Clustering in Lazard CAD

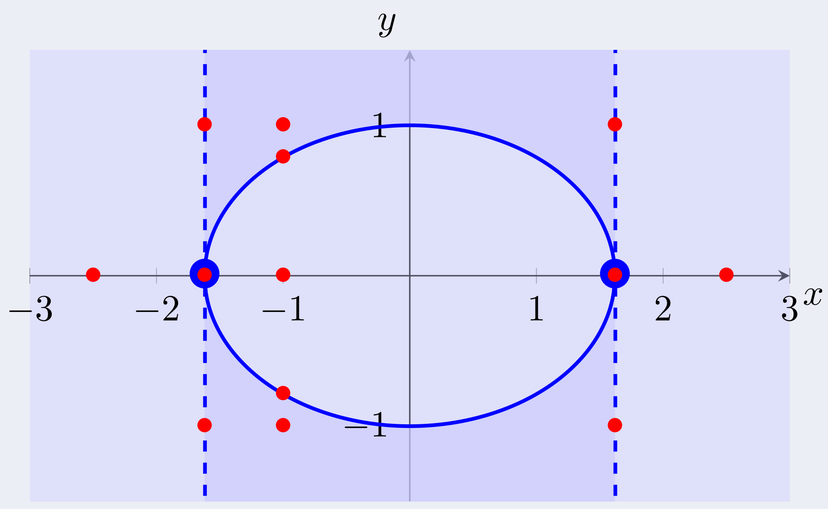
Tereso del Río, AmirHosein Sadeghimanesh and Matthew England
delriot@coventry.ac.uk

1. Introduction

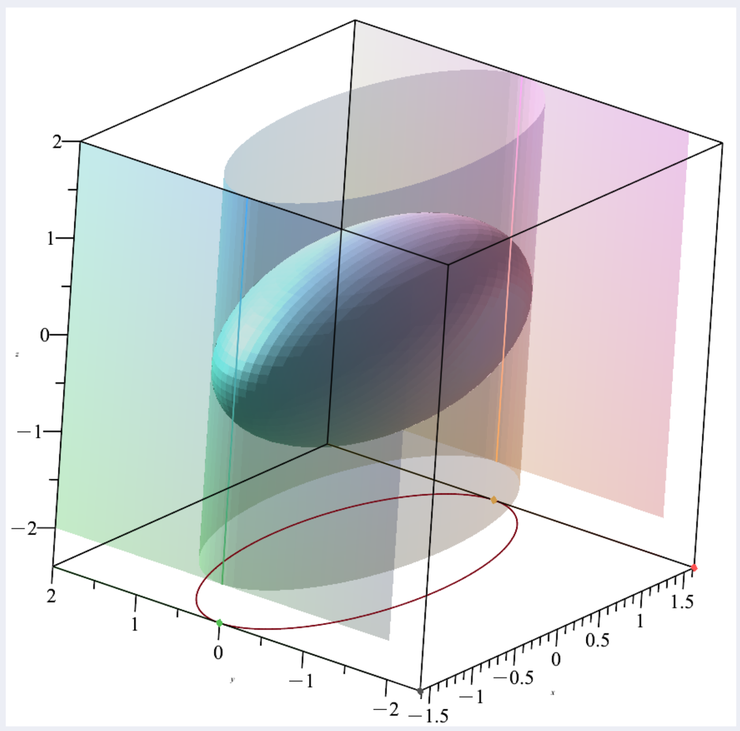
Cylindrical Algebraic Decomposition (CAD) is an algorithm, first proposed by Collins in 1975 [3], that breaks real space into cells in which given polynomials have invariant properties. It does so by producing first a decomposition in one dimension, and then one in two-dimensions by decomposing the cylinder over each 1D cell, and so on. For this, it only needs to study a finite number of points in each step.
CAD has found multiple applications since its inception, ranging from robotics to biology, but its high complexity limits its scope of application in practice. One recent development was the validation of the Lazard projection method for CAD. We focus on an important optimisation, clustering, and how this may be utilised with Lazard projection.

2. Cylindrical Algebraic Decomposition

Given a set of polynomials, the CAD algorithm based on Lazard projection breaks the space into cells in which the given polynomials have invariant properties. It does so by using the following result [2, Theorem 3].
Theorem: *Given the set of polynomials $\mathcal{T} \subset \mathbb{R}[\mathbf{x}, x_{n+1}]$ and $\mathcal{P} \subset \mathbb{R}[\mathbf{x}]$ being the CAD Lazard projection of these polynomials; $\mathcal{T}$ is analytic delineable in any connected analytic submanifold $S \subset \mathbb{R}^n$ in which $\mathcal{P}$ is Lazard valuation-invariant.*
Here, delineable in C means that the Lazard evaluations of those polynomials have a constant root structure above C. Consider for example the set of polynomials $\mathcal{T} = \{1 - 2.62x^2 - y^2 - z^2\}$; its CAD projection with respect to z is $\mathcal{P} = \{1 - 2.62x^2 - y^2\}$. A corresponding CAD of $\mathbb{R}^2$ is given below, with each cell sample point denoted in red.

