# Levelwise construction of a single cylindrical algebraic cell 

Jasper Nalbach ${ }^{\text {a,1 }}$, Erika Ábrahám ${ }^{\text {a }}$, Philippe Specht ${ }^{\text {a }}$, Christopher W. Brown ${ }^{\text {b }}$, James H. Davenport ${ }^{\text {c }}$, Matthew England ${ }^{\text {d }}$<br>${ }^{a}$ RWTH Aachen University, 52056 Aachen, Germany<br>${ }^{b}$ United States Naval Academy, 597 McNair Road, Annapolis, MD 21402-5002, United States<br>${ }^{c}$ University of Bath, Claverton Down, Bath BA2 7AY, United Kingdom<br>${ }^{d}$ Coventry University, Coventry CV1 2TL, United Kingdom


#### Abstract

Satisfiability Modulo Theories (SMT) solvers check the satisfiability of quantifier-free first-order logic formulas over different theories. We consider the theory of non-linear real arithmetic where the formulae are logical combinations of polynomial constraints. Here a commonly used tool is the Cylindrical Algebraic Decomposition (CAD) to decompose real space into cells where the constraints are truth-invariant through the use of projection polynomials.

A CAD encodes more information than necessary for checking satisfiability. One approach to address this is to repackage the CAD theory into a search-based algorithm: one that guesses sample points to satisfy the formula, and generalizes guesses that conflict constraints to cylindrical cells around samples which are avoided in the continuing search. Such an approach can lead to a satisfying assignment more quickly, or conclude unsatisfiability with far fewer cells. A notable example of this approach is Jovanović and de Moura's NLSAT algorithm. Since these cells are being produced locally to a sample there is scope to use fewer projection polynomials than the traditional CAD projection. The original NLSAT algorithm reduced the set a little; while Brown's single cell construction reduced it much further still. However, that refines a cell polynomial-by-polynomial, meaning the shape and size of the cell produced depends on the order in which the polynomials are considered.

The present paper proposes a method to construct such cells levelwise, i.e. built level-by-level according to a variable ordering instead of polynomial-by-polynomial for all levels. We still use a reduced number of projection polynomials, but can now consider a variety of different reductions and use heuristics to select the projection polynomials in order to optimise the shape of the cell under construction. The new method can thus improve the performance of the NLSAT algorithm. We formulate all the necessary theory that underpins the algorithm as a proof system: while not a common presentation for work in this field, it is valuable in allowing an elegant decoupling of heuristic decisions from the main algorithm and its proof of correctness. We expect the symbolic computation community may find uses for it in other areas too.

The work has been implemented in the SMT-RAT solver and the benefits of the levelwise construction are validated experimentally on the SMT-LIB benchmark library. We also compare several heuristics for the construction and observe that each heuristic has strengths offering potential for further exploitation of the new approach.


Keywords: Satisfiability modulo theories, Cylindrical algebraic decomposition, Non-linear real arithmetic, Model-constructing satisfiability calculus, Proof system

## 1. Introduction

In this paper we present a new method to construct around a sample point a single cylindrical cell that is truth-invariant for a set of polynomial constraints. We demonstrate how the new method allows for improved decision procedures to determine the satisfiability of formulae involving such constraints. We use a proof system presentation for our method, which we consider an important contribution for such algebraic decision procedures. We take this opportunity to explain the benefit of such a presentation to the symbolic computation community. This introduction continues with a broad overview of context for the contribution, followed by the plan of the paper.

[^0]
### 1.1. Non-linear real arithmetic and $C A D$

We are concerned with non-linear real arithmetic, formulae which are Boolean combinations of polynomial constraints with rational coefficients. This is a powerful logic that can express a wide variety of problems. This logic admits quantifier elimination [40], i.e. any quantified formula in the logic may be replaced by an equivalent quantifier-free one. In this paper we restrict our attention to the problem of determining the satisfiability of such formulae, that is quantifier elimination when all variables are existentially quantified.

The most commonly used complete methods here are based on the idea of the Cylindrical Algebraic Decomposition (CAD) introduced by Collins [16]. A CAD is a finite decomposition of $\mathbb{R}^{n}$ into cells, traditionally produced relative to a set of polynomials in $n$ variables such that each polynomial has constant sign on each cell. It thus allows us to use a finite set of sample points (one for each cell) to study sign-constraints on those polynomials over the infinite space $\mathbb{R}^{n}$. The CAD method offered the first tractable approach to real quantifier elimination and found numerous applications in the years that followed. However, its practical use is restricted by a doubly exponential worst case complexity in the number of variables [20], that is felt often in practice: the algorithm makes use of iterated resultants [16] leading to polynomials of doubly-exponential degree.

It was soon realised that a CAD encoded far more information than needed even for quantifier elimination: a CAD for a set of polynomials can be used to study any logical formula built from those polynomials, not just the one of interest. Some progress was since made in adapting the core CAD algorithm to the logical formula, e.g. [17, 23], but these were only partial solutions.

### 1.2. NLSAT, MCSAT and single cylindrical cells

A novel framework for satisfiability modulo theories (SMT) solving was introduced with the NLSAT algorithm of Jovanović and de Moura [27] in 2012. This was since generalised into the model constructing satisfiability calculus (MCSAT) framework [21] and has been applied to other logics such as non-linear integer arithmetic [25]. In MCSAT the search at the Boolean and theory levels are carried out concurrently, mutually guided by each other away from unsatisfiable regions. Partial solution candidates for the Boolean structure and for the corresponding theory constraints are constructed incrementally in parallel, with Boolean conflicts generalised using propositional resolution and, for real algebra, theory conflicts by CAD technology.

For the latter, when a theory model (sample point) is determined not to satisfy all those constraints which should hold according to the current Boolean search, then we seek to guide the future search by an explanation to which is generalised from the sample point to a region containing the point on which the same constraints fail for the same reasons. This can be achieved by having the polynomials involved in the combination of constraints which cause the failure all have invariant sign upon this region. Such regions are constructed as cylindrical algebraic cells, but they are not necessarily cells from the CAD that would be built for the problem. Instead, they are usually larger since not all constraints are involved in every conflict. The exclusion of the cell is learned by adding a new clause: the negation of the algebraic description of the cell.

This motivates the optimisation of sub-algorithms to produce single cells from a point and a set of polynomial constraints. Savings can be made not just by building cells with only a subset of the constraints, but also by restricting the combinations of those constraints that we do consider in reference to the current model. Until now, the state-of-the-art approach is that in [11]. We continue this research in the present paper by developing a new method for single cell construction.

### 1.3. First contribution: Proof system presentation

In contrast to [11], our new method allows for different choices how to construct the cell. We will describe and evaluate some of these choices, however, it is important to distinguish that area of work from the broader method to build the cell introduced in the next subsection. The choices do not effect the correctness of the cell produced: in all cases the cell meets the essential criteria of containing the sample and being invariant for the truth of the conflicting constraints. Nor do the choices have an affect on high level measures of complexity: they achieve similar reductions in algebraic work. However, it can be observed that the choices do greatly effect the cells produced and thus the performance of the algorithm and so it is worth to try to make an optimal choice. We do these choices heuristically, i.e. using methods not guaranteed to give an optimal answer but hopefully giving a reasonable answer quickly. To expedite and simplify future research on heuristics it is helpful to clearly separate out these heuristic choices from the broader algorithm and its proof of correctness.

To achieve that, we present our work as a proof system. Such a presentation clearly achieves the separation of heuristic decisions from action that need to take place for the correctness of the method. Essentially, we must find a chain of proof rules to prove our desired property, and where there is freedom in the chain to be built we can employ a heuristic method. The system is flexible, extensible, allows for detailed optimizations without changing
the fundamental algorithm, and allows for correctness proofs to be portioned nicely. We note that this is not just a presentation for the purpose of the paper, but also present in the underlying implementation we report on.

We acknowledge that a proof system presentation is uncommon in symbolic computation. It is more prevalent in the SAT and SMT communities where there is more intense work on the optimisation of such heuristic choices and proof systems are an established presentation method. However, such a system has not been used before for CAD theory, even when deployed in the SMT context. We view our proof system presentation as a contribution in its own right, which allowed for greater exploration of heuristic choices in our work. We also hope the wider symbolic computation community may find it interesting as a potential new tool to use elsewhere.

### 1.4. Second contribution: Levelwise single cells

The current state-of-the-art for single cylindrical cell construction is that of Brown and Košta [11]. This constructs the single cell gradually, processing one polynomial at a time, initialising the cell as the entirety of $\mathbb{R}^{n}$ and then gradually refining it according to the sign of each polynomial considered. For each refinement the method needs to consider only the interaction of the next polynomial with the ones currently defining the cell, rather than all those that went before, which allows for savings compared to the original approach used in [27]. However, this approach introduces a sensitivity to the order in which polynomials are considered. A machine learning approach to select the order was considered in [10].

The alternative method contributed in this paper allows to produce the single cell incrementally by level (i.e. dimension / variable). This removed the direct sensitivity to the polynomial ordering, instead introducing at each level decisions about which polynomials to use first. This approach uncovers a greater range of decision that the polynomial ordering, and allows for more reasoned heuristics than black-box machine learning. By allowing for these better heuristic decisions we can produce more optimal cells in turn improving the performance of algorithms which use them.

### 1.5. Plan of the paper

We continue in Section 2 by introducing the necessary preliminaries and notation used, followed by background material on CAD. Then in Section 3 we present the existing state of the art in single cell construction and an informal motivation for our new levelwise approach to single cell construction.

