Improved techniques for proving non-linear real arithmetic problems

Grant Passmore and Paul Jackson

University of Edinburgh

Deduction at Scale
Schloß Ringberg
8th March 2011

Problems of interest

- Boolean combinations of equalities and inequalities on polynomials over $\mathbb{R}$, with pure universal or existential quantification

$$\begin{aligned}
&\exists a, d, w, x, x_3, x_7, y, z, g, g_1, g_2 \in \mathbb{R}. \\
&\quad 45dxy - g + 45xy = 0 \quad \wedge \quad g - g_1 - g_2 - 82 > 0 \\
&\quad \wedge \quad w + 1 < 0 \quad \wedge \quad -x + y \geq 0 \\
&\quad \wedge \quad x - 1 \geq 0 \quad \wedge \quad a = 0 \\
&\quad \wedge \quad -a + wz = 0 \quad \wedge \quad x^3y^2 - z = 0 \\
&\quad \wedge \quad -3g_1^2g_2 + 12g_1x_3x_7 - xy - 11x \geq 0
\end{aligned}$$

- . . . maybe also involving transcendental functions

$$\begin{aligned}
&\forall x \in \mathbb{R}. \;\; 0 \leq x \leq 2 \Rightarrow \\
&\quad 14.2\exp(-0.318x) \\
&\qquad -\left[3.3\cos(1.16x) - 0.16\sin(1.16x)\right]e^{-1.34x} < 12
\end{aligned}$$

One common source of such problems is the verification of Hybrid Systems

Hybrid systems

- State contains both discrete and continuous components
- Time evolution described by both discrete state changes and differential equations

Many examples:

- Control engineering
  - transport - air, land, sea, space
  - robotics
  - chemical processes
- Analogue mixed-signal electronics
- Systems biology
- Aircraft collision avoidance

Proving non-linear problems over $\mathbb{R}$

A. If only polynomials, no transcendental functions

- Tarski showed decidability (with arbitrary quantification)
- Implementations exist of several approaches
  - Cylindrical Algebraic Decomposition (CAD) (Collins)
  - Partial CAD (PCAD)
    QEPCAD-B, Reduce/Redlog, Mathematica
  - Cohen-Hormander
  - Real Nullstellensatz/Positivstellensatz + Semidefinite programming (pure $\exists$) (Harrison, Platzer, Quezel, Rümmer)
- PCAD is most practical complete method
  - Still 2^{2^n} time complexity, even on pure $\exists$ and $\forall$ problems
  - Practical limit of 4 or 5 variables

B. If also have transcendental functions

- In general have undecidability
- MetiTarski (Paulson)

The RAHD system

(RAHD = Real Algebra in High Dimensions)

- Developed by Passmore as part of his PhD
- Handles pure $\exists$ or $\forall$ problems involving polynomials
- Provides framework for heuristically combining relevant techniques. e.g.
  - Interval constraint propagation
  - PCAD, especially 'open' fragment (Collins, Hong, McCallum)
  - Gröbner Basis + Nullstellensatz/Positivstellensatz (Tiwari)
  - Virtual Term Substitution (Weispfenning)
  - Simplification
  - Easy-case variable elimination
- Working hypothesis

  Tailored heuristic combinations can do much better on particular families of problems than individual techniques

RAHD Architecture

- RAHD concept of strategy based on idea of tactics/strategies found in interactive theorem provers (e.g. HOL, PVS)
- Goals are conjunctions of equalities and inequalities over polynomials
- Strategies reduce each goal to 0+ subgoals
- Strategy language permits sequencing, guarding, repeating, parameterising by, and trying alternative simpler strategies
- Desire is to open up individual techniques such as PCAD, expressing their components as atomic strategies, and allowing for easy experimentation with variations
- Strategies provide an ease of programmability missing from most computer algebra systems and automatic proof procedures

Exploiting Open PCAD

An example of a strategy.

