Recent advances in real geometric reasoning

James Davenport [1]
University of Bath, U.K.

9 July 2014

[1]Thanks to: Russell Bradford, Matthew England, David Wilson (Bath), Zongyan Huang (Cambridge), Changbo Chen (Chinese Academy of Sciences, Chongqing), Scott McCallum (Macquarie), Marc Moreno Maza (Western Ontario)

History of Quantifier Elimination

- In 1930, Tarski discovered [Tar51] that the (semi-)algebraic theory of $\mathbf{R}^n$ admitted quantifier elimination

  $\exists x_{k+1} \forall x_{k+2} \ldots \Phi(x_1, \ldots, x_n) \equiv \Psi(x_1, \ldots, x_k)$
- "Semi" = "allowing $>$, $\leq$ and $\neq$ as well as $=$"
- Needed as $\exists y : x = y^2 \Leftrightarrow x \geq 0$
- The complexity of this was indescribable
- In the sense of not being primitive recursive!
- In 1973, Collins [Col75] discovered a much better way:
- Complexity (m polynomials, degree d, n variables, coefficient length l)

$$(2d)^{2^{2n+8}} m^{2^{n+6}} l^3 \tag{1}$$

- Construct a cylindrical algebraic decomposition of $\mathbf{R}^n$, sign invariant for every polynomial
- Then read off the answer

What is a CAD?

A Cylindrical Algebraic Decomposition (CAD) is a mathematical object. Defined by Collins who also gave the first algorithm to compute one. A CAD is:

- a decomposition meaning a partition of $\mathbf{R}^n$ into connected subsets called cells;
- (semi-)algebraic meaning that each cell can be defined by a sequence of polynomial equations and inequalities;
- cylindrical meaning the cells are arranged in a useful manner — their projections are either equal or disjoint.

In addition, there is (usually) a sample point in each cell, and an index locating it in the decomposition

"Read off the answer"

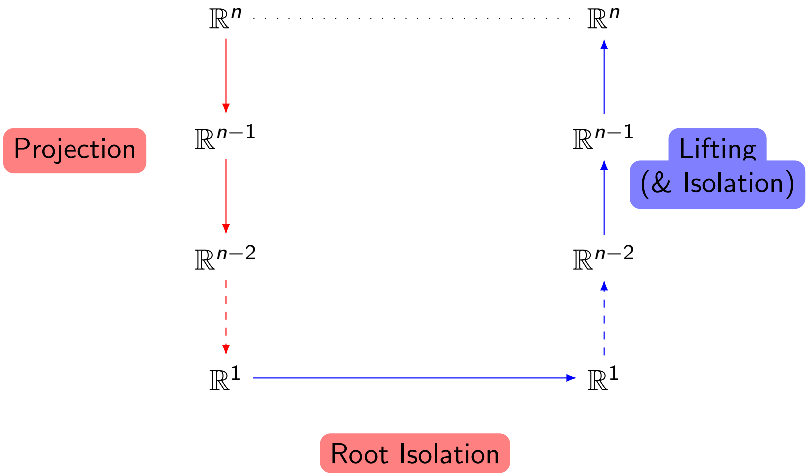
- Each cell is sign invariant, so the the truth of a formula throughout the cell is the truth at the sample point.
- $\forall x F(x) \Leftrightarrow$ "$F(x)$ is true at all sample points"
- $\exists x F(x) \Leftrightarrow$ "$F(x)$ is true at some sample point"
- $\forall x \exists y F(x, y) \Leftrightarrow$ "take a CAD of $\mathbf{R}^2$, cylindrical for y projected onto x-space, then check

$$\forall \text{ sample } x \ \exists \text{ sample } (x, y) : F(x, y) \text{ is true}\text{": finite check}$$

NB The order of the quantifiers defines the order of projection

So all we need is a CAD!

The basic idea for CAD [Col75]



So how do we project?
(Lifting is in fact relatively straight-forward)

Given polynomials $\mathcal{P}_n = \{p_i\}$ in $x_1, \ldots, x_n$, what should $\mathcal{P}_{n-1}$ be?

Naïve (Doesn't work!) Every $\mathrm{disc}_{x_n}(p_i)$, every $\mathrm{res}_{x_n}(p_i, p_j)$

i.e. where the polynomials fold, or cross: misses lots of "special" cases

[Col75] First enlarge $\mathcal{P}_n$ with all its reducta, then naïve plus the coefficients of $\mathcal{P}_n$ (with respect to x_n) the principal subresultant coefficients from the disc_{x_n} and res_{x_n} calculations

[Hon90] a tidied version of [Col75].