In Section 4, a new CAD property (projective delineability) and the theory around it is presented: we need this to prove the correctness of optimisations utilised by our new algorithm. Then in Section 5 we establish the proof system, and in Section 6 we present our new algorithm and some heuristics that may be used with it.

In Section 7 we give some qualitative analysis on our new method and the heuristics. Then an experimental evaluation on the use of the new method for explanation generation in MCSAT is given in Section 8. Finally, we conclude in Section 9 with an outlook on further research and open questions.

## 2. Preliminaries

Let $\mathbb{N}$ denote the set of all natural numbers including $0, \mathbb{N}_{>0}=\mathbb{N} \backslash\{0\}, \mathbb{Q}$ be the rational numbers, and $\mathbb{R}$ be the real numbers. For $i, j \in \mathbb{N}$ with $i<j$, we define the sets of integers $[i . . j]=\{i, \ldots, j\}$ and $[i]=[0 . . i]$. 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 $\left(r_{1}, \ldots, r_{j}\right)$. For a tuple $t=(a, b, c, \ldots)$ we denote by $t . a, t . b, t . c, \ldots$ the corresponding tuple entries.

Let $f: D \rightarrow E$ be a function, then the domain $D$ of $f$ is denoted by $\operatorname{dom}(f)$, and the restriction of $f$ to $A \subseteq D$ is denoted by $\left.f\right|_{A}$ (i.e. $\operatorname{dom}\left(\left.f\right|_{A}\right)=A$ ). Let $f, g: D \rightarrow E$ and let $<$ be a total order on $E$. We write $f<g$ if $f(d)<g(d)$ for all $d \in D$ and $f \leq g$ if $f(d) \leq g(d)$ for all $d \in D$.

### 2.1. Variables and polynomials

We assume that the reader is familiar with the common definitions and terminology related to polynomials. We will introduce some notation in this section; for further reading, we refer to [19].

We work with the variables $x_{1}, \ldots, x_{n}$ 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 $\mathbb{Q}[y]$ to denote univariate polynomials in some variable $y$ and $\mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$ for multivariate polynomials in those variables. We say that a polynomial $p$ is of level $j$ (denoted as $\operatorname{level}(p)=j$ ) if $x_{j}$ is the largest variable appearing in $p$ : i.e. either $j=0$ and $p \in \mathbb{Q}$; or $j \in[1 . . n]$ and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{j}\right] \backslash \mathbb{Q}\left[x_{1}, \ldots, x_{j-1}\right]$.

Assume in the following some $i \in[n]$ and polynomials $p, q \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$.

We write $p\left(x_{1}, \ldots, x_{i}\right)$ to indicate $p$ 's variable domain. For $j \in[1 . . i]$ and $r=\left(r_{1}, \ldots, r_{j}\right) \in \mathbb{R}^{j}$ we write $p\left(r, x_{j+1}, \ldots, x_{i}\right)$ for the polynomial $p$ after substituting $r_{1}, \ldots, r_{j}$ for $x_{1}, \ldots, x_{j}$ in $p$ and indicating the remaining free variables in $p$.

We use realRoots $(p) \subseteq \mathbb{R}^{i}$ to denote the set of real roots of $p, \operatorname{deg}_{x_{j}}(p)$ to denote the degree of $p$ in $x_{j}$, $\operatorname{ldcf}_{x_{j}}(p)$ the leading coefficient of $p$ in $x_{j}$, factors $(p)$ to denote the irreducible factors of $p, \operatorname{disc}_{x_{j}}(p)$ to denote the discriminant of $p$ with respect to $x_{j}$, and $\operatorname{res}_{x_{j}}(p, q)$ to denote the resultant of $p$ and $q$ with respect to $x_{j}$.

### 2.2. Real algebraic numbers, constraints and cells

Real algebraic numbers are real roots of univariate polynomials with rational coefficients. Although we will not distinguish between real and real algebraic numbers in the following for simplicity, the algorithms are complete when restricting all choices of constants to real algebraic numbers.

A constraint $p \sim 0$ compares a polynomial $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$, level $(p)=i$ to zero using a relation symbol, $\sim \in\{=, \neq,<,>, \leq, \geq\}$, and has the solution set $\left\{r \in \mathbb{R}^{i} \mid p(r) \sim 0\right\}$.

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 is called algebraic if it is a semi-algebraic set.

For simplifying the notation throughout this paper, we define $\mathbb{R}^{0}=\{()\}$. Given $i, j \in \mathbb{N}_{>0}$ with $j<i$, we call $\mathbb{R}^{i}$ an extension of $\mathbb{R}^{j}$ and define the projection of a set $R \subseteq \mathbb{R}^{i}$ onto $\mathbb{R}^{j}$ by $R \downarrow_{[j]}=\left\{\left(r_{1}, \ldots, r_{j}\right) \mid\left(r_{1}, \ldots, r_{i}\right) \in R\right\}$.

Given a cell $R \subseteq \mathbb{R}^{i}, i \in[1 . . n]$ and continuous functions $f, g: R \rightarrow \mathbb{R}$, we define the sets $R \times(f, g)=$ $\left\{\left(r, r_{i+1}\right) \mid r \in R, r_{i+1} \in(f(r), g(r))\right\}$, analogously $R \times(-\infty, g)=\left\{\left(r, r_{i+1}\right) \mid r \in R, r_{i+1}<g(r)\right\}, R \times(f, \infty)=$ $\left\{\left(r, r_{i+1}\right) \mid r \in R, f(r)<r_{i+1}\right\}$, and $R \times f=\left\{\left(r, r_{i+1}\right) \mid r \in R, r_{i+1}=f(r)\right\}$. Note that if $f \leq g$ (on $R$ ), then these sets are cells (as a continuous image of a connected set is connected).

The sign of $r \in \mathbb{R}$, denoted $\operatorname{sgn}(r)$, is defined to be 1 if $r>0,-1$ if $r<0$, and 0 otherwise. A polynomial $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$ is sign-invariant on a set $R \subseteq \mathbb{R}^{i}$ if $\operatorname{sgn}(p(r))=\operatorname{sgn}\left(p\left(r^{\prime}\right)\right)$ for all $r, r^{\prime} \in R$. A set of polynomials $P \subseteq \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$ is sign-invariant on $R \subseteq \mathbb{R}^{i}$ if all $p \in P$ are sign-invariant on $R$.

## 2.3. $C A D$ definition

A set $D=\left\{R_{1}, \ldots, R_{k}\right\}$ of cells of $\mathbb{R}^{n}$ such that $\cup_{i=1, \ldots, k} R_{i}=\mathbb{R}^{n}$ and $R_{i} \cap R_{j}=\emptyset$ is called a decomposition of $\mathbb{R}^{n}$. A decomposition is called algebraic if its cells are algebraic. A decomposition $D$ of $\mathbb{R}^{n}$ is called cylindrical over a decomposition $D^{\prime}$ of $\mathbb{R}^{m}, m<n$ if all projections of cells $R \in D$ onto $\mathbb{R}^{m}$ are themselves cells in $D^{\prime}$. I.e. the cells in $\mathbb{R}^{n}$ stack up in cylinders over the cells in $\mathbb{R}^{m}$. A cylindrical algebraic decomposition (CAD) is produced relative to a variable ordering: it is an algebraic decomposition $D$ such that there exists a sequence of algebraic decompositions $\left(D_{1}, \ldots, D_{n}\right), D=D_{n}$ with each $D_{i}$ a cylindrical decomposition of $\mathbb{R}^{i}$ over $D_{i-1}$ for $i \in[2 . . n]$.

A decomposition inherits the invariance properties for polynomials and constraints defined above if they apply to all its cells. A CAD will usually be computed relative to an input polynomial set to ensure such an invariance property. Collins [16] first introduced the notion of a CAD and an algorithm for computing a sign-invariant CAD in the 1970s. A central notion for this algorithm is delineability.

Definition 2.1 (Delineability [16]). Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$ be a cell, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right] \backslash\{0\}$. The polynomial $p$ is called delineable on $R$ if and only if there exist finitely many continuous functions $\theta_{1}, \ldots, \theta_{k}: R \rightarrow \mathbb{R}$ (for $k \geq 0$ ) such that

- $\theta_{1}<\ldots<\theta_{k}$;
- the set of real roots of the univariate polynomial $p\left(r, x_{i+1}\right)$ is $\left\{\theta_{1}(r), \ldots, \theta_{k}(r)\right\}$ 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 $i \in[1 . . k]$, the multiplicity of the root $\theta_{i}(r)$ of $p\left(r, x_{i}\right)$ is $m_{i}$.
The $\theta_{j}$ are called real root functions of $p$ on $R$. The cells $R \times \theta_{j}, j \in[1 . . k]$ are called $p$-sections over $R$. The cells $R \times\left(-\infty, \theta_{1}\right), R \times\left(\theta_{k}, \infty\right)$, and $R \times\left(\theta_{j}, \theta_{j+1}\right)$ for $j \in[1 . . k-1]$, and in case $k=0$ also $R \times(-\infty, \infty)$, are called $p$-sectors over $R$.

These notions are extended to finite sets of polynomials $P \subseteq \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right] \backslash\{0\}$ such that $P$ is delineable on $R$ if the product of the polynomials in $P$ is delineable on $R$. Accordingly, we define the real root functions of $P$, the $P$-sections and $P$-sectors; for the empty polynomial set there are no sections and a single sector $\mathbb{R}^{i}$.

