Automated theorem proving in algebra

David Stanovský

Charles University in Prague
Czech Republic

stanovsk@karlin.mff.cuni.cz
http://www.karlin.mff.cuni.cz/~stanovsk

Beograd, November 2009

Automated theorem proving

INPUT: A finite set of first order formulas
OUTPUT: Satisfiable / Unsatisfiable / I don't know (Timeout)

What is it good for:

- proving theorems in mathematics

Automated theorem proving

INPUT: A finite set of first order formulas
OUTPUT: Satisfiable / Unsatisfiable / I don't know (Timeout)

What is it good for:

- proving theorems in mathematics

What is it really good for:

- formalized mathematics, proof verification
  - proof assistants: Isabelle, HOL, Coq, ...
  - libraries in a formal language: Mizar, ...
- reasoning over large knowledge bases (SUMO, Cyc, ...)
- software verification (Spec#, ESC/Java,...)
- hardware verification (ACL2, PVS, ...)
- etc.

Automated theorem proving

INPUT: A finite set of first order formulas
OUTPUT: Satisfiable / Unsatisfiable / I don't know (Timeout)

Algorithms and implementations:

- Resolution calculus: *Otter/Prover9, E, Spass, Vampire, ...*
- Equational reasoning based on Knuth-Bendix: *Waldmeister*
- Instantiation based reasoning: *iProver, Darwin*
- several experimental techniques

Model building:

- Translation to SAT: *Paradox, Mace4*

Automated theorem proving

INPUT: A finite set of first order formulas
OUTPUT: Satisfiable / Unsatisfiable / I don't know (Timeout)

Algorithms and implementations:

- Resolution calculus: *Otter/Prover9, E, Spass, Vampire, ...*
- Equational reasoning based on Knuth-Bendix: *Waldmeister*
- Instantiation based reasoning: *iProver, Darwin*
- several experimental techniques

Model building:

- Translation to SAT: *Paradox, Mace4*

Benchmarks: http://www.tptp.org

- TPTP library
- CASC competition

Automated theorem proving in mathematics

... not so useful, indeed:

- first order problems within a given theory
- usually doesn't prove anything nontrivial (particularly in set theory)

Automated theorem proving in mathematics

... not so useful, indeed:

- first order problems within a given theory
- usually doesn't prove anything nontrivial (particularly in set theory)

... sometimes may by useful:

- direct proofs of open problems (very rarely successful)
- proving tedious technical steps in classical proofs
- quick experimentation, checking out (often false) conjectures
- exhaustive search

When ATP may outperform a mathematician:

- nonclassical strucures, complicated equations
- finding complicated syntactic proofs
- quick checking for (small) models

... has been applied in: quasigroups and loops, algebraic logic, ...

Automated theorem proving in mathematics

Attempting a problem with ATP:

1. formalization in first order logic
   - almost nothing formalizable directly
   - sometimes a highly non-trivial task
   - which formalization is optimal
2. finding a proof
   - choice of prover
   - parameter setting
   - using advanced strategies (hints, semantic guidance, ...)
3. reading and understanding the proof
   - decode, simplify, structure, ...
   - automatizable?

Existence of a unit element:

$$\exists z\ \forall x\ (x \cdot z = x \ \&\ z \cdot x = x).$$

In quasigroups:

$$x \cdot (y/y) = x \ \&\ (y/y) \cdot x = x,$$
$$x \cdot (y\backslash y) = x \ \&\ (y\backslash y) \cdot x = x.$$

Which choice is the right one?

Distributive groupoids are symmetric-by-medial.

On every idempotent distributive groupoid, there is a congruence α such that $\mathbf{G}/\alpha$ is medial and all blocks are symmetric.

In other words, in groupoids,

$$x * yz = xy * xz, \quad xy * z = xz * yz$$

implies

$$(xy * zu) * ((xy * zu) * (xz * yu)) = xz * yu$$

$$(xy * zu) * (xz * yu) = (xz * yu) * (xy * zu)$$

Bruck loops with abelian inner mapping group are 2-nilpotent.

```
cnf(1,axiom,mult(unit,A) = A).
cnf(2,axiom,mult(A,unit) = A).
cnf(3,axiom,mult(A,i(A)) = unit).
cnf(4,axiom,mult(i(A),A) = unit).
cnf(5,axiom,i(mult(A,B)) = mult(i(A),i(B))).
cnf(6,axiom,mult(i(A),mult(A,B)) = B).
cnf(7,axiom,rd(mult(A,B),B) = A).
cnf(8,axiom,mult(rd(A,B),B) = A).
cnf(9,axiom,mult(mult(A,mult(B,A)),C) =
mult(A,mult(B,mult(A,C)))).
cnf(10,axiom,mult(mult(A,B),C) =
mult(mult(A,mult(B,C)),asoc(A,B,C))).
cnf(11,axiom,op_l(A,B,C) = mult(i(mult(C,B)),mult(C,mult(B,A)))).
cnf(12,axiom,op_r(A,B,C) = rd(mult(mult(A,B),C),mult(B,C))).
cnf(13,axiom,op_t(A,B) = mult(i(B),mult(A,B))).
cnf(14,axiom,op_r(op_r(A,B,C),D,E) = op_r(op_r(A,D,E),B,C)).
cnf(15,axiom,op_l(op_r(A,B,C),D,E) = op_r(op_l(A,D,E),B,C)).
cnf(16,axiom,op_l(op_l(A,B,C),D,E) = op_l(op_l(A,D,E),B,C)).
cnf(17,axiom,op_t(op_r(A,B,C),D) = op_r(op_t(A,D),B,C)).
cnf(18,axiom,op_t(op_l(A,B,C),D) = op_l(op_t(A,D),B,C)).
cnf(19,axiom,op_t(op_t(A,B),C) = op_t(op_t(A,C),B)).

cnf(20,negated_conjecture,asoc(asoc(a,b,c),d,e) != unit).
```