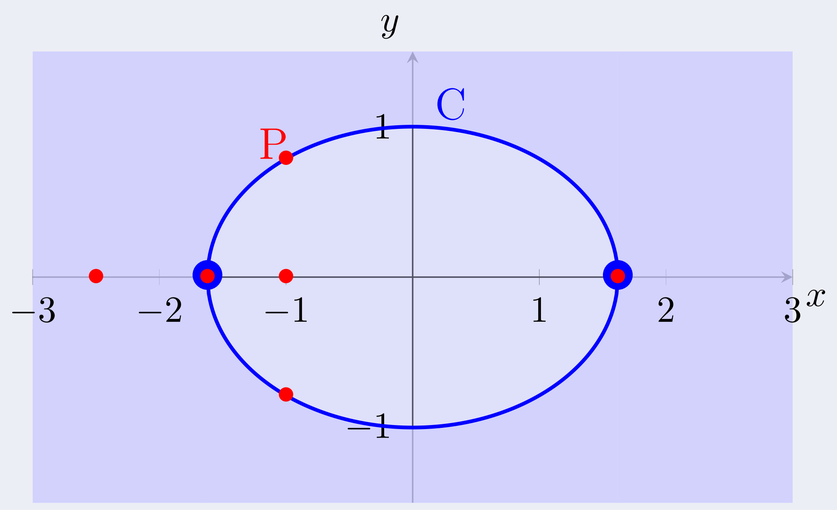


Thanks to the theorem, we know $\mathcal{T}$ has the same root structure above any point from a given cell C, and thus we can study the root structure above one sample point from each cell. It is then possible to decompose $C \times \mathbb{R}$ into cells where $\mathcal{T}$ is valuation-invariant.



3. Clustering in CAD

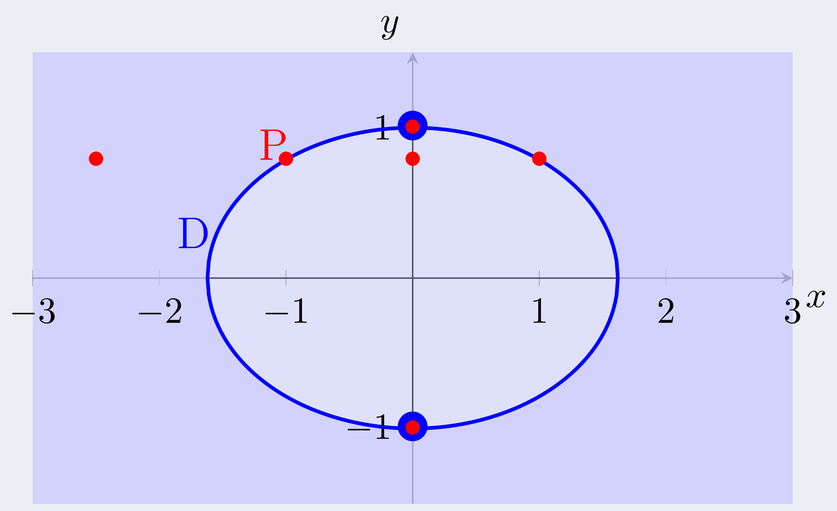
In 1984, Arnon realised that the corresponding theorem for Collins CAD projection refers to a connected region, not necessarily a cell, which is why he proposed to use the same sample point for those cells whose union is connected and share the property for all polynomials [1]. *Clustering* refers to the act of using a single sample point for multiple cells. We note the existance of powerful algorithms to identify such adjacent cells [4].



Clustering reduces the number of sample points over which the root structure should be studied. In our example, to decompose the 3D space from a 2D decomposition only 6 sample points have to be studied instead of 13.

4. Different Orderings give Different Clusters

If we had done the same computations with the ordering $y \prec x$ instead then the resulting clustering would be different.



One may hypothesise that since point P is used as a sample point for both the cell C and the cell D in the two figures of this column, then P could be used as sample point for $C \cup D$, and in fact for the whole border of the circle. However, this assumes that for both orderings the root structure above the same point is going to be the same, but this is not always true. For example, the polynomial $f = yx^2 - y^2 - 2y + zy + 2z$ has, at $(x, y) = (0, 0)$ using the ordering $x \succ y \succ z$, Lazard evaluation $\mathbf{f_{x \succ y \succ z}} = \mathbf{1}$, a polynomials without roots; while using the ordering $y \succ x; \succ z$ we have evaluation $\mathbf{f_{x \succ y \succ z}} = \mathbf{z^2 - 2}$, a polynomial that has two roots.
This shows that the same point can have different root structures for different orderings: so even though it sounds reasonable to further cluster these cells, more theory has to be developed to allow us to safely do so. This is the topic of our future work.

Bibliography

[1] Arnon, D. S., Collins, G. E., and McCallum, S. Cylindrical Algebraic Decomposition I: The Basic Algorithm. *SIAM Journal on Computing 13*, 4 (1984), 865–877.

[2] Brown, C. W., and McCallum, S. Enhancements to Lazard's Method for Cylindrical Algebraic Decomposition. In *Computer Algebra in Scientific Computation*, F. Boulier, M. England, T. Sadykov, and E. Vorozhtsov, Eds., vol. 12291 of *Lecture Notes in Computer Science*. Springer, Cham, 2020, pp. 129–149.

[3] Collins, G. E. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In *Automata Theory and Formal Languages*, H. Brakhage, Ed., vol. 33 of *Lecture Notes in Computer Science*. Springer Verlag, 1975, pp. 134–183.

[4] Strzeboński, A. CAD Adjacency Computation Using Validated Numerics. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation* (2017), M. Burr, C. K. Yap, and M. Safey El Din, Eds., ACM, pp. 413–420.

Acknowledgements

Tereso del Río is supported by Coventry University. Matthew England and AmirHosein Sadeghimanesh acknowledge the support of EPSRC grant EP/T015748/1, *Pushing Back the Doubly-Exponential Wall of Cylindrical Algebraic Decomposition* (the DEWCAD Project).