Automated Theorem Proving for Special Functions: The Next Phase

Prof. Lawrence C Paulson, University of Cambridge
1. Resolution Theorem Proving
Automated theorem proving

- combining a **logical calculus** with **syntactic algorithms**

  - *Full automation* is convenient, but requires a weak calculus.

    - Booleans + arithmetic (SMT)
    - First-order logic (**resolution**)
    - Good for program analysis.

- An **interactive theorem prover**

  - allows the construction of elaborate specifications
  - and formal mathematical proof developments
  - in an **expressive** logic,
  - but reasoning is **laborious**.
Interactive theorem proving

* Typically based on some form of *higher-order logic*
  
* Isabelle, HOL4: classical HOL, with polymorphism
  
* PVS: a classical but dependently-typed HOL
  
* Coq: a constructive type theory
  
* Used for substantial verification projects
  
* … And to formalise major results in group theory, logic, mathematical analysis, etc.
The resolution proof procedure

- Objective is to **contradict** the *negation* of the statement to be proved.
- The negated formula is translated to a *conjunction of disjunctions*.
- A *clause* is a disjunction of literals: atoms or their negations.
- A **resolution step** combines two clauses to yield a new one.
- Producing the **empty clause** terminates the proof: it is the desired contradiction.
A very simple resolution proof

\[ [\exists x P(x) \lor \exists x Q(x)] \land \forall x [P(x) \rightarrow Q(f(x))] \rightarrow \exists x Q(x) \]

\[ P(a) \lor Q(b) \]

\[ \neg P(x) \lor Q(f(x)) \]

\[ Q(b) \lor Q(f(a)) \]

\[ Q(b) \]

\[ \neg Q(x) \]
Applications of resolution

* Highly syntactic problems:
  * Robbins conjecture
  * Completeness of certain axiom systems

* Also in support of interactive theorem proving (Isabelle’s sledgehammer)

* Mainstream mathematical problems can't easily be reduced to a few first-order formulas.

  However, resolution can be modified to solve a class of problems connected with the real numbers…
2. MetiTarski
Resolution for the real numbers

- *MetiTarski* proves first-order statements involving functions such as exp, ln, sin, cos, tan^{-1}

- … using *axioms* bounding these functions by rational functions

- … and *heuristics* to isolate and remove function occurrences

- integrated with the RCF* decision procedures *QEPCAD, Mathematica, Z3*

*RCF (real-closed field): a field that’s first-order equivalent to the reals*
Some easy MetiTarski problems

\[ 0 < t \land 0 < v_f \implies ((1.565 + .313v_f) \cos(1.16t) + (.01340 + .00268v_f) \sin(1.16t))e^{-1.34t} - (6.55 + 1.31v_f)e^{-3.18t} + v_f + 10 \geq 0 \]

\[ 0 \leq x \land x \leq 1.46 \times 10^{-6} \implies (64.42 \sin(1.71 \times 10^6 x) - 21.08 \cos(1.71 \times 10^6 x))e^{9.05 \times 10^5 x} + 24.24e^{-1.86 \times 10^6 x} > 0 \]

\[ 0 \leq x \land 0 \leq y \implies y \tanh(x) \leq \sinh(yx) \]

Each proved in a few seconds!
the basic idea

Our approach involves replacing functions by rational function upper or lower bounds.

We end up with polynomial inequalities: in other words, RCF problems

... and first-order formulae involving +, −, × and ≤ (on reals) are **decidable**.

**RCF decision procedures and resolution** are the core technologies.
A simple proof:

\[ \forall x \ |e^x - 1| \leq e^{|x|} - 1 \]

- absolute value (pos) \(|c| < e^c \vee e^c < 1\)
- absolute value (pos), etc. \(c < 0 \vee e^c < 1\)
- absolute value (neg) \(e^{|c|} < 1 + |e^c - 1|\)
- absolute value (neg) \(0 \leq c \vee e^{-c} < 1 + |e^c - 1|\)
- lower bound: \(1 - c \leq e^{-c}\)
- lower bound: \(1 + c \leq e^c\)
- lower bound: \(1 + c \leq e^c\)
- lower bound: \(1 + c \leq e^c\)
- “magic” \(0 \leq c\)
What about that Magic Step?

