Projective Delineability for Single Cell Construction*

Jasper Nalbach[1,*], Lucas Michel[2], Erika Ábrahám[1], Christopher W. Brown[1], James H. Davenport[1], Matthew England[1], Pierre Mathonet[1] and Naïm Zénaïdi[1]

[1]RWTH Aachen University, Aachen, Germany
[2]University of Liège, Liège, Belgium
[3]United States Naval Academy, Annapolis, Maryland, USA
[4]University of Bath, Bath UK
[5]Coventry University, Coventry, UK

Abstract

The cylindrical algebraic decomposition (CAD) is the only complete method used in practice for solving problems like quantifier elimination or SMT solving related to real algebra, despite its doubly exponential complexity. Recent exploration-guided algorithms like *NLSAT*, *NuCAD*, and *CAlC* rely on CAD technology but reduce the computational effort heuristically. *Single cell construction* is a paradigm that is used in each of these algorithms.

The central property on which the CAD algorithm is based is called delineability. Recently, we introduced a weaker notion called *projective delineability* which can require fewer computations to guarantee, but needs to be applied carefully. This paper adapts the single cell construction for exploiting projective delineability and reports on experimental results.

Keywords

Cylindrical algebraic decomposition, Real algebra, SMT solving, Single cell construction, Non-linear real arithmetic.

1. Introduction

The *cylindrical algebraic decomposition (CAD)* method enables reasoning about formulas in *real algebra* and is implemented in various tools for *quantifier elimination* like `QEPCAD` [5], `Redlog` [23], `Mathematica` [24], and `Maple` [9], and *satisfiability-modulo-theories (SMT) solving*, like `z3` [12], `cvc5` [2], `yices2` [14], and `SMT-RAT` [11]. Despite its doubly exponential complexity, it is the most widely used complete method for these problems.

The CAD method decomposes the real space into a finite number of connected sets (called *cells*) such that the input set of polynomials have invariant sign in each cell. Although such a decomposition allows for reasoning about the formula, it is usually finer than needed for the task at hand. Algorithms like *NLSAT* [15], *NuCAD* [7], and *CAlC* [1] reduce the computational effort by computing only a set of cells where the input formula is *truth-invariant* that together cover the real space rather than decompose it. These savings are achieved by using the Boolean structure and relation symbols to determine which polynomials are relevant in a certain part (determined by some sample point) of the space, and using

10th International Workshop on Satisfiability Checking and Symbolic Computation, August 2, 2025, Stuttgart, Germany

*E. Ábrahám and J. Nalbach are supported by the German Research Foundation (DFG) as part of AB 461/9-1 (SMT-ART) and J. Nalbach by RTG 2236 (UnRAVeL). Computing resources were granted by RWTH Aachen University under project rwth1560. P. Mathonet, L. Michel and N. Zenaïdi are supported by the FNRS-DFG PDR Weaves (SMT-ART) grant 40019202. M. England and J. Davenport are supported by the UKRI EPSRC DEWCAD Project (grant EP/T015748/1 and EP/T015713/1 respectively). This collaboration was enabled by the project EuroProofNet, supported by European Cooperation in Science and Technology as COST Action CA20111. The views expressed in this paper are those of the authors and do not reflect the official policy or position of the U.S. Naval Academy, Department of the Navy, the Department of Defense, or the U.S. Government.

*Corresponding author.
nalbach@cs.rwth-aachen.de (J. Nalbach)
0000-0002-2641-1380 (J. Nalbach); 0000-0002-4115-7296 (L. Michel); 0000-0002-5647-6134 (E. Ábrahám); 0000-0001-8334-0980 (C. W. Brown); 0000-0002-3982-7545 (J. H. Davenport); 0000-0001-5729-3420 (M. England); 0000-0001-9199-8643 (P. Mathonet); 0009-0005-5520-5147 (N. Zénaïdi)

the shape of the varieties to reduce the computation steps. This leads to cells that are faster to compute and which are usually larger than the cells from the CAD.

More specifically, the CAD algorithm iteratively computes *projection polynomials* to eliminate variable by variable. These consist of *resultants*, *discriminants*, and *(leading) coefficients* which are selected to maintain the *delineability* of the polynomials, a property which allows for work at a sample point to be generalized to a wider cell. The *single cell construction* [6, 8, 21] paradigm is the foundation of the above algorithms and is able to reduce the amount of resultants and discriminants needed to maintain delineability.

This paper investigates when we can go further and leave out leading coefficients from the projection. Towards this goal, in prior theoretical work, we proposed to relax delineability to *projective delineability* [19]: a property that does not require leading coefficients to be maintained. We now report on the embedding of projective delineability within single cell construction, and its impact in experimental results. We continue by recalling the single cell construction in Section 2 and projective delineability in Section 3. Then in Section 4 we present the modification of the single cell construction to use projective delineability, in Section 5 we report an experimental evaluation from our implementation, and finally we conclude in Section 6.

2. Preliminaries

We introduce key background following [20] (full details in preliminaries of [21]).

Let $\mathbb{N}$, $\mathbb{N}_{>0}$, $\mathbb{Q}$, and $\mathbb{R}$ denote the sets of all natural (incl. 0), positive integer, rational, and real numbers respectively. For $i, j \in \mathbb{N}$ with $i < j$, we define $[i..j] = \{i, \dots, j\}$ and $[i] = [1..i]$. For $i, j \in \mathbb{N}_{>0}, j \leq i$ and $r \in \mathbb{R}^i$, we denote by r_j the j-th component of r and by $r_{[j]}$ the vector $(r_1, \dots, r_j)$. Let $f, g : D \to E$ and let $<$ be a total order on E and $\sim \in \{<, =, \neq\}$. We write $f \sim g$ *on* D if $f(d) \sim g(d)$ for all $d \in D$. Note that $f \neq g$ on D is not "not $f = g$ on D".

We work with the *variables* $x_1, \dots, x_n$ with $n \in \mathbb{N}_{>0}$ under a fixed *ordering* $x_1 \prec x_2 \prec \ldots \prec x_n$. A *polynomial* is built from a set of variables and numbers from $\mathbb{Q}$ using addition and multiplication. We use $\mathbb{Q}[x_1, \dots, x_i]$ to denote *multivariate* polynomials in those variables. A polynomial p *is of level* j if x_j is the largest variable in p with non-zero coefficient.

Let $i \in [n]$ and $p, q \in \mathbb{Q}[x_1, \dots, x_i]$ of level i. For $j \in [i]$ and $r = (r_1, \dots, r_j) \in \mathbb{R}^j$ we write $p(r, x_{j+1}, \dots, x_i)$ for the polynomial p after substituting $r_1, \dots, r_j$ for $x_1, \dots, x_j$ in p and indicating the remaining free variables in p. We use $\mathrm{rroots}(p) \subseteq \mathbb{R}^i$ to denote the set of *real roots of* p, $\deg_{x_j}(p)$ to denote the *degree of* p *in* x_j, $\mathrm{coeff}_{x_j}(p)$ for the set of *coefficients of* p *in* x_j, $\mathrm{ldcf}_{x_j}(p)$ for the *leading coefficient of* p *in* x_j, $\mathrm{disc}_{x_j}(p)$ to denote the *discriminant of* p *with respect to* x_j, and $\mathrm{res}_{x_j}(p, q)$ to denote the *resultant of* p *and* q *with respect to* x_j. Let $r \in \mathbb{R}^{i-1}$, then p *is nullified on* r if $p(r, x_i) = 0$.

