Making Cylindrical Algebraic Decomposition more problem-sensitive

James Davenport[1]
University of Bath
J.H.Davenport@bath.ac.uk

17 May 2018

[1]Thanks to Bruno Buchberger, Matthew England (Coventry), EPSRC EP/J003247/1, EU H2020-FETOPEN-2016-2017-CSA project $\mathcal{SC}^2$ (712689)

Abstract

Collins' [Col75] original motivation for Cylindrical Algebraic Decomposition was to solve Quantifier Elimination over the reals, i.e. reduce $\forall x_{j+1} \exists x_{j+2} \cdots \Phi(x_1, \ldots, x_n)$ to $\Psi(x_1, \ldots, x_j)$, where Φ and Ψ are boolean combinations of polynomial equalities and inequalities. A CAD for this is also a CAD for any other problem involving the same polynomials with the quantified variables in the same order (but possibly different quantifiers). This seems like overkill. [McC99] showed how to do better if Φ can be viewed as $f = 0 \wedge \overline{\Phi}$.

This therefore handles $\Theta := (f_1 = 0 \wedge g_1 > 0) \vee (f_2 = 0 \wedge g_2 > 0)$ by writing it as $f_1 f_2 = 0 \wedge \Theta$. We showed in [BDE+13] how to handle Θ directly, and more efficiently. In [BDE+16] we extended this to cases like $\Xi := (f_1 = 0 \wedge g_1 > 0) \vee g_2 > 0$, where there is no equational constraint.

We will describe these methods, and how they can extend to multiple equational constraints.

Cylindrical Algebraic Decomposition

Problem (Quantifier Elimination)

Given a quantified statement about polynomials $f_i \in \mathbf{Q}[x_1, \ldots, x_n]$

$$\Phi_j := Q_{j+1}x_{j+1} \cdots Q_n x_n \Phi(f_i) \qquad Q_i \in \{\forall, \exists\} \tag{1}$$

produce an equivalent $\Psi(g_i) : g_i \in \mathbf{Q}[x_1, \ldots, x_j]$*: "equivalent"* $\equiv$ *"same real solutions".*

Solution [Col75]: produce a Cylindrical Algebraic Decomposition of $\mathbf{R}^n := \bigcup \{C_\alpha\}$ such that each f_i is sign-invariant on each cell, and the cells are *cylindrical*: $\forall i, \alpha, \beta$ the projections $P_{x_1,\ldots,x_i}(C_\alpha)$ and $P_{x_1,\ldots,x_i}(C_\beta)$ are equal or disjoint. Each cell has a sample point s_i (normally arranged cylindrically) and then the truth of Φ in a cell is the truth at a sample point, and $\forall x_r$ becomes $\bigwedge_{x_r \text{ samples}}$ etc.

Plus/Minus of CAD

+ Solves the problem given, e.g. $\forall x \exists y f > 0 \wedge (g = 0 \vee h < 0)$
– The same structure solves all other problems with the same polynomials and order of quantified variables, e.g. $\forall y f = 0 \vee (g < 0 \wedge h > 0)$

e.g. Can work very hard on trivial examples: $x < -1 \wedge x > 1 \wedge \underbrace{(f_1(x) > 0 \vee \cdots)}_{\text{irrelevant}}$

– Current algorithms can be misled by spurious solutions. Consider $\{x^2 + y^2 - 2, (x-6)^2 + y^2 - 2\}$. Because $x = 3, y = \pm\sqrt{-7}$ is a common zero, current algorithms wrongly regard $x = 3$ as a critical point over $\mathbf{R}^2$ (which it would be over $\mathbf{C}^2$).