A $C A D$ projection operator is a function proj that maps a set of polynomials $P$ to a set of lower-level polynomials such that the sign-invariance of $\operatorname{proj}(P)$ on $R$ implies the delineability of $P$ on $R$. The operator proj induces a sign-invariant $C A D$ of $P$, recursively defined as follows: In case all polynomials in $P$ are of level 1, then the CAD is the set of all sections and sectors of $P$. If the polynomials are of higher level, the CAD contains all sections and sectors of $P$ over each cell of the CAD of $\operatorname{proj}(P)$.

### 2.4. McCallum's projection operator

Although Collin's original projection operator is complete, the projection set is large and thus relatively inefficient. McCallum presented an improved operator by making the projection set smaller [33]. To do so, his proof of correctness relies not on sign-invariance but on the stronger property of order-invariance. Although a stronger property, the induced cells in McCallum's CAD are actually bigger than in Collin's CAD. In the following, we present a simplified version of the McCallum CAD projection (simplified as we do not describe the optimisation using delineating polynomials).

McCallum's theory relies on some notions which we will mention here only on the level of intuition: for more details, we refer to [31, 33]. An $i$-dimensional (analytic) submanifold of $\mathbb{R}^{n}$ is a non-empty subset $R \subseteq \mathbb{R}^{n}$ that "looks locally like $\mathbb{R}^{i}$ ". Given an open subset $U \subseteq \mathbb{R}^{i}$, a function $f: U \rightarrow \mathbb{R}$ is called analytic if it has a multiple power series representation [31] around each point of $U$. Given an $i$-dimensional submanifold $R$ of $\mathbb{R}^{n}$, a function $f: R \rightarrow \mathbb{R}$ is called analytic if for all $r \in R, R$ looks locally like $\mathbb{R}^{i}$ with respect to a coordinate system about $r$ and $f$ looks locally like an analytic function $\mathbb{R}^{i} \rightarrow \mathbb{R}$. Every open subset of $\mathbb{R}^{i}, i \in[1 . . n]$ is an analytic submanifold. To simplify notation, we say $\mathbb{R}^{0}$ is an analytic submanifold. Given an analytic submanifold $R \subseteq \mathbb{R}^{i}$ and analytic functions $f, g: R \rightarrow \mathbb{R}$ with $f<g$, the sets $R \times(f, g)$ and $R \times f$ are analytic submanifolds as well [31, Theorem 2.2.3. and Theorem 2.2.4]. Note that analytic submanifolds are cells.

Let $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{n}\right]$ be a polynomial and $r \in \mathbb{R}^{n}$ be a point. Then the order of $p$ at $r$ is defined as

$$
\operatorname{ord}_{r}(p)=\min (\{k \in \mathbb{N} \mid \text { some partial derivative of total order } k \text { of } p \text { does not vanish at } r\} \cup\{\infty\})
$$

We call $p$ order-invariant on $R \subseteq \mathbb{R}^{n}$ if $\operatorname{ord}_{r}(p)=\operatorname{ord}_{r^{\prime}}(p)$ for all $r, r^{\prime} \in R$. Note that if $p$ has no root in $R$, then $\operatorname{ord}_{r}(p)=0$ for all $r \in R$ and $p$ is trivially order-invariant on $R$. Note also that order-invariance implies sign-invariance.

Additionally, the notion of delineability is extended to analytic delineability, which is only defined on connected analytic submanifolds and the real root functions are required to be analytic.

McCallum's operator requires a smaller projection set than Collins' operator. It does so by maintaining the stronger property of order-invariance instead of sign-invariance; however, order-invariance can only be concluded if no polynomial is nullified on a point in the underlying cell.

Definition 2.2 (Nullification). Let $i \in \mathbb{N}, r \in \mathbb{R}^{i}$, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, $\operatorname{level}(p)=i+1$. The polynomial $p$ is nullified on $r$ if $p\left(r, x_{i+1}\right)=0$.

The McCallum projection operator is thus incomplete. We note the recent validation of Lazard projection in [36] as an alternative to McCallum projection which is complete and is no bigger than McCallum except for those cases where McCallum cannot be applied [12]. We choose to formalise here with McCallum projection as it is extended to optimisations such as equational constraints [34] [23] which are very powerful in practice (see Section 5.6) and the existing single cell approach [11] is also based on McCallum. We expect that our work could be reformulated in Lazard projection if so desired.

### 2.5. Computing a $C A D$

In many applications, a decomposition of a set of polynomials $P$ is not computed explicitly but instead the projection of $P$ is used to generate sample points for every sign-invariant cell that would be formed in such a decomposition. The generation of cells / samples from the projection is called lifting. Polynomials can be evaluated at these sample points to check the satisfiability of constraints and formulae which involve them.

For representing cells explicitly, we need to give witnesses for the real root functions $\theta_{j}: R \rightarrow \mathbb{R}$ from Definition 2.1 defining the sectors and sections over a cell $R$.

Definition 2.3 (Indexed root expression). Let $i \in \mathbb{N}, p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, $\operatorname{level}(p)=i+1$, and $j \in \mathbb{N}_{>0}$. The indexed root expression $\operatorname{root}_{x_{i+1}}[p, j]: \mathbb{R}^{i} \rightarrow \mathbb{R} \cup\left\{\right.$ undef\} is the $j$-th real root of $p$ in $x_{i+1}$ at the given sample if it exists, and undef otherwise. That is for each $s \in \mathbb{R}^{i}$ :

$$
\operatorname{root}_{x_{i+1}}[p, j](s)= \begin{cases}\text { undef } & \text { if } j>\left|\operatorname{realRoots}\left(p\left(s, x_{i+1}\right)\right)\right| \text { or } p\left(s, x_{i+1}\right)=0, \text { and otherwise } \\ \xi_{j} & \text { where } \operatorname{realRoots}\left(p\left(s, x_{i+1}\right)\right)=\left\{\xi_{1}, \ldots, \xi_{k}\right\} \text { and } \xi_{1}<\ldots<\xi_{k}\end{cases}
$$

Indexed root expressions of this form are also called indexed root expression of level $i+1$. Assuming $\xi$ denotes the above indexed root expression, we use $\xi$.p to refer to $p$ and $\xi \cdot j$ to refer to $j$.

In the algorithms presented in this paper, we only need to evaluate the indexed root expressions for real algebraic numbers. Note that the existence and index of a real root function depends on the given sample. Thus, the same indexed root expression may refer to different real root functions at different sample points.

Definition 2.4 (Symbolic intervals, single cell and CAD data structures). A symbolic interval I of level $i$ is either (i) of the form $\mathrm{I}=($ sector, $l, u)$ where $l$ is either an indexed root expression of level $i$ or $-\infty$, and $u$ is either an indexed root expression of level $i$ or $\infty$; or (ii) of the form $\mathrm{I}_{i}=($ section, $b$ ) where $b$ is an indexed root expression of level $i$. The polynomials I.l, I.u respectively I.b are the defining polynomials of I (if they exist).

A single cylindrical cell data structure is a sequence $R=\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{n}\right)$ of symbolic intervals $\mathrm{I}_{i}$ of level $i$. For an empty single cell data structure, we define setOf( ()$)=\{()\}$. For a single cell data structure $\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i}\right)$, $i \geq 1$, we define the corresponding subset of $\mathbb{R}^{i}$ as $\operatorname{set} O f\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i}\right)=\left\{\left(r, r^{\prime}\right) \mid r \in \operatorname{set} O f\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i-1}\right), r^{\prime} \in\left(\mathrm{I}_{i} . l(r), \mathrm{I}_{i} . u(r)\right)\right\}$, respectively $\operatorname{set} O f\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i}\right)=\left\{\left(r, r^{\prime}\right) \mid r \in \operatorname{set} O f\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i-1}\right), r^{\prime}=\mathrm{I}_{i} . b(r)\right\}$ if the respective indexed root expressions are not undef, and set $O f\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i}\right)=$ undef otherwise. Similarly, we define $\operatorname{set} O f\left(R, \mathrm{I}_{i}\right)$ for arbitrary subsets $R \subseteq \mathbb{R}^{i-1}$ and $\operatorname{set} O f\left(r, I_{i}\right)$ for points $r \in \mathbb{R}^{i-1}$.
$A \mathrm{CAD}$ data structure $D$ is a set of single cell data structures. We define $\operatorname{set} O f(D)=\{\operatorname{set} O f(R) \mid R \in D\}$.
In our algorithms, indexed root expressions occurring in a single cell data structure will be always defined; the above definition covers the undef case just for the sake of completeness. Furthermore, note that given a single cell data structure $R=\left(I_{1}, \ldots, I_{i}\right)$, the restrictions of members $\left.I_{i} \cdot l\right|_{\operatorname{set} O f\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i-1}\right)},\left.\mathrm{I}_{i} \cdot u\right|_{\operatorname{set} O f\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i-1}\right)}$, $\mathrm{I}_{i} .\left.b\right|_{\operatorname{set} O f\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i-1}\right)}$ are real root functions of their defining polynomials as $\theta_{j}$ in Definition 2.1.