A *constraint* $p \sim 0$ compares a polynomial $p \in \mathbb{Q}[x_1, \dots, x_i]$ to zero using a relation symbol $\sim \in \{=, \neq, <, >, \leq, \geq\}$, and has *solution set* $\{r \in \mathbb{R}^i \mid p(r) \sim 0\}$. A subset of $\mathbb{R}^i$ for some $i \in [n]$ is called *semi-algebraic* if it is the solution set of a Boolean combination of polynomial constraints. A *cell* is a non-empty connected subset of $\mathbb{R}^i$ for some $i \in [n]$. A cell R is called *simply connected* if any loop in R can be continuously contracted to a point. A polynomial $p \in \mathbb{Q}[x_1, \dots, x_i]$ is *sign-invariant on a set* $R \subseteq \mathbb{R}^i$ if the sign of $p(r)$ is the same for all $r \in R$.

Given $i, j \in \mathbb{N}_{>0}$ with $j < i$, we define the *projection of a set* $R \subseteq \mathbb{R}^i$ *onto* $\mathbb{R}^j$ by $R_{[j]} = \{(r_1, \dots, r_j) \mid \exists r_{j+1}, \dots, r_i.\ (r_1, \dots, r_i) \in R\}$. Given a cell $R \subseteq \mathbb{R}^i$, $i \in [n]$ and continuous functions $f, g : R \to \mathbb{R}$, we define the sets $R \times f = \{(r, f(r)) \mid r \in R\}$ and $R \times (f, g) = \{(r, r_{i+1}) \mid r \in R, r_{i+1} \in (f(r), g(r))\}$ ($R \times (-\infty, g)$, $R \times (f, +\infty)$ analogously).

If $U \subseteq \mathbb{R}^i$ is open, a function $f : U \to \mathbb{R}^n$ is *analytic* if each component of f has a multiple power series representation around each point of U. An i-dimensional *analytic submanifold* of $\mathbb{R}^n$ is a non-empty subset $R \subseteq \mathbb{R}^n$ locally parametrized by coordinates through analytic functions $f : U \subseteq \mathbb{R}^i \to \mathbb{R}^n$. A function f between analytic manifolds R and R' is analytic if locally it has an expression in (analytic) coordinates which is analytic (see also [16]). Let $p \in \mathbb{Q}[x_1, \dots, x_n]$ be a polynomial and $r \in \mathbb{R}^n$ be a point. Then the *order of* p *at* r, $\mathrm{ord}_r(p)$, is defined as the minimum k such that some partial derivative of total order k of p does not vanish at r (and $+\infty$ if $p = 0$). We call p *order-invariant on* $R \subseteq \mathbb{R}^n$ if $\mathrm{ord}_r(p) = \mathrm{ord}_{r'}(p)$ for all $r, r' \in R$ (for details see [17]).

2.1. CAD and Single Cell Construction

A *cylindrical algebraic decomposition* (CAD) [10, 17, 18] is a decomposition $\mathcal{C}$ of $\mathbb{R}^n$ such that each cell $R \in \mathcal{C}$ is semi-algebraic and *locally cylindrical* (i.e. can be described as the solution set of $\psi_1(x_1) \wedge \psi_2(x_1, x_2) \wedge \psi_n(x_1, \ldots, x_n)$ where ψ_i is one of $x_i = \theta(x_1, \ldots, x_{i-1})$ or $\theta_l(x_1, \ldots, x_{i-1}) < x_i < \theta_u(x_1, \ldots, x_{i-1})$ or $\theta_l(x_1, \ldots, x_{i-1}) < x_i$ or $x_i < \theta_u(x_1, \ldots, x_{i-1})$ for some continuous functions $\theta, \theta_l, \theta_u$), and $\mathcal{C}$ is *cylindrically arranged* (i.e. either $n = 1$ or $\{R_{[n-1]} \mid R \in \mathcal{C}\}$ is a cylindrically arranged decomposition of $\mathbb{R}^{n-1}$). The shape of such a CAD allows reasoning about properties of (sets of) polynomials computationally. In particular, it is called *sign-invariant for a set of polynomials* $P \subseteq \mathbb{Q}[x_1, \ldots, x_n]$ if each $p \in P$ is sign-invariant on each $R \in \mathcal{C}$. A sign-invariant CAD for P is computed recursively: to describe the cells' boundaries for x_n, we first compute the underlying decomposition by a projection operation resulting in a set $P' \subseteq \mathbb{Q}[x_1, \ldots, x_{n-1}]$ whose sign-invariant CAD will describe the first $n-1$ levels of the cells of the sign-invariant CAD of P.

The *single cell construction* [8, 21] computes, given a set of polynomials $P \subseteq \mathbb{Q}[x_1, \ldots, x_n]$ and a sample point $s \in \mathbb{R}^n$, a locally cylindrical cell $R \subseteq \mathbb{R}^n$ such that $s \in R$ and such that P is sign-invariant on R. In the rest of this section, we introduce the *levelwise method* [21] for single cell construction.

Delineability. Delineability of a polynomial on some cell means that its variety can be described by continuous functions which are nicely ordered over that cell. This allows us to reason about the polynomial's roots using these functions.

Definition 2.1 (Delineability [18])**.** *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be a cell, and $p \in \mathbb{Q}[x_1, \ldots, x_{i+1}] \setminus \{0\}$. Polynomial p is* delineable on R *if there exist continuous functions $\theta_1, \ldots, \theta_k : R \to \mathbb{R}$, $k \in \mathbb{N}$, such that:*

- *$\theta_1 < \cdots < \theta_k$;*
- *the set of real roots of $p(r, x_{i+1})$ is $\{\theta_1(r), \ldots, \theta_k(r)\}$ for all $r \in R$; and*
- *there exist constants $m_1, \ldots, m_k \in \mathbb{N}_{>0}$ such that for all $r \in R$ and all $j \in [k]$, the multiplicity of the root $\theta_j(r)$ of $p(r, x_{i+1})$ is m_j.*

The θ_j are called real root functions of p on R. *The sets $R \times \theta_j$ are called* sections of p over R.

Analytic delineability *is similar, but R is a connected analytic submanifold of $\mathbb{R}^i$ and the real root functions are analytic.*

The following gives a projection to obtain a cell where a polynomial is delineable.

Theorem 2.1 (Delineability of a Polynomial [18, Thm. 2], [4, Thm. 3.1])**.** *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be a connected analytic submanifold, $p \in \mathbb{Q}[x_1, \ldots, x_{i+1}]$ of level $i+1$. Assume that p is not nullified at any point in R, $\mathrm{disc}_{x_{i+1}}(p)$ is not the zero polynomial and is order-invariant on R, and $\mathrm{ldcf}_{x_{i+1}}(p)$ is sign-invariant on R. Then p is analytically delineable on R and is order-invariant on its sections over R.*

Note that the discriminant of an irreducible polynomial is not the zero polynomial; in our algorithm, we replace each polynomial by its irreducible factors.

Root Orderings. Once we can describe the roots of individual polynomials by ordered root functions on the underlying cell, we can reason about intersections of graphs of root functions from different polynomials, e.g. ensure that two root functions remain in the same order on the underlying cell.

Theorem 2.2 (Lifting of Pairs of Polynomials [21, Thm. A.1])**.** *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be a connected analytic submanifold, $s \in R$, and $p_1, p_2 \in \mathbb{Q}[x_1, \ldots, x_{i+1}]$ of level $i+1$. Assume p_1 and p_2 are analytically delineable on R and $\mathrm{res}_{x_{i+1}}(p_1, p_2)$ is not the zero polynomial and is order-invariant on R. Let $\theta_1, \theta_2 : R \to \mathbb{R}$ be real root functions of p_1 and p_2 on R respectively, and $\sim \in \{<, =\}$ such that $\theta_1(s) \sim \theta_2(s)$. Then $\theta_1 \sim \theta_2$ on R.*

Note that the resultant of two coprime (and irreducible) polynomials is not the zero polynomial.

To maintain that two real root functions θ_1 and θ_2 stay in the same order on R, we could also exploit transitivity using another root function θ_3, e.g. $\theta_1 < \theta_3$ on R and $\theta_3 < \theta_2$ on R implies $\theta_1 < \theta_2$ on R. The work in [21] generalizes this idea to orderings on a set of root functions. This allows for flexibility in the choice of resultants which we compute to maintain certain invariance properties, potentially avoiding the computation of expensive resultants.