The original complexity

When Collins [Col75] produced his Cylindrical Algebraic Decomposition algorithm, the complexity was $O\left(d^{2^{2n+8}} m^{2^{n+6}}\right) l^3 k$, where n is the number of variables, d the maximum degree of any input polynomial in any variable, m the number of polynomials occurring in the input, k the number of occurrences of polynomials (essentially the length) and l the maximum coefficient length.
From now on omit l, k, and assume classical arithmetic.
Given m polynomials of degree d in x_n, we consider P_C:

1. $O(md)$ coefficients (degree $\leq d$)
2. $O(md)$ discriminants and subdiscriminants (degree $\leq 2d^2$)
3. $O(m^2 d)$ resultants and subresultants (degree $\leq 2d^2$)

Then make square-free etc., and repeat.

$$(m, d) \Rightarrow (m^2 d, 2d^2) \Rightarrow (2m^4 d^4, 8d^4) \Rightarrow (32m^8 d^{12}, 128d^8) \Rightarrow \cdots$$

This feed from d to m causes the $d^{2^{2n+O(1)}}$.

McCallum's Notational Idea [McC84]

Problem (Square-free Decomposition)

Generally a good idea, and often algorithmically necessary. But one polynomial of degree d might become $O(\sqrt{d})$ polynomials, but the degree might not reduce. Hence (m, d) gets worse when we "improve" the polynomials.

Say that a set of polynomials is (M, D) if it can be partitioned into $\leq M$ sets, with the sum of the degrees in each set $\leq D$. This *is* preserved under square-free, relatively prime, and even complete factorisation, and behaves well w.r.t. operations.

Proposition

If S is an (M, D) set of polynomials in $(x_1, \ldots, x_n)$, then $\{\mathrm{res}_{x_n}(f_i, f_j) : f_i, f_j \in S\}$ is an $\left(\frac{M(M+1)}{2}, 2D^2\right)$ set,

Why the subresultants? McCallum's solution [McC84]

Essentially because the vanishing of $\mathrm{res}(f, g)$ at $(\alpha_1, \ldots, \alpha_{n-1})$ means that f and g cross above there, but the multiplicity of the crossing is determined by the vanishing of subresultants.

Hence we may need the subresultants to determine the finer points of the geometry if the resultant vanishes on a set of positive dimension.

Given (M, D) polynomials in x_n, we consider P_M:

1. (MD, D) coefficients (equally, (M, D^2))
2. $(M, 2D^2)$ discriminants
3. $(O(M^2), 2D^2)$ resultants

   $(O(M^2), 2D^2)$ in all (no feed from D to M)

Ths works for *order-invariance*, rather than just sign-invariance, as long as no polynomial is identically zero on a set of positive dimension ("well-oriented").

Note the curiosity that a stronger result has a better algorithm.

Lower bounds

Suppose $\Phi_0(x,y)$ defines $y=f_0(x)$. Let $\Phi_i(x_i,y_i) :=$

$$\exists z_i \forall x_{i-1}, y_{i-1} \left[\begin{array}{c} (y_{i-1}=y_i \wedge x_{i-1}=z_i) \\ \vee \\ (y_{i-1}=z_i \wedge x_{i-1}=x_i) \end{array} \right] \Rightarrow \Phi_{i-1}(x_{i-1},y_{i-1}). \tag{2}$$

Then $\Phi_i(x,y)$ defines $y=f_i(x)=f_{i-1}(f_{i-1}(x))$.

Using this "trick", we build large formulae quickly:

[DH88] $d^{2^{n/5+O(1)}}$: complexes, $f_0 := (y_{\Re}+iy_{\Im}) = (x_{\Re}+ix_{\Im})^4 - 1$

[BD07] $m^{2^{n/3+O(1)}}$: reals, $f_0 := y = \left\{ \begin{array}{ll} 2x & (x<\frac12) \\ 2-2x & (x\geq\frac12) \end{array} \right.$

[BD07] Doubly exponential even for factored sparse polynomials.

Note that we have $O(n)$ alternations of quantifiers: this is necessary [Bas99, for example]

But isn't Bézout's degree bound singly exponential in n?