[McC88] Let $\mathcal{B}_n$ be a squarefree basis for the primitive parts of $\mathcal{P}_n$. Then $\mathcal{P}_{n-1}$ is the contents of $\mathcal{P}_n$, the coefficients of $\mathcal{B}_n$ and every $\mathrm{disc}_{x_n}(b_i)$, $\mathrm{res}_{x_n}(b_i, b_j)$ from $\mathcal{B}_n$

[Bro01] Naïve plus leading coefficients (not squarefree!)

Are these projections correct?

[Col75] Yes, and it's relatively straightforward to prove that, over a cell in $\mathbf{R}^{n-1}$ sign-invariant for $\mathcal{P}_{n-1}$, the polynomials of $\mathcal{P}_n$ do not cross, and define cells sign-invariant for the polynomials of $\mathcal{P}_n$

[McC88] 52 pages (based on [Zar75]) prove the equivalent statement, but for order-invariance, not sign-invariance, provided the polynomials are well-oriented, a test that has to be applied during lifting.

But if they're not known to be well-oriented?

[McC88] suggests adding all partial derivatives

In practice hope for well-oriented, and if it fails use Hong's projection.

[Bro01] Needs well-orientedness and additional checks

What about the complexity?

If the McCallum projection is well-oriented, the complexity is

$$(2d)^{n2^{n+7}} m^{2^{n+4}} l^3 \qquad (2)$$

versus the original

$$(2d)^{2^{2n+8}} m^{2^{n+6}} l^3 \qquad (1)$$

and in practice the gains in running time can be factors of a thousand, or, more often, the difference between feasibility and infeasibility

"Randomly", well-orientedness ought to occur with probability 1, but we have a family of "real-world" examples (simplification/ branch cuts) where it often fails

Need it be this hard?

The Heintz construction

$$\Phi_k(x_k, y_k) :=$$
$$\exists z_k \forall x_{k-1} y_{k-1} \left[\begin{array}{c} y_{k-1} = y_k \wedge x_{k-1} = z_k \vee y_{k-1} = z_k \wedge x_{k-1} = x_k \\ \Rightarrow \Phi_{k-1}(x_{k-1}, y_{k-1}) \end{array} \right]$$

If $\Phi_1 \equiv y_1 = f(x_1)$, then $\Phi_2 \equiv y_2 = f(f(x_2))$,
$\Phi_3 \equiv y_3 = f(f(f(f(x_3))))$
[DH88] shows $\Omega\left(2^{2^{(n-2)/5}}\right)$ (using $y_R + iy_I = (x_R + ix_I)^4$)
[BD07] shows $\Omega\left(2^{2^{(n-1)/3}}\right)$ (using a sawtooth)
Hence doubly exponential is inevitable, but there's a lot of room!
In fact, there are theoretical algorithms which are singly-exponential in n, but doubly-exponential in the number of $\exists\forall$ alternations

Useful special cases

[McC99] "equational constraints" : when $\Phi \equiv f(x, y, \ldots) = 0 \wedge (\ldots)$

Note If $\Phi \equiv (f_1(x, y) = 0 \wedge g_1(x, y) < 0) \vee (f_2(x, y) = 0 \wedge g_2(x, y) < 0$, which has no obvious equational constraint, we can consider $(f_1 \cdot f_2)(x, y) = 0 \wedge \Phi$, which is equivalent (but higher degree)

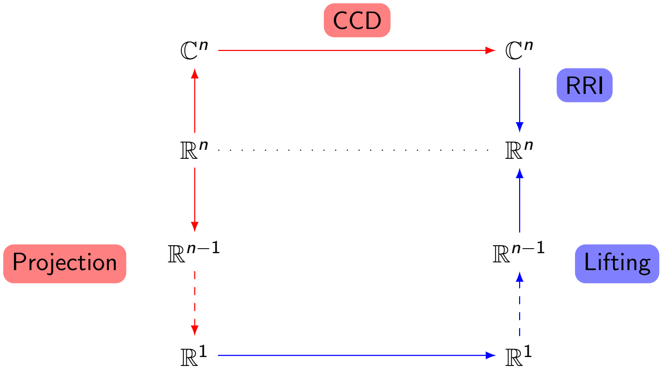
[BDE+13] "truth table invariant CAD" treats this directly

submitted also handles the case where not every clause has an equality (TTICAD)

Roughly speaking, the effect is to reduce n by 1, which square roots the complexity

An alternative approach [CMMXY09]

Proceed via the complex numbers,



Do a complex cylindrical decomposition via Regular Chains
Can be combined with truth table ideas [BCD+14]