Single Cell Construction. Given a set of polynomials $P \subseteq \mathbb{Q}[x_1, \ldots, x_i]$ and a sample $s \in \mathbb{R}^i$, we compute and sort the real roots of $p(s_{[i-1]}, x_i),\ p \in P$. We determine the greatest root $\xi_\ell \in \mathbb{R}$ below (or equal to) s_i and the smallest root $\xi_u \in \mathbb{R}$ above (or equal to) s_i. If they do not exist, we use $-\infty$ and $+\infty$ respectively. We now aim to describe the bounds of the cell $R' \subseteq \mathbb{R}^i$ to be constructed by root functions of some polynomials in P; for that, we assume that all polynomials in P are delineable on the underlying cell $R = R'_{[i-1]}$. Let θ_ℓ and θ_u be real root functions of polynomials in P such that $\theta_\ell(s_{[i-1]}) = \xi_\ell$ and $\theta_u(s_{[i-1]}) = \xi_u$. The bounds on x_i are described by the *symbolic interval* (θ_ℓ, θ_u) (whose bounds depend on $x_1, \ldots, x_{i-1}$) if $\theta_\ell(s_{[i-1]}) < \theta_u(s_{[i-1]})$ or θ_ℓ if $s_i = \xi_\ell = \xi_u$. Now, we use root orderings to make sure that $\theta_\ell < \theta_u$ on R (if applicable) and each $p \in P$ is sign-invariant in $R \times (\theta_\ell, \theta_u)$ resp. $R \times \theta_\ell$:

Theorem 2.3 (Root Ordering for Sign Invariance [21])**.** *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be connected, and $p, p_\ell, p_u \in \mathbb{Q}[x_1, \ldots, x_{i+1}]$ of level $i+1$. Assume that p, p_ℓ, p_u are delineable on R. Let $\theta_\ell, \theta_u : R \to \mathbb{R}$ be real root functions of p_ℓ and p_u on R respectively.*

- *If $\theta_\ell < \theta_u$ on R, and for each real root function θ of p on R it holds $\theta \sim \theta_\ell$ on R for some $\sim \in \{<, =\}$ or $\theta_u \sim \theta$ on R for some $\sim \in \{<, =\}$; then p is sign-invariant on $R \times (\theta_\ell, \theta_u)$.*
- *If for each real root function θ of p on R it holds $\theta_u \sim \theta$ on R for some $\sim \in \{<, =\}$; then p is sign-invariant on $R \times (-\infty, \theta_u)$.*
- *If for each real root function θ of p on R it holds $\theta \sim \theta_\ell$ on R for some $\sim \in \{<, =\}$; then p is sign-invariant on $R \times (\theta_\ell, +\infty)$.*
- *If for each real root function θ of p on R it holds either $\theta < \theta_\ell$ on R, or $\theta_\ell < \theta$ on R, or $\theta = \theta_\ell$ on R; then p is sign-invariant on $R \times \theta_\ell$.*
- *If there is no real root function θ of p on R; then p is sign-invariant on $R \times \mathbb{R}$.*

The single cell construction is given in Algorithm 1. In Algorithm 1, we determine witnesses for the real root functions of a polynomial p of level i on R (the underlying cell to be constructed). Given some $j \in \mathbb{N}_{>0}$, an *indexed root* is a partial function $\mathrm{root}_{x_i}^{p,j} : \mathbb{R}^i \to \mathbb{R}$ that maps $s \in \mathbb{R}^{i-1}$ to the j-th real root of $p(s, x_i)$ if it exists. Given a cell $R \subseteq \mathbb{R}^{i-1}$ where p is delineable, then $\mathrm{root}_{x_i}^{p,j}$ coincides with the root function θ_j from the above definition on R. We thus can evaluate the function computationally by real root isolation. Beginning from Algorithm 1, we compute projection polynomials whose order-invariance on the underlying cell $R = \mathtt{I}_1 \times \cdots \times \mathtt{I}_{i-1}$ (computed in the following iterations) maintain the desired properties of the polynomials in P_i. In Algorithm 1 the algorithm might fail, as McCallum's projection operator cannot reason about cells where a polynomial is nullified [18]. In Algorithm 1, we prevent polynomials from nullifying on any point in the constructed underlying cell by ensuring that at least one coefficient remains non-zero to meet the requirements of the stated theorems (see [21]). Algorithm 1 maintains delineability of each $p \in P_i$ on R, and order-invariance in each of its sections over R. The ordering determined in Algorithm 1 defines a set of resultants to maintain sign-invariance of each $p \in P_i$ in $R \times \mathtt{I}_i$; for this, we analyse $\theta_1, \ldots, \theta_k$ to choose a "good" set, possibly exploiting transitivity. Further, we recall that the cell $\mathtt{I}_1 \times \cdots \times \mathtt{I}_n$ is an analytic submanifold of $\mathbb{R}^n$ as it is bounded by root functions which are analytic by Theorem 2.1.

Example 2.1. *Consider the polynomials $p_1 = 0.5x_1 + 0.5 - x_2$, $p_2 = x_1^2 + x_2^2 - 1$, $p_3 = 0.5x_1 - 0.5 - x_2$, $p_4 = -x_1x_2 - 0.75$ as depicted in Figure 1, along with the sample point $s = (0.25, -0.7)$ and a cell as constructed using Algorithm 1.*

Algorithm 1: `single_cell_construction`(P, s)

Input : finite $P \subseteq \mathbb{Q}[x_1, \dots, x_n]$, $s \in \mathbb{R}^n$

Output : Symbolic intervals $\mathtt{I}_1, \dots, \mathtt{I}_n$ for $x_1, \dots, x_n$ describing a sign-invariant cell for P containing s

foreach $i = n, \dots, 1$ **do**
    $P_i := \{p \in P \mid p \text{ is of level } i\}$, $P := P \setminus P_i$
    determine the set of indexed roots $\Theta = \{\theta_1, \dots, \theta_k\}$ of all $p \in P_i$ that are defined at $s_{[i-1]}$ such that $\theta_1(s_{[i-1]}) \leq \dots \leq \theta_k(s_{[i-1]})$
    `// Determine symbolic interval` $\mathtt{I}_i$
    if $s_i = \theta_j(s_{[i-1]})$ for some j **then** $\mathtt{I}_i := \theta_j$
    else if $\theta_j(s_{[i-1]}) < s_i < \theta_{j+1}(s_{[i-1]})$ for some j **then** $\mathtt{I}_i := (\theta_j, \theta_{j+1})$
    else if $s_i < \theta_1(s_{[i-1]})$ **then** $\mathtt{I}_i := (-\infty, \theta_1)$
    else if $\theta_k(s_{[i-1]}) < s_i$ **then** $\mathtt{I}_i := (\theta_k, +\infty)$
    else $\mathtt{I}_i := (-\infty, +\infty)$
    foreach $p \in P_i$ **do**
        `// Ensure order invariance for each polynomial`
        if $p(s_{[i-1]}, x_i) = 0$ **then return** FAIL
        $P := P \cup \{c\}$ for some $c \in \mathrm{coeff}_{x_i}(p)$ such that $c(s) \neq 0$
        $P := P \cup \{\mathrm{disc}_{x_i}(p), \mathrm{ldcf}_{x_i}(p)\}$ `// delineability, Theorem 2.1`
    choose $\preceq \subseteq \Theta^2$ s.t. its reflexive and transitive closure $\preceq^{rt}$ is a partial order on Θ with $\theta_\ell \preceq^{rt} \theta_u$ (if $\mathtt{I}_i = (\theta_\ell, \theta_u)$) and ensures sign-invariance of each $p \in P_i$ by Theorem 2.3
    $P := P \cup \{\mathrm{res}_{x_i}(p, p') \mid (\mathrm{root}_{x_i}^{p,j}, \mathrm{root}_{x_i}^{p',j'}) \in \preceq\}$ `// Theorem 2.2`

return $\mathtt{I}_1, \dots, \mathtt{I}_n$

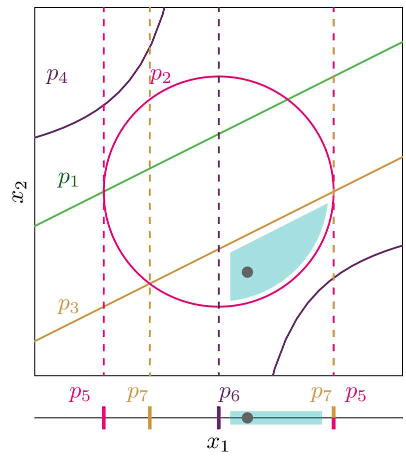


Figure 1: The single cell construction for Example 2.1.

