

# ATP in algebra

David Stanovský

Charles University in Prague  
Czech Republic

[stanovsk@karlin.mff.cuni.cz](mailto:stanovsk@karlin.mff.cuni.cz)  
<http://www.karlin.mff.cuni.cz/~stanovsk>

Warszawa, April 2009

- Automated theorem proving

**INPUT:** a finite set of first order formulas

**OUTPUT:** Satisfiable / Unsatisfiable / I don't know (Timeout)

- Finding a proof: Prover9, Waldmeister, E, Vampire, ...
- Finding a finite model: Paradox, Mace4, ...

- Automated theorem proving

**INPUT:** a finite set of first order formulas

**OUTPUT:** Satisfiable / Unsatisfiable / I don't know (Timeout)

- Finding a proof: Prover9, Waldmeister, E, Vampire, ...
- Finding a finite model: Paradox, Mace4, ...

- Automated theory building

- Proof verification

- Isabelle, HOL, Coq
- Mizar

- etc.

# Automated theorem proving

**INPUT:** a finite set of first order formulas

**OUTPUT:** Satisfiable / Unsatisfiable / I don't know (Timeout)

**Main troubles:**

- undecidable, fast growth of search space
- first order within a theory (ZFC is difficult)

# Automated theorem proving

**INPUT:** a finite set of first order formulas

**OUTPUT:** Satisfiable / Unsatisfiable / I don't know (Timeout)

Main troubles:

- undecidable, fast growth of search space
- first order within a theory (ZFC is difficult)

Typical use:

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

# Automated theorem proving

**INPUT:** finite set of first order formulas

**OUTPUT:** Satisfiable / Unsatisfiable / I don't know (Timeout)

**Algorithms and implementations:**

- Resolution with paramodulation: Vampire, E, Otter/Prover9
- Term rewriting: Waldmeister
- few other experimental techniques
- model building — translation to SAT: Paradox, Mace4

# Automated theorem proving

**INPUT:** finite set of first order formulas

**OUTPUT:** Satisfiable / Unsatisfiable / I don't know (Timeout)

**Algorithms and implementations:**

- Resolution with paramodulation: Vampire, E, Otter/Prover9
- Term rewriting: Waldmeister
- few other experimental techniques
  
- model building — translation to SAT: Paradox, Mace4

**Benchmarks:** <http://www.tptp.org>

- TPTP library
- CASC competition

# Automated theorem proving in mathematics

- 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
  - decipher, simplify, structure, ...
  - automatizable?

*Existence of a unit element:*

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

In quasigroups:

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

$$x \cdot (y \setminus y) = x \ \& \ (y \setminus y) \cdot x = x.$$

Which choice is the right one?

## *Distributive groupoids are symmetric-by-medial.*

On every idempotent distributive groupoid, there is a congruence  $\alpha$  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(sos, axiom, mult(unit, A) = A).
cnf(sos, axiom, mult(A, unit) = A).
cnf(sos, axiom, mult(A, i(A)) = unit).
cnf(sos, axiom, mult(i(A), A) = unit).
cnf(sos, axiom, i(mult(A, B)) = mult(i(A), i(B))).
cnf(sos, axiom, mult(i(A), mult(A, B)) = B).
cnf(sos, axiom, rd(mult(A, B), B) = A).
cnf(sos, axiom, mult(rd(A, B), B) = A).
cnf(sos, axiom, mult(mult(A, mult(B, A)), C) =
mult(A, mult(B, mult(A, C)))).
cnf(sos, axiom, mult(mult(A, B), C) =
mult(mult(A, mult(B, C)), asoc(A, B, C))).
cnf(sos, axiom, op_l(A, B, C) =
mult(i(mult(C, B)), mult(C, mult(B, A)))).
cnf(sos, axiom, op_r(A, B, C) = rd(mult(mult(A, B), C), mult(B, C))).
cnf(sos, axiom, op_t(A, B) = mult(i(B), mult(A, B))).
cnf(sos, axiom, op_r(op_r(A, B, C), D, E) = op_r(op_r(A, D, E), B, C)).
cnf(sos, axiom, op_l(op_r(A, B, C), D, E) = op_r(op_l(A, D, E), B, C)).
cnf(sos, axiom, op_l(op_l(A, B, C), D, E) = op_l(op_l(A, D, E), B, C)).
cnf(sos, axiom, op_t(op_r(A, B, C), D) = op_r(op_t(A, D), B, C)).
cnf(sos, axiom, op_t(op_l(A, B, C), D) = op_l(op_t(A, D), B, C)).
cnf(sos, axiom, op_t(op_t(A, B), C) = op_t(op_t(A, C), B)).
cnf(goals, negated_conjecture, asoc(asoc(a, b, c), d, e) != unit).
```

## 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:

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

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

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

(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

$$x \setminus (x \cdot y) = y, \quad x \cdot (x \setminus y) = y, \quad (y/x) \cdot x = y, \quad (y \cdot x)/x = y$$

$$x \cdot 1 = 1 \cdot x = x$$

*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](http://www.karlin.mff.cuni.cz/~stanovsk/qptp)
- a benchmark (selected provers from CASC):  
Waldmeister  $\gg$  E, Gandalf, Prover9, Vampire  $\gg$  Spass

Read our paper! :-)

- Distributive groupoids are symmetric-by-medial:
  - $x * yz = xy * xz, \quad xy * z = xz * yz$ 
    - $\Rightarrow (xy * zu) * ((xy * zu) * (xz * yu)) = xz * yu$
    - $\Rightarrow (xy * zu) * (xz * yu) = (xz * yu) * (xy * zu)$
- Simplifying axioms of biquandles
- Complex algebras of subalgebras
- 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
- (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: searches 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

Automated theorem provers have helped some mathematicians.  
Maybe they can help you, too.

- Go, download Prover9, and play :-)
- If you prove a nontrivial theorem, let me know (feedback to developers)

<http://www.prover9.org>

<http://www.waldmeister.org>

<http://www.tptp.org>

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