\[ 1 \leq e^c \lor 0 \leq c \]

an upper bound for \( \exp(x) \), for \( x \leq 0 \):

\[ e^x \leq \frac{2304}{(-x^3 + 6x^2 - 24x + 48)^2} \]

using that upper bound

\[ 1 \leq \frac{2304}{(-c^3 + 6c^2 - 24c + 48)^2} \lor 0 < c \lor 0 \leq c \]

eliminating the division

\[ (-c^3 + 6c^2 - 24c + 48)^2 \leq 2304 \]

\[ \lor (-c^3 + 6c^2 - 24c + 48)^2 \leq 0 \lor 0 < c \lor 0 \leq c \]

**deleting redundant literals**

\[ 0 \leq c \]
The key: *algebraic literal deletion*

- A list of RCF clauses (algebraic, with no variables) is maintained.
- Every literal of each new clause is examined.
- A literal will be *deleted* if—according to the RCF decision procedure—it is *inconsistent* with its context.
- MetiTarski also uses the decision procedure to detect *redundant* clauses (those whose algebraic part is deducible from known facts).
Examples of literal deletion

- **Unsatisfiable** literals such as \( p^2 < 0 \) are deleted.

- If \( xy + 1 > 1 \) has previously been deduced, then \( x=0 \) will be deleted.

- The context includes the *negations of adjacent literals* in the clause:
  \( z > 5 \) is deleted from \( z^2 > 3 \lor z > 5 \)

- … because quantifier elimination reduces \( \exists z \ [z^2 \leq 3 \land z > 5] \) to **FALSE**.

- Or in our example,
  \[
  \exists x \ [x < 0 \land (-x^3 + 6x^2 - 24x + 48)^2) \leq 2304]
  \]
Architecture

A superposition theorem prover (Joe Hurd's Metis) +

an RCF decision procedure for nonlinear arithmetic

Standard ML code for arithmetic simplification

new inference rules to attack nonlinear terms

Axioms: upper and lower bounds of functions
Inherent limitations

- Only **non-sharp** inequalities can be proved.
- Not suitable for developing mathematics:
  - ugly, mechanical proofs
  - ... relying on approximations alone, not “insights”
- **Nested** function calls? Difficult.
A few (engineering) applications

- Abstracting non-polynomial *dynamical systems* (Denman)
- KeYmaera linkup: nonlinear *hybrid systems* (Sogokon et al.)
- Collision-avoidance projects for NASA (Muñoz & Denman)

In engineering applications, inequalities typically hold “by accident”
PVS: an interactive theorem prover heavily used by NASA

... to verify flight control software, etc

Now PVS uses MetiTarski as an oracle via a trusted interface

... complementing PVS’s branch-and-bound methods for polynomial estimation

In NASA’s ACCoRD project, MetiTarski has been effective!
3. Upper and Lower Bounds
MetiTarski works for any real-valued function that can be approximated by upper and lower bounds.

Bounds valid over various intervals, of varying accuracy and complexity, are chosen automatically.
Some bounds for \( \ln \)

\[
\begin{align*}
\frac{x - 1}{x} &\leq \ln x \leq x - 1 \\
\frac{(1 + 5x)(x - 1)}{2x(2 + x)} &\leq \ln x \leq \frac{(x + 5)(x - 1)}{2(2x + 1)}
\end{align*}
\]

* based on the continued fraction for \( \ln(x+1) \)
* \textit{much} more accurate than the Taylor expansion
* including inaccurate but very simple bounds
Some bounds for exponentials

\[ e^x \geq 1 + x + \cdots + \frac{x^n}{n!} \quad (n \text{ odd}) \]

\[ e^x \leq 1 + x + \cdots + \frac{x^n}{n!} \quad (n \text{ even, } x \leq 0) \]

\[ e^x \leq \frac{1}{(1 - x + \frac{x^2}{2!} - \frac{x^3}{3!})} \quad (x < 1.596) \]

\[ e^x \leq -\frac{x^3 + 12x^2 + 60x + 120}{x^3 - 12x^2 + 60x - 120} \quad (0 \leq x \leq 4.644) \]