The algorithm adds coefficients and discriminants to guarantee delineability of these polynomials; in this case, we add $p_5 = \mathrm{disc}_{x_2}(p_2)$ *and* $p_6 = \mathrm{ldcf}_{x_2}(p_4)$ *(the others are trivial). Further, as the cell is described by* $\mathtt{I}_2 = (\mathrm{root}_{x_2}^{p_2,1}, \mathrm{root}_{x_2}^{p_3,1})$ *on level* 2 *we aim to maintain* $\mathrm{root}_{x_2}^{p_4,1} < \mathrm{root}_{x_2}^{p_2,1}$ *on* R, $\mathrm{root}_{x_2}^{p_2,1} < \mathrm{root}_{x_2}^{p_3,1}$ *on* R, $\mathrm{root}_{x_2}^{p_3,1} < \mathrm{root}_{x_2}^{p_1,1}$ *on* R*, and* $\mathrm{root}_{x_2}^{p_3,1} < \mathrm{root}_{x_2}^{p_2,2}$ *on* R*. We thus add the corresponding resultants, of which only* $p_7 = \mathrm{res}_{x_2}(p_2, p_3)$ *is non-trivial. On level* 1*, we determine* $\mathtt{I}_1 = (\mathrm{root}_{x_1}^{p_6,1}, \mathrm{root}_{x_1}^{p_5,2})$ *as describing the interval.*

3. Projective Delineability

We now summarize the theory of projective delineability introduced in [19].

Real Projective Line. Roughly speaking, the real projective line $\mathbb{P}$ is defined by adding a single point ∞ to the real line, so $\mathbb{P} = \mathbb{R} \cup \{\infty\}$. We can add more structure to $\mathbb{P}$ or visualize it by using alternative definitions: identifying the real number m with the line $x = my$ and ∞ with $y = 0$, we see that the real projective line $\mathbb{P}$ is the set of lines of $\mathbb{R}^2$ passing through the origin. Such a line is determined by

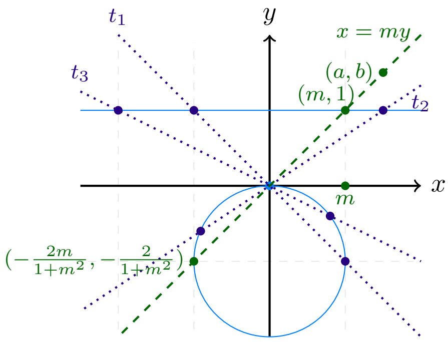


Figure 2: Embedding of $\mathbb{R}$ into $\mathbb{P}$, and an identification with a circle. $(a : b)$ is identified with the dashed green line, the real number m (consider the intersection with $y = 1$), and a point on the unit circle at $(0, -1)$. t_1, t_2, and t_3 are elements of $\mathbb{P}$.

any of its non-zero vectors $(a, b) \in \mathbb{R}^2$ or by any of its non-zero multiples, so if we denote by $(a : b)$ the set (equivalence class) of such vectors, we have $\mathbb{P} = \{(a : b) : (a, b) \in \mathbb{R}^2 \setminus \{(0, 0)\}\}$. This set identifies with $\mathbb{R} \cup \{\infty\}$ by mapping $(a : b)$ to $\frac{a}{b}$ if $b \neq 0$ and to ∞ otherwise. Finally, $\mathbb{P}$ identifies with a circle (as an analytic manifold). Possible identifications are visualized in Figure 2.

As $\mathbb{P}$ identifies with a circle, we cannot use a linear order on $\mathbb{P}$; however, $\mathbb{P}$ has a *cyclic ordering* that extends the usual order on $\mathbb{R}$, as intuitively given in Figure 2. For distinct $t_1, t_2, t_3 \in \mathbb{P}$, we use $[t_1, t_2, t_3]$ to denote that "after t_1, one reaches t_2 before t_3" in that cyclic ordering on $\mathbb{P}$. We use $[t_1, \ldots, t_k]$ for $k > 3$ to denote $\forall j < j' < j'' \in [k].\ [t_j, t_{j'}, t_{j''}]$.

Projective Roots. The introduction of the projective line enables us to handle roots at infinity of (univariate) polynomials, and their multiplicities (see [19, Defn. 2, 3 and 5]): if $p \in \mathbb{Q}[x]$ has degree less than or equal to $d \in \mathbb{N}$, we associate with p the homogeneous bivariate polynomial $H^d(p)$ (also called a binary form of degree d) defined by $H^d(p)(x, y) = y^d p(\frac{x}{y})$. The concepts of roots and multiplicities are well-defined for binary forms, and we thus import them for polynomials: $(a : b) \in \mathbb{P}$ is a projective root of multiplicity k of p with respect to d if (a, b) is a root of $H^d(p)$ with multiplicity k.

The set of projective roots of p (with respect to d) splits into the real roots on the one hand and the root at infinity on the other hand (see [19, Lemmas 2 and 3]): $(a : b) \in \mathbb{P}$ is a projective root of p of multiplicity k w.r.t. d if and only if either $b \neq 0$ and $\frac{a}{b}$ is a real root p of multiplicity k or $b = 0$ and $k = d - \deg(p)$.

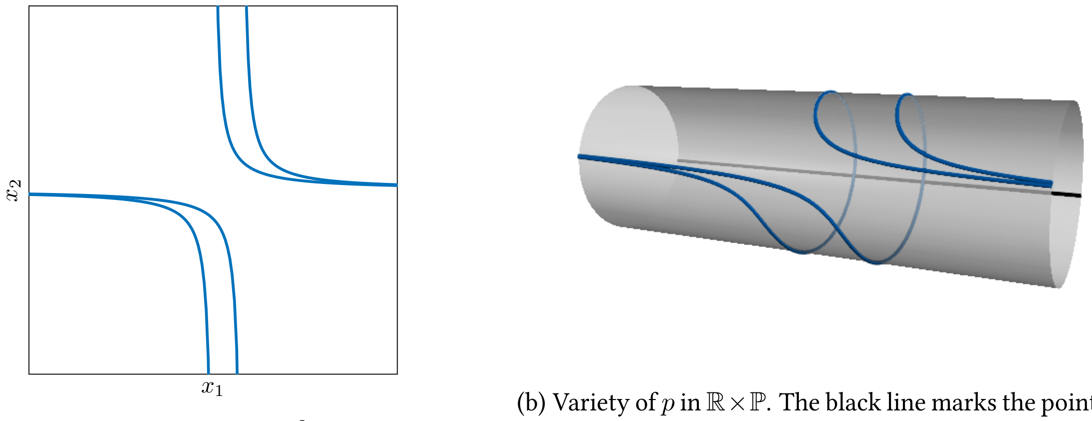
Projective Delineability. We finally formalize the notion of projective delineability, by transferring the concept of projective roots to multivariate polynomials.