Automated theorem proving in algebra

Milestones:

- since early 1990's: short axioms for various theories
- 1996, W. McCune: Robbins algebras are Boolean algebras
- 1996, K. Kunen: Moufang quasigroups are loops
- since early 2000's: standard technique in loop theory (M. Kinyon, JD Phillips, P. Vojtěchovský)
- recently: algebraic logic

Robbins' problem

(Huntington, 1933) Short axioms for Boolean algebras:

$$x + y = y + x, \qquad (x + y) + z = x + (y + z),$$
$$(x' + y)' + (x' + y')' = x.$$

(Robbins, 1934) Shorter axioms, conjectured to axiomatize BA's:

$$x + y = y + x, \qquad (x + y) + z = x + (y + z),$$
$$((x + y)' + (x + y')')' = x.$$

(Winker, 1979) Sufficient to prove that

$$\text{Robbins} \ \vdash \ (\exists A)(\exists B)\ (A + B)' = A'$$

Confirmed by EQP prover by McCune in 1996, reported in NY Times (!)

Single axioms

(McCune, 1993) The *shortest* axiom for *abelian groups*:

$$((x * y) * z) * (x * z)' = y$$

(Kunen, 1992; McCune, 1993) Short single axioms for *groups*:

3 variables: $((z * (x * y)')' * (z * y')) * (y' * y)' = x$

4 variables: $y * (z * (((w * w') * (x * z)') * y))' = x$

(McCune, Padmanabhan, Veroff, 2002) A short axiom for *lattices*:

$$(((y \vee x) \wedge x) \vee (((z \wedge (x \vee x)) \vee (u \wedge x)) \wedge v)) \wedge (((w \vee x) \wedge (r \vee x)) \vee s) = x$$

(McCune, Veroff, Fitelson, Harris, Feist, Wos, 2002)
A *shortest* axiom for *Boolean algebras* in terms of Sheffer stroke:

$$((x|y)|z)|(x|((x|z)|x)) = z$$

Moufang quasigroups

Quasigroup = latin square = $(G, \cdot)$, all translations are permutations
Loop = quasigroup with a unit = non-associative group

$$x\backslash(x \cdot y) = y, \quad x \cdot (x\backslash y) = y, \quad (y/x) \cdot x = y, \quad (y \cdot x)/x = y$$
$$x \cdot 1 = 1 \cdot x = x$$

Moufang identity (weak associativity):

$$((x \cdot y) \cdot x) \cdot z = x \cdot (y \cdot (x \cdot z))$$

Is every Moufang quasigroup a loop?

Proved with McCune's Otter by Kenneth Kunen in 1996.

Results in quasigroup and loop theory

To date: 28 papers assisted by ATP

(Kinyon, Kunen, Phillips) *Diassociative A-loops are Moufang*

- diassociative = 2-generated subloops are groups
- A-loop = inner mappings are automorphisms
- by hand: in A-loops, diassociativity $\Leftrightarrow$ IP property

(Kepka, Kinyon, Phillips) *Every F-quasigroup is isotopic to a Moufang loop*

- F-quasigroup = several identities
- isotopy to a Moufang loop = easily formalizable
- open problem #1 in Belousov's book
- original proof mostly by hand (only several lemmas by Prover9)
- Waldmeister can prove it in 40 minutes from scratch

And much more...

QPTP = Quasigroup Problems for Theorem Provers

(recently with JD Phillips)

= a collection of results in loop theory obtained with assistance of ATP

- all 28 papers covered, about 100 problems selected (about 80% equational)
- both formal (TPTP) and informal (paper) description
- downloadable at `www.karlin.mff.cuni.cz/~stanovsk/qptp`
- a benchmark (selected provers from CASC):
  Waldmeister $\gg$ E, Gandalf, Prover9, Vampire $\gg$ Spass

Read our paper! :-)

My results

- Linear theories of groupoids
  - automated construction of free groupoids in 2-, 3- and 4-linear theories — sizes 4, 21, 184
  - exhaustive search for about 2 months, followed by a classification theorem for *-linear theories done by hand
- Non-trivial equations for group conjugation
- Distributive groupoids are symmetric-by-medial:
  - $x * yz = xy * xz, \quad xy * z = xz * yz$
    $$\Rightarrow \quad (xy * zu) * ((xy * zu) * (xz * yu)) = xz * yu$$
    $$\Rightarrow \quad (xy * zu) * (xz * yu) = (xz * yu) * (xy * zu)$$
- Simplifying axioms of biquandles
- (with Phillips) loops with abelian inner mapping loops

Combining systems = future of ATP?

(A random choice of recent projects I found interesting.)

- Search for isomorphism/isotopy *invariants* for loops
  - Paradox: generates models
  - HR: searchs for interesting formulas valid in a given model
  - ATP's: prove that invariants cover all models of given size
- *MPTP:* automated reasoning in ZFC
  - Problems for ATP's based on the Mizar library of formalized mathematics
  - MPTP \$100 challenge: automated proof of Bolzano-Weierstraß theorem (with hints)
- *Malarea:* machine learning in service of automated reasoning
  - Reasoning in large theories (like ZFC with some math background)
  - Problem: Which axioms are useful for given problem?
    Machine learning based on syntactical analysis of given conjectures.
  - Relatively succesful on the MPTP challenge

Summary

Automated theorem provers have helped some mathematicians.

Maybe they can help you, too.

References

http://www.prover9.org

http://www.waldmeister.org

http://www.tptp.org

http://www.karlin.mff.cuni.cz/~stanovsk/qptp