Definition 2.5 (Indexed root expressions of polynomials). Let $i \in \mathbb{N}, s \in \mathbb{R}^{i}$, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$ of level $i+1$. The set of indexed root expressions of $p$ at $s$ is defined as

$$
\operatorname{irExpr}(p, s)= \begin{cases}\text { undef }^{2} & p\left(s, x_{i+1}\right)=0 \\ \left\{\operatorname{root}_{x_{i+1}}[p, j] \mid j \in\left[1 . .\left|\operatorname{realRoots}\left(p\left(s, x_{i+1}\right)\right)\right|\right]\right\} & \text { otherwise }\end{cases}
$$

Let $P \subseteq \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$ be a set of polynomials of level $i+1$. The set of indexed root expressions of $P$ at $s$ is defined as

$$
\operatorname{ir} \operatorname{Expr}(P, s)=\bigcup_{p \in P} \operatorname{irExpr}(p, s)
$$

Let $\xi \in \mathbb{R}$. The set of indexed root expressions of $P$ for $s$ and $\xi$ is defined as

$$
\operatorname{ir} \operatorname{Expr}(P, s, \xi)=\left\{\operatorname{root}_{x_{i+1}}[p, j] \mid p \in P, \operatorname{level}(p)=i+1, \xi=\operatorname{root}_{x_{i+1}}[p, j](s)\right\}
$$

A description of a sign-invariant CAD defined by a projection operator with respect to a set of polynomials can be computed as follows: First the projection operator is applied from level $n$ to 1 , called the projection phase. This is followed by the lifting phase where, starting from level 1 , all cells from level $i-1$ are extended to the level $i$ such that all sections and sectors in the cylinder above every cell of level $i-1$ are identified as cells. To achieve this, for each cell R of dimension $i-1$, the polynomials on level $i$ are partially evaluated up to their last dimension using a sample $s$ from R. This results in univariate polynomials whose roots can be isolated and sorted to give intervals: point intervals at the roots of the polynomials, the open intervals between them, and the two open intervals below and above all roots. Delineability allows the conclusions drawn at the sample to generalise over R. For each interval a symbolic description is determined that together extend R to level $i$.

## 3. Single cell computation

For our problem of study, we are not interested in computing a full sign-invariant CAD, but only a single signinvariant algebraic cell. That is, given a set of polynomials $P \subseteq \mathbb{Q}\left[x_{1}, \ldots, x_{n}\right]$ and a sample $s \in \mathbb{R}^{n}$, we need to compute a cell $R \subseteq \mathbb{R}^{n}$ such that $s \in R$ and $P$ is sign-invariant on $R$.

In this paper, we will focus on the computation of algebraic cells that adhere to Definition 2.4. These cells have a triangular algebraic description with respect to the variable ordering (i.e. a condition on $x_{1}$ alone; then one on $\left(x_{1}, x_{2}\right)$, and so on). Such cells are called (locally) cylindrical [11, 2] as this shape is implied by the cylindricity property of a full CAD. Although we do not construct entire decompositions in this paper, we make use of CAD theory which results in cells having this property. Such cells have the advantage of being easy to visualise and compute with. For example, projection with respect to the variable ordering is trivial, as is checking whether a point is inside such a cell, which is particularly important for our use of such cells.

### 3.1. Previous work on optimizing single cell computations

As described in Section 1.2, Jovanović and de Moura use a single cell construction for model explanation in their algorithm to decide satisfiability for non-linear arithmetic [27]. This was based on Collins' CAD projection operator [16] but did not compute a full CAD projection: it left out coefficients based on the current sample (as we only need consider subsequent coefficients when the leading coefficient vanishes).

Brown enhanced this single cell construction, first in the open case [7] (i.e. building only cells with maximal dimension) and then for the general case [11] in collaboration with Košta. This work was based on McCallum projection theory, and in addition to coefficients also avoided some computation of resultants and discriminants based on the current sample. The work led in turn to the consideration of entire decompositions built one cell at a time [8] and their use for quantifier elimination [9].

The approach in [8] started with a single cell data structure describing the whole of $\mathbb{R}^{n}$ which was continually refined by merging in new polynomials one at a time. The process maintains several invariants: (1) the cell contains the sample point $s ;(2)$ the cell is cylindrical; and (3) all polynomials merged so far have constant sign (more precisely, constant order) in the cell. The polynomials are merged one-by-one, and the order in which they are merged affects the cell that gets produced. Figure 1 illustrates this process for Example 3.1 below, showing different results for two different orderings of the polynomials. From now on we will refer to this approach as refinement-based single cell construction. The merge operation, unless on the lowest level, projects the current polynomial $p$ and then calls itself iteratively on the projection result. When the call returns to $p$, the polynomial is used to refine the bounds of the cell in the dimension that corresponds to the level of $p$. This is done by isolating the roots of the univariate polynomial resulting from partially evaluating $p$ up to its last dimension using the sample $s$. If there is a root closer to $s_{i}$ than the current bound then it becomes the new bound of the sector. In the special case that a root coincides with $s_{i}$, the sector collapses into a section described by said root. By this recursive refinement, the transitivity of the ordering on real root functions of polynomials induced by the order-invariance of resultants is exploited, so that at most two resultants are calculated per merge-operation and polynomial. Furthermore, as soon as a section is identified, some superfluous discriminants can be detected, and using the current sample irrelevant coefficients can be identified.

Example 3.1. We consider the point $s=\left(\frac{1}{8},-\frac{3}{4}\right)$ and polynomials $p_{1}=x_{1}-2 x_{2}+1, p_{2}=x_{1}^{2}+x_{2}^{2}-1$ and $p_{3}=x_{1}-2 x_{2}-1$. These are the polynomials studied in Figure 1 which demonstrates the refinement-based single cell construction process for two different orderings. We see that one produces a larger cell than the other. It is not obvious how to choose good orders for adding polynomials in this approach (a machine learning heuristic was presented in [10]). Moreover, because the method works by completely refining the cell by the chosen polynomial, the order in which lower-level polynomials resulting from projections are added is not very flexible.

Our approach will build a single cell levelwise, i.e. constructing the same level-by-level according to a variable ordering instead of polynomial-by-polynomial for all levels. We note there exists another levelwise single cell construction in an unpublished work on the arXiv, [30]. In comparison, our formulation using a proof system will enables the use of more sophisticated optimizations in the following.

### 3.2. Our levelwise approach to single cell computation

Our new levelwise approach is based on using information about the ordering of the roots of polynomials relative to the sample point $s$ and to one another to make decisions during the cell construction process that are likely to lead to larger cells.

We will exploit said information and the transitivity of the ordering on the roots induced by order-invariance of resultants. Thus, before projection, the roots of the polynomials on the current level are isolated and ordered to see which resultants are required for maintaining sign- or order-invariance of the given polynomials.

Example 3.2. Figure 2 shows how our new levelwise approach would operate on the formula from Example 3.1. We start by considering $x_{1}=s_{1}$, the first coordinate of the sample $s$. We evaluate the polynomials at this coordinate and isolate the real roots to find $\xi_{1}=\operatorname{root}_{x_{2}}\left[p_{2}, 1\right], \xi_{2}=\operatorname{root}_{x_{2}}\left[p_{3}, 1\right], \xi_{3}=\operatorname{root}_{x_{2}}\left[p_{1}, 1\right]$, and $\xi_{4}=\operatorname{root}_{x_{2}}\left[p_{2}, 2\right]$. We order these, along with the second coordinate of $s$, as in the top-left image.

Our sample lies in the interval for $x_{2}$ denoted $\mathrm{I}_{2}=\left(\right.$ sector, $\left.\xi_{1}, \xi_{2}\right)$ in the top-right image. We thus know that, local to $s_{1}$, the cell we want will be bounded from below by $p_{2}$ and from above by $p_{3}$. We also know that the other section of $p_{2}$ and the section of $p_{1}$ are above the cell. In this figure the graphs of the polynomials are greyed out because the algorithm is not aware of them: it knows only their roots when evaluated at the sample. However, this is enough to allow the algorithm to infer behaviour around the sample, as illustrated by the partial thick lines.

As we generalise from the sample we must ensure than $\xi_{1}$ and $\xi_{2}$ remain well-defined and that no root function crosses the symbolic interval described by $\mathrm{I}_{2}$. To achieve this we compute a projection consisting of $p_{4}=\operatorname{disc}_{x_{2}}\left(p_{2}\right)$


Figure 1: Refinement-based construction of a single cell for Example 3.1. In the upper row the original cell (the entire plane) is refined first by polynomial $p_{1}$, then by $p_{2}$, then by $p_{3}$. In the lower row a different order of refinement is used: $p_{3}, p_{2}, p_{1}$ giving a larger cell. Note that in the figures of this paper we label the varieties of polynomials with their name, $p_{1}$ labels $p_{1}=0$.
and $p_{5}=\operatorname{res}_{x_{2}}\left(p_{3}, p_{2}\right)$ (also $\operatorname{disc}_{x_{2}}\left(p_{1}\right)$ and $\operatorname{disc}_{x_{2}}\left(p_{3}\right)$ but they do not have any real roots and so play no further role in this example). Crucially, because $p_{1}$ has only one section and that section lies above the cell of interest, the resultant of $p_{1}$ and the lower-boundary polynomial $p_{2}$ is not required or computed. The bottom-left figure shows how the zeros of the projection map onto the geometric features.

Analogously to $x_{2}$, we isolate the real roots $\xi_{1}^{\prime}=\operatorname{root}_{x_{1}}\left[p_{4}, 1\right], \xi_{2}^{\prime}=\operatorname{root}_{x_{1}}\left[p_{5}, 1\right], \xi_{3}^{\prime}=\operatorname{root}_{x_{1}}\left[p_{4}, 2\right]$ in $x_{1}$. We determine the interval $\mathrm{I}_{1}=\left(\right.$ sector $\left., \xi_{2}^{\prime}, \xi_{3}^{\prime}\right)$ around $s_{1}$ for $x_{1}$ and thus generate the cell in the bottom-right image.

So we see that for Example 3.2 the new levelwise approach produces the second of the two possible outcomes from the refinement-based construction (Figure 1). The example illustrates how the levelwise approach avoids "mistakes" that the refinement-based construction can make due to poor choices of orderings for the polynomials.