Definition 3.1 (Projective Delineability [19, Defn. 11]). *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be a cell, and $p \in \mathbb{Q}[x_1, \ldots, x_{i+1}] \setminus \{0\}$. The polynomial p is called* projectively delineable *on R if there exist continuous functions $\theta_1, \ldots, \theta_k : R \to \mathbb{P}$ (for $k \in \mathbb{N}$) such that:*

- *for any point in R, the values of $\theta_1, \ldots, \theta_k$ are distinct;*
- *the projective roots of the univariate polynomial $p(r, x_{i+1})$ with respect to $\deg_{x_{i+1}}(p)$ are $\theta_1(r), \ldots, \theta_k(r)$ for all $r \in R$; and*
- *there exist constants $m_1, \ldots, m_k \in \mathbb{N}_{>0}$ such that for all $r \in R$ and all $j \in [k]$, the multiplicity of the root $\theta_j(r)$ of $p(r, x_{i+1})$ w.r.t. $\deg_{x_{i+1}}(p)$ is m_j.*

The θ_j are called projective root functions *of p on R. The cells $R \times \theta_j$, $j \in [k]$ are called* projective p-sections *over R.*

Analytic projective delineability *is similar, but R is a connected analytic submanifold of $\mathbb{R}^i$ and the projective root functions are analytic.*



(a) Variety of p in $\mathbb{R}^2$.

(b) Variety of p in $\mathbb{R} \times \mathbb{P}$. The black line marks the points at ∞.

Figure 3: $p = (x_1 x_2 - 1)((x_1 - 1)x_2 - 1)$ is projectively delineable on $\mathbb{R}$, described by two root functions which cross ∞.

In particular, the first condition means the function values as points around the unit circle maintain a cyclic ordering. Figure 3 illustrates an example of a polynomial that is projectively delineable.

The central theorem for this work states that order-invariance of the discriminant plus non-nullification is enough to guarantee projective delineability, with no need to maintain the sign-invariance of the leading coefficient. Note that the theorem requires the underlying cell to be simply connected, which is stronger than connectedness. This assumption is always met for locally cylindrical cells, since they are homeomorphic to open cubes of Euclidean spaces [3, Proposition 5.3] (which are simply connected).

Theorem 3.1 (Projective Delineability [19, Thm. 2])**.** *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be a simply connected analytic submanifold, and $p \in Q[x_1, \ldots, x_{i+1}]$ of level $i+1$. If p is not nullified on any point in R and $\mathrm{disc}_{x_{i+1}}(p)$ is not the zero polynomial and is order-invariant on R, then p is analytically projectively delineable on R and p is order-invariant in each projective p-section over R.*

Delineability is guaranteed by projective delineability plus sign-invariance of the leading coefficient.

Lemma 3.1 (Delineability and Projective Delineability [19, Cor. 1])**.** *Let $i \in \mathbb{N}_{>0}$, $R \subseteq \mathbb{R}^i$ be connected, and $p \in \mathbb{Q}[x_1, \ldots, x_{i+1}]$ of level $i+1$. Assume that p is projectively delineable on R, and $\mathrm{ldcf}_{x_{i+1}}(p)$ is sign-invariant on R. Then p is delineable on R.*

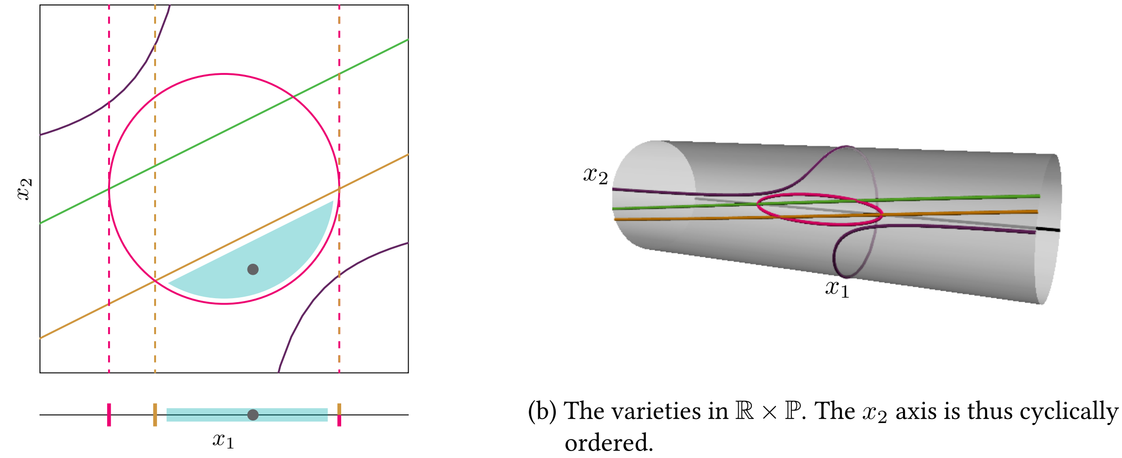
The definition of projective delineability is equivalent to delineability over cells where the polynomial does not have any roots. In those cases, we can guarantee delineability without sign-invariance of the leading coefficient.

Lemma 3.2. *Let $i \in \mathbb{N}_{>0}$, $R \subseteq \mathbb{R}^i$, $s \in R$, and $p \in \mathbb{Q}[x_1, \ldots, x_{i+1}]$ of level $i+1$. If p is projectively delineable on R, $\mathrm{ldcf}_{x_{i+1}}(p)(s) \neq 0$, and $\mathrm{rroots}(p(s, x_{i+1})) = \emptyset$, then p is delineable on R.*

Proof. Let $\theta_1, \ldots, \theta_k$ be the projective p-sections over R. If $\mathrm{ldcf}_{x_{i+1}}(p)(s) \neq 0$, then it holds $\deg_{x_{i+1}}(p(s, x_{i+1})) = \deg_{x_{i+1}}(p)$, thus the projective roots $p(s, x_{i+1})$ are all real. Thus $\{\theta_1(s), \ldots, \theta_k(s)\} = \mathrm{rroots}(p(s, x_{i+1})) = \emptyset$, and it follows $k = 0$. It follows that $p(r, x_{i+1}) \neq 0$ for all $r \in R$, and thus p is delineable on R. □

Root Orderings. As single cell construction relies on root orderings, we give the analogous statement for projective delineability. Note that in contrast to Theorem 2.2, we can only ensure that two root functions are disjoint or equal.

Theorem 3.2 (Lifting of Pairs of Polynomials [19, Theorem 3])**.** *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be a connected analytic submanifold, $s \in R$, and $p_1, p_2 \in \mathbb{Q}[x_1, \ldots, x_{i+1}]$ of level $i+1$. Assume p_1 and p_2 are analytically*



(a) The cell which we aim to construct.

(b) The varieties in $\mathbb{R} \times \mathbb{P}$. The x_2 axis is thus cyclically ordered.

Figure 4: Motivational example for projective delineability.

projectively delineable on R and $\mathrm{res}_{x_{i+1}}(p_1, p_2)$ is not the zero polynomial and is order-invariant on R. Let $\theta_1, \theta_2 : R \to \mathbb{P}$ be real projective root functions of p_1 and p_2 respectively, and $\sim \in \{=, \neq\}$ such that $\theta_1(s) \sim \theta_2(s)$. Then $\theta_1 \sim \theta_2$ on R.

4. Projective Delineability in Single Cell Construction

We motivate the use of projective delineability by considering Example 2.1 again.