Indeed so, but it applies to $\exists x_2 \ldots \exists x_n f_1 = 0 \wedge \cdots f_n = 0$.
[McC99] showed that Quantifier Elimination on

$$Q_{j+1}x_{j+1}\cdots Q_n x_n \underbrace{\left(f = 0 \wedge \Phi'(g_i)\right)}_{\Phi} \qquad Q_i \in \{\forall, \exists\} \qquad (3)$$

allowed reducing the double exponent of m by 1.
Take $P_M(f) \cup \{res_{x_n}(f, g_i)\}$, then, when $f = 0$, the g_i will be sign-invariant, and hence the truth of Φ is invariant on each cell, even if the truth of Φ' is not.

Extensions (1)

[McC99] applies to

$$Q_{j+1}x_{j+1}\cdots Q_n x_n \underbrace{\left(f_1 = 0 \wedge \Phi'(g_i)\right) \vee \left(f_2 = 0 \wedge \Phi''(h_i)\right)}_{\Phi} \qquad (4)$$

because we can rewrite this as

$$Q_{j+1}x_{j+1}\cdots Q_n x_n \underbrace{f_1 f_2 = 0 \wedge \Phi}_{\Phi_0} \qquad (5)$$

Therefore $P_M(f_1 f_2) \cup \{res_{x_n}(f_1 f_2, g_i)\} \cup \{res_{x_n}(f_1 f_2, h_i)\}$ is the projection to consider.
[BDEW13] reduces this to
$P_M(f_1 f_2) \cup \{res_{x_n}(f_1, g_i)\} \cup \{res_{x_n}(f_2, h_i)\}$.
Further extended by [BDE$^+$16] to cases where $f_i = 0$ only governed parts of the formula, so (5) not applicable.

Extensions (2)

[McC01] extended to

$$Q_{j+1}x_{j+1}\cdots Q_nx_nf_1=0\wedge\cdots\wedge f_r=0\wedge\Phi(g_j) \tag{6}$$

After eliminating x_n, we have (as well as contents and coefficients)

$=0$	All $\mathrm{res}_{x_n}(f_1,f_i)$ $(2\le i\le r)$
Also	All $\mathrm{disc}_{x_n}(f_i)$, all $\mathrm{res}_{x_n}(f_1,g_j)$
and	All $\mathrm{disc}_{x_n}(g_j)$ (somewhat unexpected)

Under assumptions of primitivity, [EBD15] used this to reduce the double exponent of m by r.
But the double exponent of d is still there, and this conflicts with Bézout.

Iterated Resultants [BM09]: $f_i(x, y, z)$

Consider $\mathrm{res}_y\left(\mathrm{res}_x(f_1, f_2), \mathrm{res}_x(f_1, f_3)\right)$. This has degree $O(d^4)$, again apparently contradicting Bézout. Consider the roots

- $O(d^3)$ z: $\exists y, x : f_1(x, y, z) = f_2(x, y, z) = f_3(x, y, z)$
- $O(d^4)$ z: $\exists y, x_1, x_2 :$ $\begin{array}{c} f_1(x_1, y, z) = f_2(x_1, y, z) \\ \wedge f_1(x_2, y, z) = f_3(x_2, y, z) \end{array}$
- These last are (generally) not roots of $\mathrm{res}_y\left(\mathrm{res}_x(f_1, f_2), \mathrm{res}_x(f_2, f_3)\right)$
- Hence a potentially complicated scheme of gcds of resultants
- BB Instead, compute a Gröbner base of the f_i
- But Aren't Gröbner bases doubly exponential?
- Yes but only in the codimension [MR13], so we require that the f_i really reduce the dimension (and we can't extend this to the partial equation constraint setting of [BDE$^+$16])
- And we require that all the polynomials thus appearing are primitive.