From Taylor series, continued fractions, identities.
Bounding $e^x$ from above

\[ cf3 \ x \triangleq - \frac{x^3 + 12x^2 + 60x + 120}{x^3 - 12x^2 + 60x - 120} \]

- Based on a continued fraction
- **Singularity** around 4.644
- All exponential upper bounds must have singularities!
Verifying MetiTarski’s Axioms

- **Taylor series expansions**: already verified (using Isabelle, PVS, etc.) for the elementary functions $\sin$, $\cos$, $\tan^{-1}$, $\exp$, $\ln$.

- **continued fractions**: more accurate; advanced theory

- The axioms for the five transcendental functions have been verified using Isabelle — using simple methods.

- no formalisations of their *general* continued fraction expansions
$$cf3 \ x \geq e^x \quad (0 \leq x \leq 4.644)$$

By the monotonicity of $\ln$, it’s enough to show

$$\ln(cf3 \ x) \geq x$$

Take the derivative of the difference:

$$\frac{d}{dx}[\ln(cf3 \ x) - x] =$$

$$\frac{x^6}{(x^3 - 12x^2 + 60x - 120)(x^3 + 12x^2 + 60x + 120)}$$
Plotting that derivative...

- Singularities at ±4.644
- *Nonnegative* within that interval
Continuing the proof sketch

That derivative is positive provided

\[ x^3 - 12x^2 + 60x - 120 < 0 \]

and in particular if \( 0 < x < 4.644 \). And since

\[ \text{cf3}(0) = 1 = \exp 0 \]

The result follows.

Similar techniques justify a lower bound axiom:

\[ \text{cf3}x \leq e^x \quad (x \leq 0) \]
4. The Next Phase
Correctness concerns

* floating point arithmetic:
  * inevitable rounding errors
  * programmers responsible for correctness

* computer algebra systems:
  assumptions are made, and users are responsible

* automated theorem provers:
  * the system is responsible for correctness
  * users must be prevented from making errors

How can we know that MetiTarski is sound?
MetiTarski soundness questions

- The axioms have been verified.
- MetiTarski produces proofs detailing all first-order reasoning steps.
- Its arithmetic simplification uses straightforward identities.
- So what is left?

Those decision procedure calls.
Cylindrical algebraic decomposition (CAD)

- Given a logical formula involving a set of polynomials in \( n \) variables
- ... *partition* \( \mathbb{R}^n \) into a finite number of cells
- ... such that each polynomial has a *constant sign* on each cell.
- Then quantifiers can be eliminated by picking a member of each cell.

*The computational effort is hyper-exponential in \( n \)!*
Simpler: CAD in one variable

Most MT problems are univariate

Hardly any have more than three variables.

In the one-dimensional case, we just need the roots of the polynomials.
CAD within MetiTarski

* An experimental extension to MetiTarski solves RCF problems

* … while returning detailed proofs. [Univariate problems only]

* To verify these requires a formalisation of the Sturm-Tarski theorem.

* Then MetiTarski could be soundly integrated with interactive theorem provers.
Future aspirations

- MetiTarski works well!
- It will work even better after future improvements to decision procedures.
- *Interactive theorem proving* is also effective in mathematical analysis.
- It is time to formalise substantial bodies of complex analysis, real algebraic geometry, etc,
- … and integrate algebraic and analytical reasoning into our theorem-proving tools.
the Cambridge team

James Bridge
William Denman
Zongyan Huang
to 2008: Behzad Akbarpour
Acknowledgements

- **Edinburgh Team**: Paul Jackson, Grant Passmore, Andrew Sogokon.

- Assistance from J. H. Davenport, J. Hurd, D. Lester, C. Muñoz, E. Navarro-López, etc.

- Supported by the Engineering and Physical Sciences Research Council [grant numbers EP/C013409/1, EP/I011005/1, EP/I010335/1].

 MetiTarski (like Isabelle) is coded in **Standard ML**.