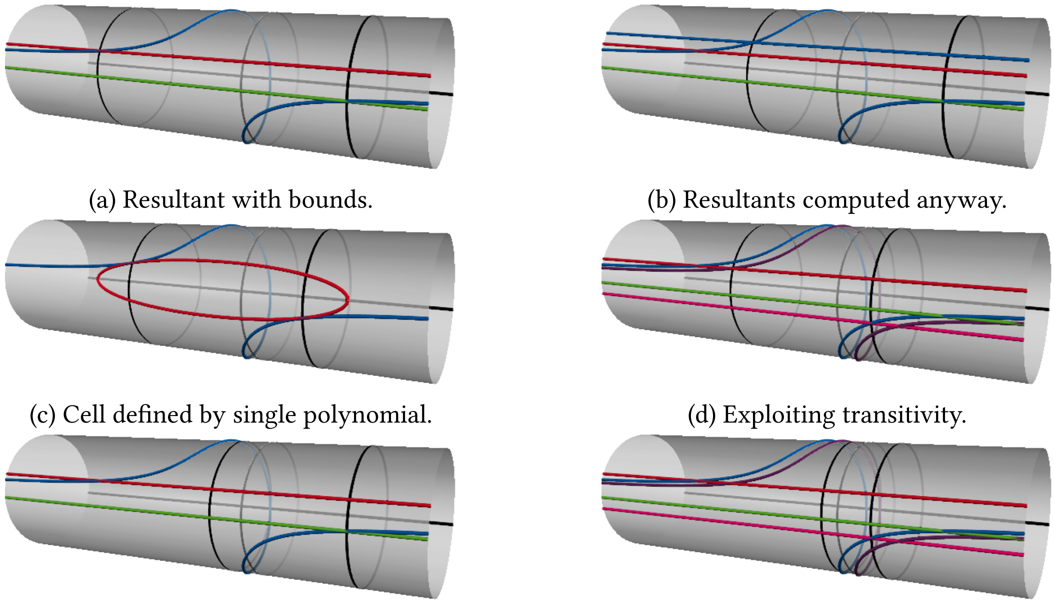
Example 4.1. *The singularity of p_4 (witnessed by its leading coefficient) was a boundary to the cell in Figure 1, but crossing that boundary does not change the sign of any input polynomial. Figure 4a shows the cell that we aim to construct instead: if we detect that the singularity of p_4 does not affect the cell, we can omit the leading coefficient of p_4 and build the enlarged version.*

For this reasoning, we view the roots of the polynomials in the projective real line, as depicted in Figure 4b, where intuitively $-\infty$ and $+\infty$ are identified with the same point ∞. Above the singularity of p_4, the two distinct root functions of p_4 coincide at the point ∞, and thus can be described as a unique real projective root function of p_4. The order-invariance of the discriminant of p_4 ensures that the variety of p_4 can be described using such projective root functions (Theorem 3.1). Now, by adding the resultant of p_2 and p_4, we ensure that the mentioned root function does not intersect the circle, and thus does not enter the cell (Theorem 3.2). By projective delineability of p_4, we know that there are no other roots (Definition 3.1), and we thus can omit the leading coefficient from our projection polynomials.

We modify the single cell construction algorithm as follows. We still describe the cell boundaries using real root functions, as we can encode them using indexed roots which we can evaluate in a straightforward way. Thus, their defining polynomials are still required to be delineable. For the other polynomials, however, we only maintain projective delineability, and hence allow their roots to go through the point at ∞. We thus need to adapt Theorem 2.3, as a root function may not stay below (above) the lower (upper) bound even if it does not cross it by going through ∞.

Theorem 4.1 (Projective Root “Ordering” for Sign Invariance). *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be connected, $s \in R$, and $p, p_\ell, p_u \in \mathbb{Q}[x_1, \ldots, x_{i+1}]$ of level $i+1$. Assume that p is projectively delineable on R, p_ℓ, p_u are delineable on R. Let $\theta_\ell, \theta_u : R \to \mathbb{R}$ be real root functions of p_ℓ and p_u on R respectively.*

- *If $\theta_\ell < \theta_u$ on R, and for each projective root function θ of p on R either $\theta = \theta_\ell$ on R, $\theta = \theta_u$ on R, or $\theta_\ell \neq \theta \neq \theta_u$ on R and $[\theta_\ell(s), \theta_u(s), \theta(s)]$; then p is sign-invariant on $R \times (\theta_\ell, \theta_u)$.*
- *If for each projective root function θ of p on R either $\theta = \theta_u$ on R, or $\infty \neq \theta \neq \theta_u$ on R and $[\infty, \theta_u(s), \theta(s)]$; then p is sign-invariant on $R \times (-\infty, \theta_u)$.*



(a) Resultant with bounds. (b) Resultants computed anyway.

(c) Cell defined by single polynomial. (d) Exploiting transitivity.

(e) Adding the leading coefficient. (f) Combining the previous two ideas.

Figure 5: Projective root "orderings" that guarantee sign-invariance. The grey line indicates the first coordinate of the sample point. We aim to describe the cell bounded by the green and red polynomials. The black lines indicate the resulting cell boundaries on the first dimension.

- *If for each projective root function θ of p on R either $\theta = \theta_\ell$ on R, or $\theta_\ell \neq \theta \neq \infty$ on R and $[\theta(s), \theta_\ell(s), \infty]$; then p is sign-invariant on $R \times (\theta_\ell, +\infty)$.*
- *If for each projective root function θ of p on R either $\theta \neq \theta_\ell$ on R, or $\theta = \theta_\ell$ on R; then p is sign-invariant on $R \times \theta_\ell$.*
- *If there is no projective root function θ of p on R; then p is sign-invariant on $R \times \mathbb{R}$.*

Example 4.2. *This theorem only allows to omit leading coefficients if the interval is bounded in both directions (otherwise we need the leading coefficient for preventing crossing ∞) and the resultant of the polynomial with the polynomials defining the lower and upper bound are computed: e.g. we can omit the leading coefficient of blue polynomial in Figure 5a (which we would add for its delineability in the classical setting) by additionally adding the resultant of the blue and red polynomial (which is not added in the classical setting). The trade-off may not be attractive; but if the polynomial has a root below and above the cell at the current sample point (e.g. Figure 5b), or if both bounds are defined by the same polynomial (e.g. Figure 5c), this would not require additional resultants.*

The single cell construction allows for flexibility for different sets of resultants to maintain sign-invariance by exploiting the transitivity of root orderings. In the projective real line, there is only a cyclic ordering. The following insight transfers the idea to the new setting.

Lemma 4.1 (Transitive Projective Root "Ordering"). *Let $i \in \mathbb{N}$, $R \subseteq \mathbb{R}^i$ be connected, $s \in R$, $\theta_1, \theta_2, \theta_3, \theta_4 : R \to \mathbb{P}$ be continuous. Assume that $[\theta_1(s), \theta_2(s), \theta_3(s), \theta_4(s)]$ and $\theta_1 \neq \theta_2 \neq \theta_3 \neq \theta_4 \neq \theta_1$ on R. Then $\theta_1 \neq \theta_3$ on R.*

Example 4.3. *Consider a projective root (as depicted in blue in Figures 5d to 5f) that is below the symbolic interval at the current sample point. Preventing the root from crossing the lower bound is analogous to the case of regular delineability. However, we now also need to prevent it to cross the upper bound; for that, we now have three options. (1) We compute a chain of resultants that maintain a cyclic ordering of the given root, the upper bound and some roots in between (in the cyclic sense), exploiting Theorem 3.2 (in Figure 5d, by adding the resultants of the blue with the purple and magenta polynomials respectively, its root is trapped between the other; additionally, we add the resultant of the purple with the red one, and of* *the magenta with the green). (2) We add the leading coefficient of the defining polynomial to the projection, avoiding the intersection of the root with ∞, thus making it delineable using Lemma 3.1 (see Figure 5e). (3) We mix the two approaches, e.g. for some polynomial in the chain maintaining the cyclic ordering, we add its leading coefficient to avoid crossing ∞ (in Figure 5f, like (1) the blue root is trapped between the magenta and purple, however, instead of the resultant of the purple and red polynomials, we add the leading coefficient of the purple).*

To summarize, we replace Lines 9 to 14 of Algorithm 1 by Algorithm 2. We maintain projective delineability for each polynomial (Algorithm 2) using Theorem 3.1, and delineability using Lemma 3.1 for polynomials defining the bounds (Algorithm 2) and for polynomials without roots (Algorithm 2) where we make use of the optimization by Lemma 3.2 to omit leading coefficients. In Algorithm 2, we determine a relation for maintaining sign-invariance of polynomials (Theorem 4.1), where we may exploit transitivity (using Lemma 4.1), now maintaining a cyclic ordering. The relation involves ∞ to enable the choice of adding leading coefficients (Algorithm 2) using Lemma 3.1 instead of resultants (Algorithm 2) using Theorem 3.2.