Example Complex CD

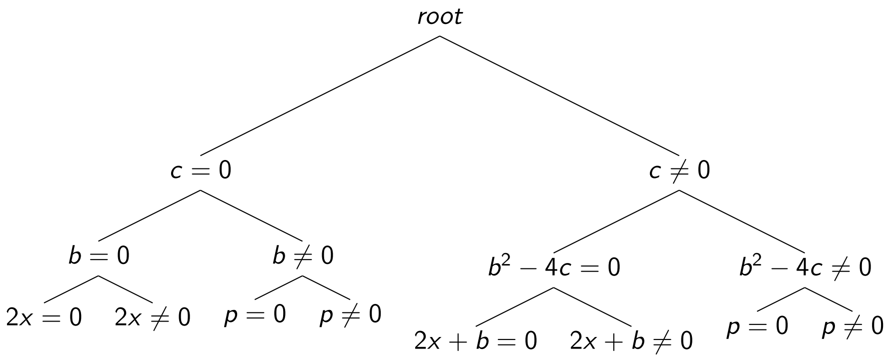


Figure: Complete complex cylindrical tree for the general monic quadratic equation, $p := x^2 + bx + c$, under variable ordering $c \prec b \prec x$.

Note that $b = 0$ is only tested where relevant

So how do I use these tools?

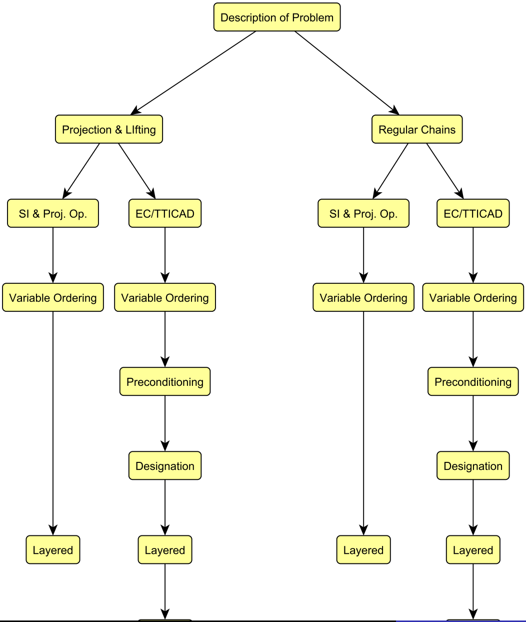
That's actually a very good question: there's a lot of choice in how to phrase the question

1. Choice of variable ordering (where permitted)
2. Choice of equalities
3. Choice of overall technology (Projection/Regular Chains/...)
4. Choice of how the problem is posed
5. (including Gröbner pre-conditioning)

- Choice of software: no software has (even close to) all the techniques, and each has extra "features"

These are **not** independent questions

How might this look? Wilson's thesis



Variable ordering

Theorem ([BD07])

There are CAD problems doubly exponential (in n) for all orderings, and other problems which are doubly exponential (in n) for some orderings, but constant for others

How to tell which case we're in?
How to choose the best (legal) ordering?
This was described in yesterday's CICM talk by Huang:
a variety of heuristics, with a machine-learning meta-heuristic

TTICAD: Kahan Example: [Kah87, pp. 187–189]

With the usual definitions, the conformal map to solve his "fluid flow in a slotted strip" problem

$$w = g(z) := 2\,\mathrm{arccosh}\left(1 + \frac{2z}{3}\right) - \mathrm{arccosh}\left(\frac{5z+12}{3(z+4)}\right) \qquad (3)$$

is the same as the ostensibly more efficient

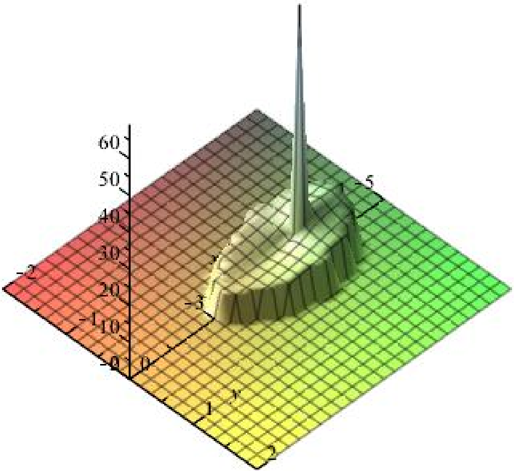
$$w \stackrel{?}{=} q(z) := 2\,\mathrm{arccosh}\left(2(z+3)\sqrt{\frac{z+3}{27(z+4)}}\right), \qquad (4)$$

only if we avoid the teardrop shaped area

$$\left\{z = x + iy : |y| \le \sqrt{\frac{-(x+3)^2(2x+9)}{2x+5}}, -\frac{9}{2} \le x \le -3\right\} \qquad (5)$$

We must analyse the branch cuts of (3) and (4)

Plots of the absolute value of $g(z) - q(z)$.