The levelwise approach also allows greater flexibility in how polynomials resulting from projection are handled, but that aspect of the algorithm requires more variables than in this simple example in order to be observed.

## 4. Projective delineability

It is already known that if a polynomial is not nullified at any point of a cell, then a projection consisting solely of the leading coefficient and the discriminant suffices to guarantee its delineability. In this section, we will explore a new notion called projective delineability (Definition 4.1) for which the order-invariance of the discriminant alone is enough to guarantee. It is similar to delineability bu allows for the existence of an additional special root function below or above all other roots that has a singularity at points (exactly where the leading coefficients vanishes) where it approaches plus or minus infinity.

This new property is often enough for constructing a sign-invariant cell, as in the example is depicted in Figure 3. We will utilise this notion in Section 5 to save leading coefficients.

The proofs of the lemmas in this section may be found in Appendix B.
Definition 4.1 (Projective delineability). Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$ be a cell, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, level $(p)=i+1$ be a polynomial. Then $p$ is projectively delineable on $R$ if and only if either (i) $p\left(r, r^{\prime}\right) \neq 0$ for all $r \in R$ and $r^{\prime} \in \mathbb{R}$, or (ii) $p$ is delineable on $R$, or (iii) $S=\operatorname{realRoots}\left(\operatorname{ldcf}_{x_{i+1}}(p)\right) \cap R \neq \emptyset$ and there exist finitely many continuous functions $\theta_{1}, \ldots, \theta_{k}: R \rightarrow \mathbb{R}$ and a function $\theta_{*}: R \backslash S \rightarrow \mathbb{R}$ continuous on the connected components of its domain such that:

- $\theta_{1}(r)<\ldots<\theta_{k}(r)$ for all $r \in R$;


Figure 2: Construction of a single cell for Example 3.1 following the new levelwise approach as described in Example 3.2.

- if $k>0$, then for all $r \in R \backslash S$, either $\theta_{*}(r)<\theta_{1}(r)$ or $\theta_{k}(r)<\theta_{*}(r)$;
- for any continuous $L:[0,1] \rightarrow R$ such that $\operatorname{ldcf}_{x_{i+1}}(p)(L(t))=0$ at $t=1$ and for no other $L\left(t^{\prime}\right), t^{\prime} \in[0,1)$ it holds that $\lim _{t \rightarrow 1-} \theta_{*}(L(t))= \pm \infty$;
- the set of real roots of the univariate polynomial $p\left(r, x_{i+1}\right)$ in $x_{i+1}$ is $\left\{\theta_{1}(r), \ldots, \theta_{k}(r), \theta_{*}(r)\right\}$ for all $r \in R \backslash S$ and $\left\{\theta_{1}(r), \ldots, \theta_{k}(r)\right\}$ for all $r \in S$; and
- there exist constants $m_{1}, \ldots, m_{k}, m_{*} \in \mathbb{N}_{>0}$ such that for all $r \in R$ and all $j \in[k]$, $m_{j}$ is the multiplicity of the root $\theta_{j}(r)$ of $p\left(r, x_{i+1}\right)$ and for all $r \in R \backslash S, m_{*}$ is the multiplicity of the root $\theta_{*}(r)$ of $p\left(r, x_{i+1}\right)$.

Definition 4.2 (Analytic projective delineability). Analytic projective delineability is defined as projective delineability in Definition 4.1, with the modification that it is only defined on connected analytic submanifolds (instead of general cells) and the functions $\theta_{1}, \ldots \theta_{k}$ are required to be analytic and $\theta_{*}$ analytic on the connected components of its domain (instead of only continuous).

The key property, that order-invariance of the discriminant alone, without looking at the leading coefficient, is enough to guarantee projective delineability, is captured in Lemma 4.1 below.

Lemma 4.1. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$ be a connected analytic submanifold, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, level $(p)=i+1$ be irreducible. If $p$ is not nullified on any point in $R$ and $\operatorname{disc}_{x_{i+1}}(p)$ is order-invariant on $R$, then $p$ is analytically projectively delineable on $R$.

The following theorem states the relationship between delineability and projective delineability.
Lemma 4.2. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$ be a cell and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, $\operatorname{level}(p)=i+1$.
If $p$ is projectively delineable on $R$ and $\operatorname{ldcf}_{x_{i+1}}(p)$ is sign-invariant on $R$, then $p$ is delineable on $R$.
Note that we already defined real root functions of delineable polynomials on arbitrary cells in Definition 2.1; here, we define real root functions for the case of analytic projective delineability.


Figure 3: In this situation, projective delineability of the polynomial is enough for constructing a sign-invariant cell, i.e the leading coefficient could be omitted from the projection.

Definition 4.3 (Real root function). Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$ be a connected analytic submanifold, $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, $\operatorname{level}(p)=i+1$, and $\theta: R \rightarrow \mathbb{R}$ be an analytic function. We call $\theta$ a real root function of $p$ on $R$ if $p(r, \theta(r))=0$ for all $r \in R$.

Definition 4.4. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$ be an analytic submanifold, $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, $\operatorname{level}(p)=i+1, r \in R$ and $\xi \in \operatorname{irExpr}(p, r)$.

If $p$ is analytically projectively delineable on $R$ and there exists a real root function $\theta$ of $p$ on $R$ such that $\theta(r)=\xi(r)$, then $\theta_{\xi, r}$ denotes this (unique) real root function.

The following results are analogous to the known properties for delineability, showing how to lift order invariance and ensure delineability of sets of polynomials.

We first give a lemma proving order-invariance of polynomials for the non-trivial case where a polynomial is zero (remembering that a polynomial is trivially order-invariant on sets where it is non-zero).

Lemma 4.3. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$ be a cell, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$, level $(p)=i$ such that $p(r)=0$ for all $r \in R$.
If $i=0$, then $p$ is order-invariant on $R$. If $i>0, R \downarrow_{[i-1]}$ is an analytic submanifold, $p$ is not nullified on any point in $R \downarrow_{[i-1]}$, and $p$ is analytically projectively delineable on $R \downarrow_{[i-1]}$, then $p$ is order-invariant on $R$.

We next present a lemma for ensuring the order of two real root functions from different polynomials; if they stem from the same polynomial, projective delineability of this polynomial already suffices.

Lemma 4.4. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$ be a connected analytic submanifold and $p_{1}, p_{2} \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$ irreducible and coprime such that level $\left(p_{1}\right)=\operatorname{level}\left(p_{2}\right)=i+1, \theta_{1}, \theta_{2}: R \rightarrow \mathbb{R}$ be real root functions of $p_{1}$ and $p_{2}$ respectively, and $\sim \in\{=,<,>\}$.

If $p_{1}, p_{2}$ are not nullified on any point in $R$ and $\operatorname{res}_{x_{i+1}}\left(p_{1}, p_{2}\right)$ is order-invariant on $R$ and $p_{1}$ and $p_{2}$ are projectively delineable on $R$, then $\theta_{1}(r) \sim \theta_{2}(r) \Longleftrightarrow \theta_{1}\left(r^{\prime}\right) \sim \theta_{2}\left(r^{\prime}\right)$ for all $r, r^{\prime} \in R$.

## 5. Proof system for single cells

In Section 6 we will present our new levelwise procedure to construct a single cell. We lay the basis for this by first formalising our work as a proof system in this section. The system expresses how the properties we want to conclude rely on other "smaller" properties. From this, the procedure in the next section gains maximal freedom to make heuristic choices. Before we go into the formal presentation, we outline the idea for our running example.

