapping and/or matching.

Proof Sketches (Mechanics)

- Idea: subsumption as part of a search strategy
  - input list of *hint* clauses
  - hint subsumers are notable milestones
  - bias search accordingly
- A *proof sketch* is a "coherent" set of hints
- Uses of proof sketches
  - proof checking and completion (automated referee?)
  - proof mapping
    - strictly forward, AC-free, demod free
    - mapping proofs between Otter and Prover9
  - a method for finding new results (e.g., challenge theorems, finding simple axiom systems)

The Proof Sketches Method

- Proof sketches property
  - emphasis on sufficiency rather than necessity
  - operationally, natural and convenient to use multiple proof sketches simultaneously
- Systematically generate new sketches, including all previous sketches as hints

In some sense, the objective is to transform a proof finding problem into a proof completion problem.

Proof Sketches (The Challenge)

The hints mechanism is straightforward. The interesting problem is the generation of proof sketches and the role these play in the solution to difficult problems.

An idea that has worked very well in several problem domains is to start with simplifying assumptions and systematically eliminate these assumptions in a sequence of experiments (using previous proofs as proof sketches).

What assumptions?

Example Assumptions

- Prove theorem in a stronger theory (e.g., lattice hierarchy)
- Assume generalizations (e.g., by renaming variables)
- Permit term-oriented versions of literal-oriented inferences, e.g., given `P(i(x,n(n(x))))` and `P(i(n(n(x)),x))` permit paramodulation with `n(n(x)) = x`

It also has been effective to

- Prove instances of theorems (e.g., by renaming variables)
- Replace assumptions by proper instances (as part of the process of eliminating assumptions)

Example Applications

- Boolean algebra / lattice Theory
- Distributivity in many-valued logic
- Double negation
- HBCK
- BCK, BCSK, SBL, Nelson algebra, V3 ...
- Median algebra
- ... your problem goes here ...

See, for example,

`http://www.cs.unm.edu/~veroff/{BA,LT,CD,HBCK,MEDIAN_ALGEBRA}/.`

Observations

- Difficult problems (in our terms)
- Wildly different areas
- Little prior knowledge
- Proof sketches played a key role
- Individual steps still may be very difficult

Current Developments

- Autosketches

  Manage sequence of Prover9 searches given an input list of extra assumptions

- Saxfinder

  Find single axioms in a set of candidates

Hopefully coming soon ... automated generation of assumptions.