Analysing the Kahan cuts

One branch cut is

$$\big[8y^3x + 8yx^3 + 20y^3 + 84yx^2 + 288yx + 324y = 0,$$
$$-225x^2 - 324x + 63y^2 - 4x^4 - 52x^3 + 12y^2x + 4y^4 < 0\big].$$

Previous: a sign-invariant CAD would need to be constructed with respect to all polynomials: producing 409 $(x \prec y)$ or 1143 $(y \prec x)$ cells for the Kahan example.

TTICAD: for the sets will suffice for deciding the validity of the simplification with respect to these branch cuts: 55 $(x \prec y)$ and 39 $(y \prec x)$ cells for both projection and lifting TTICAD and regular chains TTICAD.

QEPCAD: 261 and 1143 cells

MATHEMATICA: 72 and 278 cells

More choices

- The most recent Regular Chains algorithm [CMM12] is *incremental*, which means order of clauses matters (for the first time in this field)
- Most previous heuristic work has been based on “size” heuristics such as sotd (sum of total degrees) or total degree, which are order invariant
- Need to develop a new set of guiding principles — England’s talk yesterday

Example from [EBC+14]

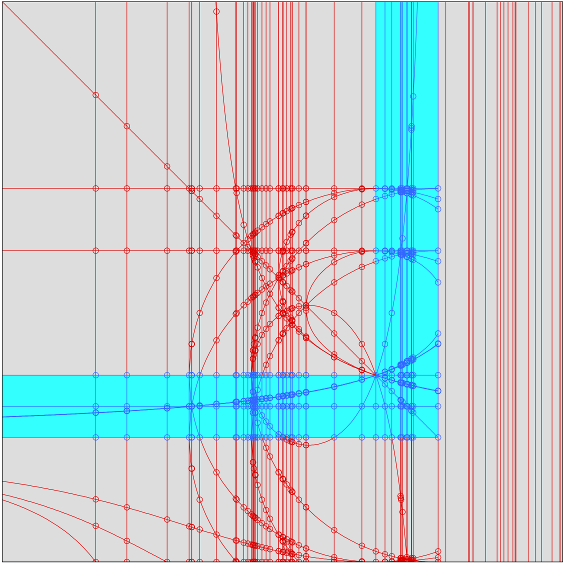
Table: Details on the TTICADs that can be built using RC-TTICAD

$$\phi_1 \vee \phi_2 \text{ where } \begin{array}{ll} \phi_1 & := (f_1 = 0 \wedge h = 0 \wedge a > 0 \wedge a < 2), \\ \phi_2 & := (f_2 = 0 \wedge h = 0 \wedge a > 0 \wedge a < 2) \end{array}$$

Constraint Ordering $\mathfrak{o}$			**TTICAD**		$\mathcal{C}_{\mathfrak{o}}$	
Formula	ϕ_1 order	ϕ_2 order	Cells	Time	sotd	deg
$\phi_1 \to \phi_2$	$h \to f_1$	$h \to f_2$	24545	86.082	16	2
$\phi_1 \to \phi_2$	$h \to f_1$	$f_2 \to h$	73849	499.595	114	8
$\phi_1 \to \phi_2$	$f_1 \to h$	$h \to f_2$	67365	414.314	114	8
$\phi_1 \to \phi_2$	$f_1 \to h$	$f_2 \to h$	105045	1091.918	8	6
$\phi_2 \to \phi_1$	$h \to f_1$	$h \to f_2$	24545	87.378	16	2
$\phi_2 \to \phi_1$	$h \to f_1$	$f_2 \to h$	67365	401.598	114	8
$\phi_2 \to \phi_1$	$f_1 \to h$	$h \to f_2$	73849	494.888	114	8
$\phi_2 \to \phi_1$	$f_1 \to h$	$f_2 \to h$	105045	1075.568	8	6

Note how sotd spectacularly fails to predice the winner!

A 2D CAD of (x, y)-space: moving a ladder [WBDE14]



So might I trust these results?