We now elaborate how me make determine the ordering in Algorithm 2, exploiting the options elaborated in Example 4.3. [21] describes some heuristics for choosing root orderings (in the non-cyclic setting). The BIGGEST CELL heuristic is the straightforward ordering that fulfils the minimal requirement from Theorem 2.3, and the LOWEST DEGREE BARRIER heuristic greedily minimizes the degrees of the computed resultants using transitivity. In our modification, our aim is - compared to the classical setting - to avoid additional resultants but omit leading coefficients whenever possible. We thus compute the ordering according to the BIGGEST CELL or LOWEST DEGREE BARRIER heuristic as in the classical setting (as if every polynomial would be delineable). To transfer this ordering to the projective delineability setting, for each root θ above (below) the cell, we either add (θ, ∞) (effectively adding a leading coefficient) or (θ, θ') for some root θ' below (above) the cell (effectively adding a resultant with a polynomial that has roots on the "other" side). As we do not want to add additional resultants, we do the latter only if the corresponding resultant would have been added in the non-projective case.

5. Experiments

We implemented our single cell construction algorithm based on the proof system described in [21] in our solver SMT-RAT, which uses it for generating explanations for the NLSAT algorithm. For this paper, we test the following variants: BC and LDB are the baseline variants using the BIGGEST CELL and LOWEST DEGREE BARRIERS heuristic respectively. BC-PD and LDB-PD are the modified versions using projective delineability as described above. Although we use the incomplete McCallum's projection operator, the implementation of our proof system is complete: in case a polynomial is nullified, we add some of its partial derivatives to ensure its order invariance, as suggested in [17, Section 5.2].

We conduct our experiments on Intel®Xeon®8468 Sapphire CPUs with 2.1 GHz per core, testing upon the SMT-LIB *QF_NRA* benchmark set [22] which contains 12 154 instances. We use a time limit of

Algorithm 2: Modifications of `single_cell_construction`

9 **foreach** $p \in P_i$ **do**
10   **if** $p(s_{[i-1]}, x_i) = 0$ **then return** FAIL
11     $P := P \cup \{c\}$ for some $c \in \mathrm{coeff}_{x_i}(p)$ such that $c(s) \neq 0$
12     $P := P \cup \{\mathrm{disc}_{x_i}(p)\}$ `// proj. delineability, Theorem 3.1`
13 $P := P \cup \{\mathrm{ldcf}_{x_i}(p_\ell), \mathrm{ldcf}_{x_i}(p_u)\}$ where p_ℓ,p_u define $\mathtt{I}_i$ `// del., Lemma 3.1`
14 **foreach** $p \in P_i$ s.t. $\mathrm{rroots}(p(s_{[i-1]}, x_i)) = \emptyset$ and $\mathrm{ldcf}_{x_i}(p) \neq 0$ **do** `// Lemma 3.2`
15     $P := P \cup \{\mathrm{ldcf}_{x_i}(p)\}$ `// delineability, Lemma 3.1`
16 **choose** symmetric $\not\approx \subseteq (\Theta \cup \{\infty\})^2$ that ensures $\theta_\ell \neq \theta_u$ (if $\mathtt{I}_i = (\theta_\ell, \theta_u)$) and sign-invariance of each $p \in P_i$ by Theorem 4.1, using "transitivity" by Lemma 4.1
17 $P := P \cup \{\mathrm{res}_{x_i}(p, p') \mid (\mathrm{root}_{x_i}^{p,j}, \mathrm{root}_{x_i}^{p',j'}) \in \not\approx\}$ `// Theorem 3.2`
18 $P := P \cup \{\mathrm{ldcf}_{x_i}(p) \mid (\mathrm{root}_{x_i}^{p,j}, \infty) \in \not\approx\}$ `// Lemma 3.1`

60 seconds and a memory limit of 4 gigabytes. The code, instructions for reproducing and raw results are available at: https://doi.org/10.5281/zenodo.14900915.

Overall Results. The number of solved instances is reported in Table 1, showing that the use of projective delineability does not greatly affect which problems are tractable within the time limit. The actual running times are depicted in Figure 6a: they show similar performance of the modified and baseline versions on the majority of instances, but significantly different behaviour on some instances. The differences largely even out over the whole benchmark set (a typical picture for changes to the projection heuristics in our experience). Nevertheless, it is clear that many instances do benefit from projective delineability. Identifying a criterion to predict whether the optimization pays off *a priori* is desirable (a machine learning based approach may be possible [13]), but that is not in the scope of this paper.

In the remainder of this section we compare the behaviour of BC and BC-PD variants to better understand these results.

Number of Applications. The results may suggest that the optimization from projective delineability is only applied in very rare cases, but this is not the case. Considering the instances solved by BC-PD: in total, the leading coefficient can be omitted for 307 822 polynomials, while for 826 795 polynomials, the optimization cannot be applied as the cell is unbounded in some direction, and for 4 089 polynomials, the optimization cannot be applied as it does not have a root on both sides of the bounds, or we did not find an appropriate resultant that may replace the leading coefficient. The optimization is thus applied in a substantial 37% of the cases. However, we also need to add a coefficient for each polynomial to ensure its non-nullification: if we add the leading coefficient, this suffices in most of the cases. Still, if it is omitted, we may choose another coefficient which is potentially simpler (e.g. of lower degree, fewer variables, etc) that the leading coefficient. We thus investigate whether this choice has an impact on another metric.

Quality of Cells. A good indicator for the quality of the generated cells is their size, which may be indirectly measured by the number of cells constructed (Figure 6b). By leaving out leading coefficients, we also hope to decrease the number of projection polynomials (Figure 6c). By choosing "simpler" coefficients than the leading coefficients (in terms of a lower degree), we aim to reduce the degree of the computed polynomials (not depicted in Figure 6 as there is no visible difference between the variants) and indirectly the number of computed roots (Figure 6d). However, all these measures do not seem to differ significantly in aggregate between the baseline and modified variants. An explanation could be that the (degree of the) leading coefficients do not carry much weight, or that the alternative coefficients are not much simpler.

Role of Projection Polynomials. To address the latter hypothesis, we compare the impact of the different projection polynomials (resultants, discriminants, and (leading) coefficients). On the 10 157 instances solved by BC-PD, of the time spent on algebraic computations, 55% is spent on computing discriminants and 5% is spent on computing resultants, and almost no time is spent on computing coefficients. Regarding the polynomials with a maximum total degree of the ones occurring in an

Table 1
Number of instances solved by BC resp. LDB.