### 5.1. Motivating example

Example 5.1. Recall Example 3.1 and Figure 2 where we sought to construct a cell $R \subseteq \mathbb{R}^{2}$ around $s=\left(s_{1}, s_{2}\right)$ such that the sign-invariance of $p_{1}, p_{2}, p_{3}$ hold on $R$. We started by eliminating $x_{2}$ : we isolated real roots of $p_{j}\left(s_{1}, x_{2}\right)$ to define $\xi_{1}, \ldots \xi_{4}$, sorted them, and observed that $\xi_{1}\left(s_{1}\right)<s_{2}<\xi_{2}\left(s_{1}\right)$, thus we determined that in the second level the cell was represented by $\mathrm{I}_{2}=\left(\right.$ sector $\left., \xi_{1}, \xi_{2}\right)$.

```
            \(\operatorname{sgn\_ inv}\left(p_{1}\right)(R) \dashv \operatorname{sample}(s)(R), \operatorname{repr}\left(\mathrm{I}_{2}, s_{1}\right)(R), \operatorname{ir\_ ord}\left(\preceq, s_{1}\right)(R)\), an_proj_del\(\left(p_{1}\right)(R)\),
                \(\operatorname{sgn\_ inv}\left(\operatorname{ldcf}_{x_{2}}\left(p_{1}\right)\right)(R), \operatorname{an\_ sub}(1)(R)\), connected \((1)(R)\)
    \(\operatorname{sgn\_ inv}\left(p_{2}\right)(R) \dashv \operatorname{sample}(s)(R), \operatorname{repr}\left(\mathrm{I}_{2}, s_{1}\right)(R), \operatorname{ir\_ ord}\left(\preceq, s_{1}\right)(R), \operatorname{an\_ proj\_ del}\left(p_{2}\right)(R)\),
                                    an_sub \((1)(R)\), connected \((1)(R)\)
    \(\operatorname{sgn\_ inv}\left(p_{3}\right)(R) \dashv \operatorname{sample}(s)(R)\), repr \(\left(\mathrm{I}_{2}, s_{1}\right)(R), r_{\_} \operatorname{ord}\left(\preceq, s_{1}\right)(R)\), an_proj_del \(\left(p_{3}\right)(R)\),
                sgn_inv \(\left(\operatorname{ldcf}_{x_{2}}\left(p_{3}\right)\right)(R)\), an_sub \((1)(R)\), connected \((1)(R)\)
    sample \((s)(R) \dashv \operatorname{repr}\left(\mathrm{I}_{2}, s_{1}\right)(R)\), sample \(\left(s_{1}\right)(R)\)
    \(\operatorname{repr}\left(\mathrm{I}_{2}, s_{1}\right)(R) \dashv R=\operatorname{set} O f\left(R \downarrow_{[1]}, \mathrm{I}_{2}\right), \operatorname{an\_ proj\_ del}\left(p_{2}\right)(R), \operatorname{an\_ proj\_ del}\left(p_{3}\right)(R)\),
        sgn_inv \(\left(\operatorname{ldcf}_{x_{2}}\left(p_{2}\right)\right)(R), \operatorname{sgn\_ inv}\left(\operatorname{ldcf}_{x_{2}}\left(p_{3}\right)\right)(R), \operatorname{sample}\left(s_{1}\right)(R)\)
        \(\operatorname{ir\_ ord}\left(\preceq, s_{1}\right)(R) \dashv a n_{-} p r o j \_d e l\left(p_{1}\right)(R), a n \_p r o j \_d e l\left(p_{2}\right)(R), a n_{-} p r o j \_d e l\left(p_{3}\right)(R), o r d \_i n v\left(\operatorname{res}_{x_{2}}\left(p_{3}, p_{1}\right)\right)(R)\),
        \(\operatorname{ord\_ inv}\left(\operatorname{res}_{x_{2}}\left(p_{3}, p_{2}\right)\right)(R), \operatorname{an\_ sub}(1)(R)\), connected \((1)(R), \operatorname{sample}\left(s_{1}\right)(R)\)
    an_proj_del \(\left(p_{1}\right)(R) \dashv\) non_null \(\left(p_{1}\right)(R)\), ord_inv \(\left(\operatorname{disc}_{x_{2}}\left(p_{1}\right)\right)(R)\), an_sub \((1)(R)\), connected \((1)(R)\)
    an_proj_del \(\left(p_{2}\right)(R) \dashv\) non_null \(\left(p_{2}\right)(R)\), ord_inv \(\left(\operatorname{disc}_{x_{2}}\left(p_{2}\right)\right)(R)\), an_sub \((1)(R)\), connected \((1)(R)\)
    an_proj_del \(\left(p_{3}\right)(R) \dashv\) non_null \(\left(p_{3}\right)(R)\), ord_inv \(\left(\operatorname{disc}_{x_{2}}\left(p_{3}\right)\right)(R)\), an_sub \((1)(R)\), connected \((1)(R)\)
    non_null \(\left(p_{1}\right)(R) \dashv t r u e\)
    non_null \(\left(p_{2}\right)(R) \dashv\) true
    non_null \(\left(p_{3}\right)(R) \dashv\) true
    \(\operatorname{ord\_ inv}\left(\operatorname{disc}_{x_{2}}\left(p_{1}\right)\right)(R) \dashv\) true
    \(\operatorname{ord\_ inv}\left(\operatorname{disc}_{x_{2}}\left(p_{2}\right)\right)(R) \dashv \operatorname{sgn\_ inv}\left(p_{4}\right)(R)\), sample \(\left(s_{1}\right)(R)\)
    ord_inv \(\left(\operatorname{disc}_{x_{2}}\left(p_{3}\right)\right)(R) \dashv\) true
        \(\operatorname{sgn\_ inv}\left(p_{4}\right)(R) \dashv \operatorname{repr}\left(\mathrm{I}_{1}\right)(R)\)
ord_inv \(\left(\operatorname{res}_{x_{2}}\left(p_{3}, p_{1}\right)\right)(R) \dashv\) true
\(\operatorname{ord\_ inv}\left(\operatorname{res}_{x_{2}}\left(p_{3}, p_{2}\right)\right)(R) \dashv \operatorname{sgn} \_i n v\left(p_{5}\right)(R)\), sample \(\left(s_{1}\right)(R)\)
```

```
            \(\operatorname{sgn} \_i n v\left(p_{5}\right)(R) \dashv \operatorname{repr}\left(\mathrm{I}_{1}\right)(R)\)
```

            \(\operatorname{sgn} \_i n v\left(p_{5}\right)(R) \dashv \operatorname{repr}\left(\mathrm{I}_{1}\right)(R)\)
    $\operatorname{sgn\_ inv}\left(\operatorname{ldcf}_{x_{2}}\left(p_{1}\right)\right)(R) \dashv$ true
$\operatorname{sgn\_ inv}\left(\operatorname{ldcf}_{x_{2}}\left(p_{1}\right)\right)(R) \dashv$ true
sgn_inv $\left(\operatorname{ldcf}_{x_{2}}\left(p_{2}\right)\right)(R) \dashv$ true
sgn_inv $\left(\operatorname{ldcf}_{x_{2}}\left(p_{2}\right)\right)(R) \dashv$ true
sgn_inv $\left(\operatorname{ldcf}_{x_{2}}\left(p_{3}\right)\right)(R) \dashv$ true
sgn_inv $\left(\operatorname{ldcf}_{x_{2}}\left(p_{3}\right)\right)(R) \dashv$ true
$\operatorname{an\_ sub}(1)(R) \dashv \operatorname{repr}\left(\mathrm{I}_{1}\right)(R)$
$\operatorname{an\_ sub}(1)(R) \dashv \operatorname{repr}\left(\mathrm{I}_{1}\right)(R)$
connected $(1)(R) \dashv$ true
connected $(1)(R) \dashv$ true
sample $\left(s_{1}\right)(R) \dashv \operatorname{repr}\left(\mathrm{I}_{1}\right)(R)$
sample $\left(s_{1}\right)(R) \dashv \operatorname{repr}\left(\mathrm{I}_{1}\right)(R)$
$\operatorname{repr}\left(\mathrm{I}_{1}\right)(R) \dashv R \downarrow_{[1]}=\operatorname{set} O f\left(\mathrm{I}_{1}\right)$

```
    \(\operatorname{repr}\left(\mathrm{I}_{1}\right)(R) \dashv R \downarrow_{[1]}=\operatorname{set} O f\left(\mathrm{I}_{1}\right)\)
```

Figure 4: Proof for Example 5.1 (slightly truncated, read in columns). $Q \dashv P_{1}, \ldots, P_{k}$ means that $Q$ is derived from $P_{1}, \ldots, P_{k}$.

This root ordering and the representation should now be generalized such that the underlying cell $R_{1}$ is restricted by specifying certain properties ensuring $\xi_{1}$ and $\xi_{2}$ remain as cell boundaries: they must remain well-defined over $R_{1}$; no further root of $p_{1}, p_{2}, p_{3}$ should appear that intersects $\mathrm{I}_{2}$ over $R_{1}$; and the other roots, $\xi_{3}$ and $\xi_{4}$, should remain outside of the symbolic interval $\mathrm{I}_{2}$ over $R_{1}$. For the latter, we extend to a partial ordering $\preceq$ with $\xi_{2} \preceq \xi_{3}$ and $\xi_{2} \preceq \xi_{4}$.

To conclude that $p_{1}, p_{2}, p_{3}$ are sign-invariant in $R$, the proof system shows we need to maintain the properties that: the sample $s$ is included in $R ; \mathrm{I}_{2}$ describes the cell's boundaries on the current level; the underlying cell $R_{1}$ is a connected analytic submanifold and; the that $p_{1}, p_{2}, p_{3}$ are projective delineability. For some polynomials we also need to ensure full delineability. Furthermore, we maintain that the partial ordering $\preceq$ is maintained over $R_{1}$.

To maintain these properties the proof system concludes that we must also prove order-invariance and signinvariance of some coefficients, resultants, and discriminants of the input polynomials. After simplification, we find that on level 1 the polynomial $p_{4}=\operatorname{disc}_{x_{2}}\left(p_{2}\right)$ and $p_{5}=\operatorname{res}_{x_{2}}\left(p_{3}, p_{2}\right)$ must be ensured sign-invariant. This leads to a representation $\mathrm{I}_{1}=\left(\right.$ sector, $\left.\xi_{2}^{\prime}, \xi_{3}^{\prime}\right)$ and then since all polynomials are univariate and their zeros are algebraic numbers, no further projection or proof steps are necessary. The constructed cell is described by $R=\left(\mathrm{I}_{1}, \mathrm{I}_{2}\right)$.

The graph of the proof constructed is shown in Figure 4, in which the sign-invariance properties sgn_inv $\left(p_{1}\right)$, $\operatorname{sgn\_ inv}\left(p_{2}\right)$, sgn_inv $\left(p_{3}\right)$ lead to $R=\operatorname{setOf}\left(R \downarrow_{[1]}, \mathrm{I}_{2}\right)$ and $R \downarrow_{[1]}=\operatorname{setOf}\left(\mathrm{I}_{1}\right)$. The exact proof rules used in that figure will be defined in the rest of this section.

Note how in the example there were multiple choices for $\preceq$. Choosing $\xi_{2} \preceq \xi_{3}$ and $\xi_{3} \preceq \xi_{4}$ would have been a valid choice but that leads to a smaller cell (the one depicted in the top right of Figure 1). We will discuss ordering heuristics to make this choice in Section 7.1.

### 5.2. Proof system: Properties and rules

Our system is for constructing cells. We will define properties of the cell that is being constructed and rules of inference to infer that a property holds for a cell given some other properties and conditions. As indicated in the previous example, there may be different rules that can be used to prove a property.