- PCAD can be made to run much faster on open problems (McCallum)
  - Open problems involve $>$, but not $=$ and $\geq$.
  - Projection phase of PCAD much simpler
  - Can avoid expensive computations with algebraic numbers
- Strategy splits $p \geq 0$ into $p > 0 \vee p = 0$.
- Additional equations feed into Gröbner-basis simplification strategy
- Open PCAD run on conjunctions of $>$ formulas.
- Increase in PCAD performance wins over growth in number of cases
  - In preliminary experiments, saw $10 - 100\times$ speed-up on some examples (See Calculemus '09 paper)

Abstract PCAD

- A new RAHD strategy
- Breaks out internal step of PCAD algorithm as parameter
- Allows use of more general strategies for this step

CAD in a nutshell

For simplicity, stick with pure $\exists$ problems

- Input: $\exists \vec{x}.\phi$ where $\vec{x} = x_1, \ldots x_n$ and ϕ is quantifier free
- Let $P =$ set of all polynomials in ϕ
- A CAD for P is a finite partition of $\mathbb{R}^n$ into cells, such that all $p \in P$ are sign invariant on each cell
- ϕ is truth-invariant over all points in a cell
- Suffices to check truth of ϕ at single sample point in each cell

Partial CAD idea

- Determination of sample points often most expensive phase of CAD, by far
- Sample point coordinates determined sequentially in CAD
  - Consider proving $\exists x, y, z.\ \phi(x, y, z)$
  - At some step might have partially-determined sample point $\langle \_, \_, c \rangle$
  - for next step might generate partial sample points

    $\langle \_, b_1, c \rangle$

    $\vdots$

    $\langle \_, b_k, c \rangle$ where $b_1 < \ldots < b_k$
  - Cf. determination of Boolean assignments in DPLL
- PCAD uses propositional reasoning to check if a partial sample point determines truth of ϕ.
  - E.g. if $\phi = y^2 < z \vee \ldots$
  - If so, can skip tree of partial sample points that elaborate this point

Abstract Partial CAD idea

- Parameterise PCAD algorithm by strategy for more powerful investigation of truth of ϕ on single partial sample points or whole sets of partial sample points
- Given the set of next sample points

  $\langle \_, b_1, c \rangle$

  $\vdots$

  $\langle \_, b_k, c \rangle$ where $b_1 < \ldots < b_k$

  APCAD might try interval constraint propagation to check satisfiability of

  $$b_1 \leq y \leq b_k \wedge \phi(x, y, c)$$

- Have promising very-preliminary results

The AutoPolyFun Project

AutoPolyFun =

Automation with Polynomials and Special Functions

- 4 years from Nov 2010
- Joint with Larry Paulson at Cambridge
- Funds 2 post-docs (inc. Passmore) + 2 PhDs
- On integration and enhancement of RAHD and MetiTarski
- Will explore a variety of applications, including hybrid systems

`http://www.cl.cam.ac.uk/~lp15/Grants/AutoPolyFun/`

MetiTarski

- A prover for inequalities involving special functions

$$\forall x \in \mathbb{R}.\ \ 0 \leq x \leq 2 \Rightarrow$$
$$14.2\exp(-0.318x)$$
$$-\left[3.3\cos(1.16x) - 0.16\sin(1.16x)\right]e^{-1.34x} < 12$$

- Currently combines
  - Metis, Resolution prover for FOL $+ =$ (Hurd)
  - QEPCAD-B (Hong, Brown)
- Formula simplification provided by Metis + axiom set for rational function bounds on special functions
- New Metis rules apply QEPCAD-B to ground purely-algebraic components of clauses:
  1. If an algebraic literal in a clause is subsumed by other literals in the same clause (in context of other clauses), delete it
  2. If whole clause is purely algebraic and is consequence of other clauses, delete it
- Are in process of replacing QEPCAD-B with RAHD

Applications of RAHD of interest

(Directly or perhaps with MetiTarski integration)

- Integration with KeYmaera hybrid systems prover (Platzer)
- Integration with PVS theorem prover
  - NASA interest in aircraft collision avoidance (Mũnoz)
- Hybrid control systems (Navarro)
- SMT integration (e.g. with Z3)
- Proof of SPARK-Ada verification conditions (Praxis)

Conclusions

- Great need for improved non-linear arithmetic reasoning
- General purpose techniques only go so far
- Hope is that strategies tuned to problem classes can go much further
- RAHD provides a framework for rapid assembly and exploration of these tuned strategies

Further reading

- RAHD home page: `http://homepages.inf.ed.ac.uk/s0793114/rahd`
  - Links to development version on googlecode
- *Combined Decision Techniques for the Existential Theory of the Reals*. Passmore and Jackson. Calculemus 2009
- Abstract Gröbner basis work of Passmore & de Moura (see Passmore's home page)