Referee: "primitivity is an artificial constraint"

Indeed, it's certainly a tedious constraint.
The key construct from lower bounds in (2) was

$$L_i := (y_{i-1} = y_i \wedge x_{i-1} = z_i) \vee (y_{i-1} = z_i \wedge x_{i-1} = x_i) \qquad (7)$$

This can be rewritten as $L'_i :=$

$$\left[\begin{array}{c} (y_{i-1} - y_i)(y_{i-1} - z_i) = 0 \wedge \underbrace{(y_{i-1} - y_i)(x_{i-1} - x_i)}_{\text{imprimitive}} = 0 \\ \wedge (x_{i-1} - z_i)(y_{i-1} - z_i) = 0 \wedge (x_{i-1} - z_i)(x_{i-1} - x_i) = 0 \end{array}\right] \qquad (8)$$

Let $Q_i := \exists z_i \forall x_{i-1}, y_{i-1}$ and consider $Q_i L_i \Rightarrow (Q_{i-1} L_{i-1} \Rightarrow \Phi_{i-2})$.
We can rewrite this as

$$Q_i Q_{i-1} \neg L'_i \vee \neg L'_{-1} \vee \Phi_{i-2}, \qquad (9)$$

and its negation is

$$\neg \Phi_i := \overline{Q}_i \overline{Q}_{i-1} L'_i \wedge L'_{-1} \wedge \neg \Phi_{i-2}, \qquad (10)$$

so the [DH88, BD07] examples are purely conjunctions of imprimitive equational constraints.

The Lazard projection [Laz94, MPP17]

P_L is very similar to P_M (only needs leading and trailing coefficients).

What is guaranteed is Lazard-invariance, not order-invariance.

Like order-invariance, Lazard-invariance is stronger than sign-invariance (but the two are incomparable).

The lifting process is different: if a polynomial is nullified, we divide through by the nullifying multiple (and therefore locally lift w.r.t. a different polynomial).

Does any of this equational constraint work generalise to the Lazard projection?

Conclusions

1. The true complexity of quantifier elimination comes from the logical structure, especially alternation of quantifiers.
2. True of Virtual Term Substitution methods as well
3. Imprimitive polynomials implicitly encode an $\vee$, hence logical structure.
4. The definition of cylindricity means that the results must be applicable to all quantifier structures (with the variables in the same order).
5. However, while the worst case is very bad, there is a lot that can be done.
6. Standard "Satisfiability Modulo Theories" will always produce conjunctions of elementary formulae, so this special case is worth optimising.

Bibliography

I

S. Basu.
New results on quantifier elimination over real closed fields and applications to constraint databases.
J. ACM, 46:537–555, 1999.

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.

Bibliography II

R.J. Bradford, J.H. Davenport, M. England, S. McCallum, and D.J. Wilson.
Truth table invariant cylindrical algebraic decomposition.
J. Symbolic Computation, 76:1–35, 2016.

R.J. Bradford, J.H. Davenport, M. England, and D.J. Wilson.
Optimising Problem Formulation for Cylindrical Algebraic Decomposition.
In J. Carette *et al.*, editor, *Proceedings CICM 2013*, pages 19–34, 2013.

L. Busé and B. Mourrain.
Explicit factors of some iterated resultants and discriminants.
Math. Comp., 78:345–386, 2009.

Bibliography III

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, and J.H. Davenport.
Improving the Use of Equational Constraints in Cylindrical Algebraic Decomposition.
In D. Robertz, editor, *Proceedings ISSAC 2015*, pages 165–172, 2015.

Bibliography IV

D. Lazard.
An Improved Projection Operator for Cylindrical Algebraic Decomposition.
In *Proceedings Algebraic Geometry and its Applications: Collections of Papers from Shreeram S. Abhyankar's 60th Birthday Conference*, pages 467–476, 1994.

S. McCallum.
An Improved Projection Operation for Cylindrical Algebraic Decomposition.
PhD thesis, University of Wisconsin-Madison Computer Science, 1984.

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

S. McCallum.
On Propagation of Equational Constraints in CAD-Based Quantifier Elimination.
In B. Mourrain, editor, *Proceedings ISSAC 2001*, pages 223–230, 2001.

S. McCallum, A. Parusinski, and L. Paunescu.
Validity proof of Lazard's method for CAD construction.
`https://arxiv.org/pdf/1607.00264v2.pdf`, 2017.

VI

E.W. Mayr and S. Ritscher.
Dimension-dependent bounds for Gröbner bases of polynomial ideals.
J. Symbolic Comp., 49:78–94, 2013.