Trivially: for $\exists$ problems a positive result, or negative for $\forall$ problems, is easily verified (witness computation)

Negative $\exists$: is essentially refutation [JdM12]

Otherwise: we're believing a complicated software package and some maths

[Col75]: Algebra system + 3200LOC + "some maths"

[McC88]: Algebra system + 3200LOC + "a lot of maths"

[CMMXY09]: Algebra system + 5000LOC + "medium maths"

[BDE+13]: Algebra system + 6200LOC + "medium maths"

Proven software? [CM12] does QE (not full CAD), loosely based on [Col75], in COQ, but terribly impractical
Note that CAD has other applications — algebraic simplification [BCD+02], robot path planning [SS83], which tends to require adjacency(unsolved in general dimension)

Bibliography

R.J. Bradford, R.M. Corless, J.H. Davenport, D.J. Jeffrey, and S.M. Watt.
Reasoning about the Elementary Functions of Complex Analysis.
Annals of Mathematics and Artificial Intelligence, 36:303–318, 2002.

R. Bradford, C. Chen, J.H. Davenport, M. England, M. Moreno Maza, and D. Wilson.
Truth table invariant cylindrical algebraic decomposition by regular chains.
Proc. CASC '14 (to appear). Preprint available at `http://opus.bath.ac.uk/38344/`, 2014.

C.W. Brown and J.H. Davenport.
The Complexity of Quantifier Elimination and Cylindrical Algebraic Decomposition.
In C.W. Brown, editor, *Proceedings ISSAC 2007*, pages 54–60, 2007.

R.J. Bradford, J.H. Davenport, M. England, S. McCallum, and D.J. Wilson.
Cylindrical Algebraic Decompositions for Boolean Combinations.
In *Proceedings ISSAC 2013*, pages 125–132, 2013.

C.W. Brown.
Improved Projection for Cylindrical Algebraic Decomposition.
J. Symbolic Comp., 32:447–465, 2001.

Bibliography III

C. Cohen and A. Mahboubi.
Formal Proofs in Real Algebraic Geometry: From Ordered Fields to Quantifier Elimination.
Logical Methods in Computer Science, 8:1–40, 2012.

C. Chen and M. Moreno Maza.
An Incremental Algorithm for Computing Cylindrical Algebraic Decompositions.
http://arxiv.org/abs/1210.5543, 2012.

C. Chen, M. Moreno Maza, B. Xia, and L. Yang.
Computing Cylindrical Algebraic Decomposition via Triangular Decomposition.
In J. May, editor, *Proceedings ISSAC 2009*, pages 95–102, 2009.

G.E. Collins.
Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition.
In *Proceedings 2nd. GI Conference Automata Theory & Formal Languages*, pages 134–183, 1975.

J.H. Davenport and J. Heintz.
Real Quantifier Elimination is Doubly Exponential.
J. Symbolic Comp., 5:29–35, 1988.

M. England, R. Bradford, C. Chen, J.H. Davenport, M. Moreno Maza, and D. Wilson.
Problem formulation for truth-table invariant cylindrical algebraic decomposition by incremental triangular decomposition.
In Press: Accepted for publication in CICM '14, 2014.

H. Hong.
Improvements in CAD-Based Quantifier Elimination.
PhD thesis, OSU-CISRC-10/90-TR29 Ohio State University, 1990.

D. Jovanović and L. de Moura.
Solving Non-Linear Arithmetic.
In *Proceedings IJCAR 2012*, pages 339–354, 2012.

W. Kahan.
Branch Cuts for Complex Elementary Functions.
In A. Iserles and M.J.D. Powell, editors, *Proceedings The State of Art in Numerical Analysis*, pages 165–211, 1987.

S. McCallum.
An Improved Projection Operation for Cylindrical Algebraic Decomposition of Three-dimensional Space.
J. Symbolic Comp., 5:141–161, 1988.

S. McCallum.
On Projection in CAD-Based Quantifier Elimination with Equational Constraints.
In S. Dooley, editor, *Proceedings ISSAC '99*, pages 145–149, 1999.

J.T. Schwartz and M. Sharir.
On the "Piano-Movers" Problem: II. General Techniques for Computing Topological Properties of Real Algebraic Manifolds.
Adv. Appl. Math., 4:298–351, 1983.

A. Tarski.
A Decision Method for Elementary Algebra and Geometry.
2nd ed., Univ. Cal. Press. Reprinted in *Quantifier Elimination and Cylindrical Algebraic Decomposition* (ed. B.F. Caviness & J.R. Johnson), Springer-Verlag, Wein-New York, 1998, pp. 24–84., 1951.

D.J. Wilson, R.J. Bradford, J.H. Davenport, and M. England.
A "Piano Movers" Problem Reformulated.
In *Proceedings SYNASC 2013*, pages 53–60, 2014.

O. Zariski.
On equimultiple subvarieties of algebraic hypersurfaces.
Proc. Nat. Acad. Sci. USA, 72:1425–1426, 1975.