	BC	LDB
solved by no variant	1977	1978
solved by BC/LDB but not by BC-PD/LDB-PB	20	16
solved by BC-PD/LDB-PB but not by BC/LDB	16	18
solved by both variants	10 141	10 142

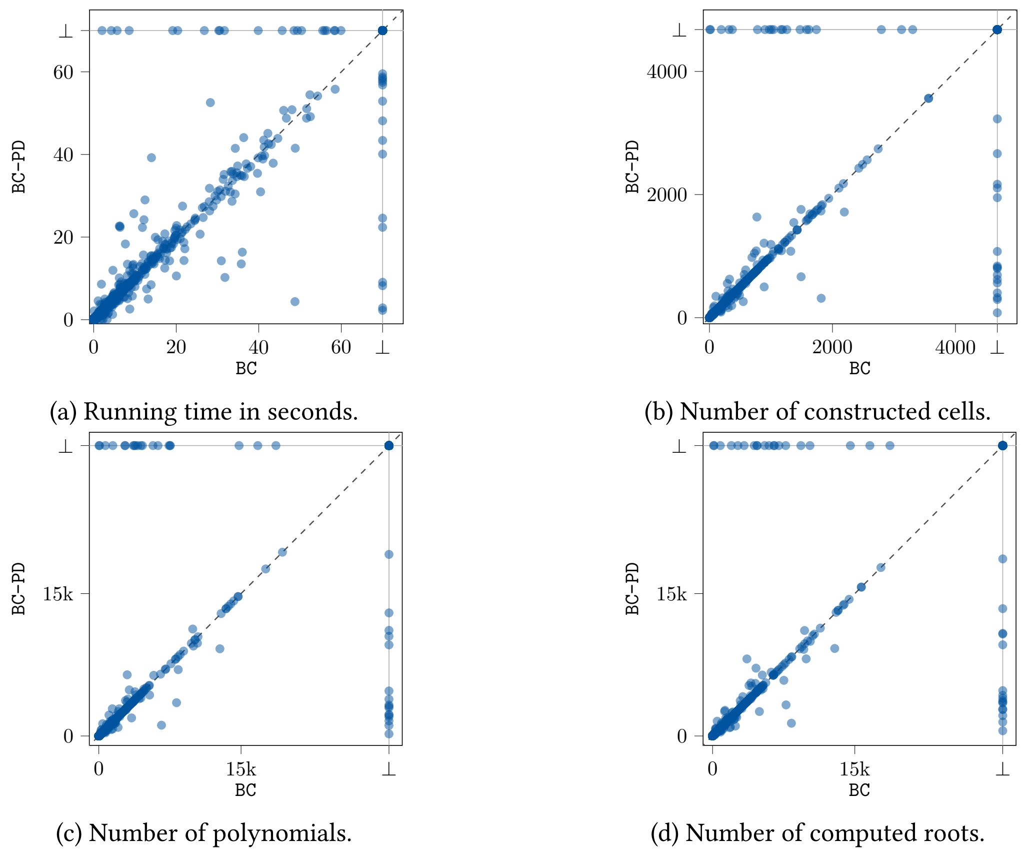


Figure 6: Comparison of different metrics. Each point indicates an instance. ⊥ indicates a timeout of the corresponding solver.

instance, in 15% of the instances, that polynomial is a discriminant, in 30% of the instances, that polynomial is a resultant, and in only 3% of the instances, that polynomial is a coefficient. Both measures show that the coefficients have a minor impact on the complexity of the projection.

6. Conclusion

We provided a first application for the notion of projective delineability recently introduced in [19], that formally describes the role of the leading coefficients in the projection. We modified the single cell construction algorithm accordingly and evaluated the result in the context of the NLSAT algorithm for SMT solving. The results offer a variety of possibilities for modifying the projection, and thus fit nicely as an extension of the proof system introduced in [21].

Our experimental evaluation shows the resulting optimization is applied in many cases, however, these do not translate to significant improvements in terms of running times or quality of intermediate results, as other projection polynomials play a greater role for the computational effort. That is the case when viewing the dataset as a whole: we have found many individual instances that benefit from the optimisation bringing further the future research question of how to recognise this in advance. Also in future work, further symbiotic optimizations may lead to practical improvements, such as reducing the amount of coefficients required for maintaining non-nullification of polynomials.

Declaration on Generative AI

No generative AI was used.

References

[1] Erika Ábrahám et al. “Deciding the Consistency of Non-Linear Real Arithmetic Constraints with a Conflict Driven Search Using Cylindrical Algebraic Coverings”. In: *Journal of Logical and Algebraic Methods in Programming* 119 (Feb. 2021).

[2] Haniel Barbosa et al. “Cvc5: A Versatile and Industrial-Strength SMT Solver”. In: *Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2022)*. 2022.

[3] Saugata Basu, Richard Pollack, and Marie-Françoise Roy. *Algorithms in Real Algebraic Geometry*. 2nd ed. Algorithms and Computation in Mathematics volume 10. Berlin: Springer, 2006.

[4] Christopher W. Brown. “Improved Projection for Cylindrical Algebraic Decomposition”. In: *Journal of Symbolic Computation* 32.5 (Nov. 2001).

[5] Christopher W. Brown. “QEPCAD B: A Program for Computing with Semi-Algebraic Sets Using CADs”. In: *ACM SIGSAM Bulletin* 37.4 (Dec. 2003).

[6] Christopher W. Brown. “Constructing a Single Open Cell in a Cylindrical Algebraic Decomposition”. In: *Symbolic and Algebraic Computation (ISSAC 2013)*. 2013.

[7] Christopher W. Brown. “Open Non-uniform Cylindrical Algebraic Decompositions”. In: *Symbolic and Algebraic Computation (ISSAC 2015)*. June 2015.

[8] Christopher W. Brown and Marek Košta. “Constructing a Single Cell in Cylindrical Algebraic Decomposition”. In: *Journal of Symbolic Computation* 70 (Sept. 2015).

[9] Changbo Chen et al. “Computing Cylindrical Algebraic Decomposition via Triangular Decomposition”. In: *Symbolic and Algebraic Computation (ISSAC 2009)*. July 2009.

[10] George E. Collins. “Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition”. In: *Automata Theory and Formal Languages*. 1975.

[11] Florian Corzilius et al. “SMT-RAT: An Open Source C++ Toolbox for Strategic and Parallel SMT Solving”. In: *Theory and Applications of Satisfiability Testing (SAT 2015)*. 2015.

[12] Leonardo de Moura and Nikolaj Bjørner. “Z3: An Efficient SMT Solver”. In: *Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2008)*. 2008.

[13] Tereso del Río and Matthew England. “Lessons on Datasets and Paradigms in Machine Learning for Symbolic Computation: A Case Study on CAD”. In: *Mathematics in Computer Science* 18.3 (Oct. 2024).

[14] Bruno Dutertre. “Yices 2.2”. In: *Computer Aided Verification (CAV 2014)*. 2014.

[15] Dejan Jovanović and Leonardo de Moura. “Solving Non-linear Arithmetic”. In: *Automated Reasoning (IJCAR 2012)*. 2012.

[16] Steven G. Krantz and Harold R. Parks. *A Primer of Real Analytic Functions*. Boston, MA: Birkhäuser, 2002.

[17] Scott McCallum. “An Improved Projection Operation for Cylindrical Algebraic Decomposition”. PhD thesis. University of Wisconsin-Madison, 1985.

[18] Scott McCallum. “An Improved Projection Operation for Cylindrical Algebraic Decomposition”. In: *Quantifier Elimination and Cylindrical Algebraic Decomposition*. 1998.

[19] Lucas Michel et al. “On Projective Delineability”. In: *Symbolic and Numeric Algorithms for Scientific Computing (SYNASC 2024)*. Sept. 2024.

[20] Jasper Nalbach and Erika Ábrahám. “Merging Adjacent Cells During Single Cell Construction”. In: *Computer Algebra in Scientific Computing (CASC 2024)*. 2024.

[21] Jasper Nalbach et al. “Levelwise Construction of a Single Cylindrical Algebraic Cell”. In: *Journal of Symbolic Computation* 123 (July 2024).

[22] Mathias Preiner et al. *SMT-LIB Release 2024 (Non-Incremental Benchmarks).* 2024.

[23] Andreas Seidl and Thomas Sturm. “A Generic Projection Operator for Partial Cylindrical Algebraic Decomposition”. In: *Symbolic and Algebraic Computation (ISSAC 2003).* Aug. 2003.

[24] Adam Strzeboński. “Cylindrical Algebraic Decomposition Using Local Projections”. In: *Symbolic and Algebraic Computation (ISSAC 2014).* July 2014.