Definition 5.1 (Property). Let $i \in \mathbb{N}$. A property of level $i$ is a function $q:\left\{R \mid R \subseteq \mathbb{R}^{i}\right\} \rightarrow \mathbb{B}$. A property $q$ of level $i$ holds on a set $R \subseteq \mathbb{R}^{i}$ if $q(R)=1$. The set Prop denotes the set of all properties of any level. For a set $Q \subseteq$ Prop, we denote by $\left.Q\right|_{i}$ the set of properties of level $i$ and by $\left.Q\right|_{[i]}$ the set of properties of level at most $i$.

Note that the concept of levels of polynomials has been extended to properties in the sense that the satisfaction of a property of level $i$ cannot be influenced by values of the variables $x_{i+1}, \ldots, x_{n}$.

We denote a rule of inference in the form $P_{1}, \ldots, P_{k} \vdash Q$, where $A_{1}, \ldots, A_{k}$ are the antecedents and $Q$ is the consequent.

For convenience, we allow to lift properties from lower levels to higher levels as follows.
Definition 5.2. We extend every property $q$ of level $i$ to $q:\left\{R \mid R \subseteq \mathbb{R}^{j}, j \geq i\right\} \rightarrow \mathbb{B}$ such that $q(R)=q\left(R \downarrow_{[i]}\right)$ for any $R \subseteq \mathbb{R}^{j}, j \geq i$. Accordingly, we introduce the following rule of inference:

$$
R \subseteq \mathbb{R}^{i+1},\left.q \in \operatorname{Prop}\right|_{i}, q\left(R \downarrow_{[i]}\right) \quad \vdash q(R)
$$

So, for example, this rule allows us to say that because the property of sign-invariance holds for $x^{2}-2$ in the one-level cell $-1<x<1$, sign-invariance holds for $x^{2}-2$ in the two-level cell which is the interior of the unit circle, since this projects down to $-1<x<1$.

The set of inference rules will be defined in such a way that the property of sign-invariance of a polynomial is reduced to the property of sign-invariance of polynomials at a lower level: thus the chain of rules is finite. On each level, we will additionally determine a representation describing the symbolic boundaries of the cell on this level as well as an ordering of the indexed roots of some polynomials assuring their sign-invariance which will be used later for heuristic decisions.

### 5.3. Basic cell properties

We will first define the basic properties on polynomials and cells that follow from the original work on McCallum CAD projection. Throughout, $i$ will refer to the level of the given property and $s$ refers to algebraic points. It is important to recall that an $i$-level property is a function that maps subsets of $\mathbb{R}^{i}$ to Boolean values. Consider for example the first element of Property 5.1: "sample(s)" where $s \in \mathbb{R}^{i}$ is a property, meaning that sample is a function that maps a point (and a level $i$ that is given implicitly by the point) to a function mapping subsets of $\mathbb{R}^{i}$ to Booleans. Thus, sample $(s)$ is not a true/false value. Rather, it is the function sample(s) applied to a given subset of $\mathbb{R}^{i}$ that yields a true/false value. Note that for the fifth and sixth items in Property 5.1 the level $i$ must be given explicitly, while with the others the level is implicit in other arguments.

Property 5.1. Let $i \in \mathbb{N}$, and $R \subseteq \mathbb{R}^{i}$.

1. Let $s \in \mathbb{R}^{i}$ be a sample point. The property sample $(s)$ holds on $R$ if and only if $s \in R$.
2. Let $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$ be a polynomial of level $i$. The property ord_inv $(p)$ holds on $R$ if and only if $p$ is order-invariant on $R$.
3. Let $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$ be a polynomial of level $i$. The property sgn_inv $(p)$ holds on $R$ if and only if $p$ is sign-invariant on $R$.
4. Let $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$ be a polynomial of level $i+1$. The property non_null $(p)$ holds on $R$ if and only if $p$ is not nullified on any point in $R$.
5. The property an_sub(i) holds on $R$ if and only if $R$ is an analytic submanifold.
6. The property connected $(i)$ holds on $R$ if and only if $R$ is connected.
7. Let $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$ be a polynomial of level $i+1$. The property an_proj_del $(p)$ holds on $R$ if and only if $p$ is analytically projectively delineable on some connected superset $R^{\prime} \supseteq R$.

Note that an_proj_del $(p)$ is defined such that it holds on a subset $R \subseteq \mathbb{R}^{i}$ if and only if there is a connected superset of $R$ on which $p$ is analytically projectively delineable. This is due to the fact that in general, we cannot assume that $R$ is a connected set, but projective delineability is only defined on connected sets. We will use this trick in the following for other properties as well.

Further note that for some properties such as $\operatorname{sgn} \operatorname{inv}(p)$, if they hold on $R \subseteq \mathbb{R}^{i}$, then they also hold on all subsets of $R$. For others such as $a n_{\_} s u b(i)$ or connected $(i)$, this is not the case: a subset of $R$ is not necessarily an
analytic submanifold nor connected! This is one of the reasons why the proof rules below cannot cover all subsets where a given property holds, but they do cover sufficiently many subsets for our purposes.

For the remainder of this sub-section we will present proof rules for these properties (with proofs of correctness available in Appendix C). For the ease of navigation we give references to the rules which apply to prove each property above (and provide similar references after introducing further properties in later sub-sections).

## References.

sample(s) Rule 5.13
$\boldsymbol{o r d}$ _inv(p) Rule 5.3, Rule 5.4, Rule 5.5
$\boldsymbol{s g n} \boldsymbol{i n v}(\boldsymbol{p})$ Rule 5.3, Rule 5.4, Rule 5.6, Rule 5.8, Rule 5.11
non_null(p) Rule 5.2
$\boldsymbol{a n} \_\boldsymbol{s u b}(\boldsymbol{i})$ Rule 5.7
connected(i) Rule 5.12
an_proj_del(p) Rule 5.1

In the following rule definitions we list assumptions before the actual inference rule. These assumptions are formally part of the inference rule, but we keep them separated to improve readability.

We start by defining rules that relate to the material in Section 4.
Rule 5.1. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, level $(p)=i+1$. Assume that $p$ is irreducible.

$$
\operatorname{an\_ sub}(i)(R), \text { connected }(i)(R), \text { non_null }(p)(R), \operatorname{ord\_ inv}\left(\operatorname{disc}_{x_{i+1}}(p)\right)(R) \quad \vdash \text { an_proj_del }(p)(R)
$$

Rule 5.2. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i}$, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, level $(p)=i+1$. Assume that $p$ is irreducible, and $p=c_{m} \cdot x_{i+1}^{m}+\ldots+c_{1} \cdot x_{i+1}+c_{0}$ such that $c_{m}, \ldots, c_{0} \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$.

$$
\begin{array}{ll}
\operatorname{sample}(s)(R), \operatorname{deg}_{x_{i+1}}(p)>1, \operatorname{disc}_{x_{i+1}}(p)(s) \neq 0, \operatorname{sgn\_ inv}\left(\operatorname{disc}_{x_{i+1}}(p)\right)(R) & \vdash \text { non_null }(p)(R) \\
\operatorname{sample}(s)(R), \exists j \in[m] .\left(c_{j}(s) \neq 0 \wedge \operatorname{sgn\_ inv}\left(c_{j}\right)(R)\right) & \vdash \text { non_null }(p)(R)
\end{array}
$$

Note that the condition $\operatorname{deg}_{x_{i+1}}(p)>1 \wedge \operatorname{disc}_{x_{i+1}}(p)(s) \neq 0$ already implies that $p\left(s, x_{i+1}\right) \neq 0$, as does $c_{j}(s) \neq 0$. In practice it is good to delay choosing which rule to use because we may observe that one of the non-zero $c_{j}$ 's is already required to be sign-invariant from some other part of the projection process (e.g. leading coefficients are often added to the projection). Moreover, the second case is trivial if the $c_{j}$ is constant.

Rule 5.3. Let $p \in \mathbb{Q}$.

$$
\begin{aligned}
& \vdash \operatorname{ord\_ inv}(p)\left(\mathbb{R}^{0}\right) \\
& \vdash \operatorname{sgn\_ inv}(p)\left(\mathbb{R}^{0}\right)
\end{aligned}
$$

Rule 5.4. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}$, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$, level $(p)=i$. Assume that $p$ is reducible, and factors $(p)=$ $\left\{q_{1}, \ldots, q_{k}\right\}$.

$$
\begin{array}{ll}
\operatorname{ord\_ inv}\left(q_{1}\right)(R), \ldots, \text { ord_inv }\left(q_{k}\right)(R) & \vdash \operatorname{ord\_ inv}(p)(R) \\
\operatorname{sgn} i n v\left(q_{1}\right)(R), \ldots, \operatorname{sgn\_ inv}\left(q_{k}\right)(R) & \vdash \operatorname{sgn\_ inv}(p)(R)
\end{array}
$$

Rule 5.5. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i}$, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$, level $(p)=i$. Assume that $p$ is irreducible.

$$
\begin{array}{ll}
p(s) \neq 0, \text { sample }(s)(R), \text { sgn_inv }(p)(R) & \vdash \text { ord_inv }(p)(R) \\
p(s)=0, \text { sample }(s)(R), \text { an_sub }(i-1)(R), \text { connected }(i)(R), \operatorname{sgn\_ inv}(p)(R), \text { an_proj_del }(p)(R) & \vdash \text { ord_inv }(p)(R)
\end{array}
$$

### 5.4. Sign invariance for polynomials without roots

For polynomials without roots over the current sample, we must ensure that no further root appears over the underlying cell. Whenever the leading coefficient of such a polynomial is not zero at the underlying sample, projective delineability ensures that no further root appears. Otherwise, if the leading coefficient is zero, its signinvariance must be maintained to prevent the special root function $\theta_{*}$ from Definition 4.1 popping up over the underlying cell. Figure 5 illustrates the two cases.


Figure 5: Examples of the two cases of polynomials without roots at the projection $s_{1}$ of the sample $s=\left(s_{1}, s_{2}\right)$. Dashed lines denote cylinder boundaries over the projected cell.

Rule 5.6. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i}$, and $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i+1}\right]$, $\operatorname{level}(p)=i+1$. Assume that $p$ is irreducible, and $\operatorname{realRoots}\left(p\left(s, x_{i+1}\right)\right)=\emptyset$.

$$
\begin{aligned}
\text { sample }(s)(R), \text { an_proj_del }(p)(R), \operatorname{ldcf}_{x_{i+1}}(p)(s) \neq 0 & \vdash \operatorname{sgn\_ inv}(p) \\
\operatorname{sample}(s)(R), \text { an_proj_del }(p)(R), \operatorname{ldcf}_{x_{i+1}}(p)(s)=0, \operatorname{sgn\_ inv}\left(\operatorname{ldcf}_{x_{i+1}}(p)\right)(R) & \vdash \operatorname{sgn\_ inv(p)}
\end{aligned}
$$

### 5.5. Cell boundary representations

We are aiming to generate the description of a sign-invariant cell for the input polynomials. In the language of our proof system: at every level we identify a symbolic interval upon which we prove sign-invariance of the polynomials of that level. To do this, we take their real roots over the current sample into account.

We determine whether we are in a section or sector (i.e. whether a polynomial has a root exactly at the current sample point or not) and pick indexed root expressions (as given by Definition 2.5) to be symbolic descriptions of the cell boundaries that generalize a concrete interval around the sample point: either the root at the current sample or the largest root below and the lowest root above. Note that the choice of such boundaries might not be unique in the cases where two root functions cross over the underlying sample. We will discuss more general choices of the symbolic intervals in Section 6.2.

Our next property is designed to state that a designated representation describes the cell boundaries on the current level.

Property 5.2. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i-1}$, and I be a symbolic interval of level $i$.
The property repr $(\mathrm{I}, s)$ holds on $R$ if and only if $\mathrm{I} . l \in \operatorname{irExpr}(\mathrm{I} . l . p, s)$ (if $\mathrm{I} . l \neq-\infty)$, $\mathrm{I} . u \in \operatorname{irExpr}(\mathrm{I} . u . p, s)$ (if I. $u \neq \infty$ ) respectively I.b $\in \operatorname{irExpr}(\mathrm{I} . b . p, s)$ and one of the following hold:

- $\mathrm{I}=($ sector $, l, u), \operatorname{dom}\left(\theta_{l, s}\right) \cap \operatorname{dom}\left(\theta_{u, s}\right) \supseteq R \downarrow_{[i-1]}$ and $R=\left\{\left(r, r^{\prime}\right) \mid r \in R \downarrow_{[i-1]}, r^{\prime} \in\left(\theta_{l, s}(r), \theta_{u, s}(r)\right)\right\}$; or
- $\mathrm{I}=($ sector $,-\infty, u), \operatorname{dom}\left(\theta_{u, s}\right) \supseteq R \downarrow_{[i-1]}$ and $R=\left\{\left(r, r^{\prime}\right) \mid r \in R \downarrow_{[i-1]}, r^{\prime} \in\left(-\infty, \theta_{u, s}(r)\right)\right\}$; or
- $\mathrm{I}=($ sector $, l, \infty), \operatorname{dom}\left(\theta_{l, s}\right) \supseteq R \downarrow_{[i-1]}$ and $R=\left\{\left(r, r^{\prime}\right) \mid r \in R \downarrow_{[i-1]}, r^{\prime} \in\left(\theta_{l, s}(r), \infty\right)\right\}$; or
- $\mathrm{I}=($ sector $,-\infty, \infty)$ and $R=\left\{\left(r, r^{\prime}\right) \mid r \in R \downarrow_{[i-1]}, r^{\prime} \in \mathbb{R}\right\}$; or
- $\mathrm{I}=($ section, $b), \operatorname{dom}\left(\theta_{b, s}\right) \supseteq R \downarrow_{[i-1]}$ and $R=\left\{\left(r, r^{\prime}\right) \mid r \in R \downarrow_{[i-1]}, r^{\prime}=\theta_{b, s}(r)\right\}$,
(for the real root functions $\theta_{l, s}, \theta_{u, s}$ respectively $\theta_{u, b}$ according to Definition 4.4).
References. $\operatorname{repr}(\mathrm{I}, s)$ Rule 5.14
Since the cell's boundaries are described by indexed root expressions, we can prove it is an analytic submanifold.
Rule 5.7. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i-1}$, and I be a symbolic interval of level $i$.

$$
\vdash \operatorname{an\_ sub}(0)\left(\mathbb{R}^{0}\right)
$$

$$
\operatorname{repr}(\mathrm{I}, s)(R), \text { an_sub }(i-1)(R) \quad \vdash \operatorname{an\_ sub}(i)(R)
$$

### 5.6. Equational constraint projection

An optimisation to CAD which tailors it to the logical structure of the problem is the theory of equational constraints. A polynomial equation is an equational constraint if it is logically implied by the truth of the input formula. If the input to CAD has an equational constraint then we may perform fewer projection and lifting operations to achieve truth-invariance. This optimisation dates back to [34] with the recent paper [23] summarising the state of the art. The examples there demonstrate the drastic savings that are achievable in these cases (the analysis in [23] shows the double exponent in the complexity bound decreases for each equational constraint).

In the context of building a single cell we have even greater scope to use equational constraint savings. Here, any time we are generalising in the section case, we can apply the reduced projection and lifting. I.e., if a polynomial is zero at our sample then it must also be zero throughout the cell we are generalising the sample to, and so for the purposes of this cell, it is an equational constraint. In this case we need only make the section defining polynomial delineable. All other constraints can be made sign-invariant simply by including resultants with the section defining polynomial (but not other projection polynomials).

Rule 5.8. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i-1}, p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$, level $(p)=i$, and I be a symbolic interval of level $i$. Assume that $p$ is irreducible, and $\mathrm{I}=($ section, $b)$.

Let

$$
\begin{array}{rlr}
Q & :=\text { an_sub }(i-1)(R) \wedge \text { connected }(i-1)(R) \wedge r e p r(I, s)(R), \text { and } \\
& Q_{\text {delineable }}(q) & :={\text { an_proj_del }(q)(R) \wedge \operatorname{sgn\_ inv}\left(\operatorname{ldcf}_{x_{i}}(q)\right)(R) .} \\
Q, Q_{\text {delineable }}(b . p), b . p=p & & \vdash \operatorname{sgn\_ inv}(p)(R) \\
Q, Q_{\text {delineable }}(b . p), b . p \neq p, \operatorname{ord\_ inv}\left(\operatorname{res}_{x_{i}}(b . p, p)\right)(R) & \vdash \operatorname{sgn\_ inv}(p)(R)
\end{array}
$$

### 5.7. Root orderings

We define a partial ordering on the indexed root expressions of the given set of polynomials, which we will use to ensure that none of their roots cross the cell's boundary on the current level.

We first give a general definition of indexed root orderings.
Definition 5.3 (Indexed root ordering). Let $i \in \mathbb{N}$, and $\Xi$ be a set of indexed root expressions of level $i+1$. An indexed root ordering on $\Xi$ is a relation $\preceq \subseteq \Xi \times \Xi$ such that its reflexive and transitive closure $\preceq^{t}$ is a partial order on $\Xi$. Indexed root orderings of this form are also called indexed root ordering of level $i+1$, and we define $\operatorname{dom}(\preceq)=\left\{\xi, \xi^{\prime} \mid\left(\xi, \xi^{\prime}\right) \in \preceq\right\}$.

Let $s \in \mathbb{R}^{i}$. An indexed root ordering $\preceq$ of level $i+1$ matches $s$ if and only if $\xi \in \operatorname{irExpr}(\xi . p, s)$ for all $\xi \in \operatorname{dom}(\preceq)$ and $\xi(s) \leq \xi^{\prime}(s)$ for all $\left(\xi, \xi^{\prime}\right) \in \preceq$.

In our algorithm we will pick an indexed root ordering such that, in the sector case, roots lower than the intervals's lower bound remain lower, and roots greater than the intervals's upper bound remain greater. In the section case, the lower and upper bounds in that condition both refer to a single bound. We do not give an explicit definition here yet, as it will be part of the rules defined below.

The motivation for introducing the general concept of indexed root orderings is to allow for choices between different root orderings. We will discuss them in Section 6.2.

Example 5.2. Given indexed root expressions $\Xi=\left\{\xi_{1}, \ldots, \xi_{5}\right\}$ of level $i+1$ and a sample $s \in \mathbb{R}^{i}$ such that $\xi_{1}(s)<\ldots<\xi_{5}(s)$ and $\mathrm{I}=\left(\right.$ sector, $\left.\xi_{1}, \xi_{2}\right) ;$ then $\preceq=\left\{\left(\xi_{2}, \xi_{3}\right),\left(\xi_{2}, \xi_{4}\right),\left(\xi_{2}, \xi_{5}\right)\right\}, \preceq^{\prime}=\left\{\left(\xi_{2}, \xi_{3}\right),\left(\xi_{3}, \xi_{4}\right),\left(\xi_{2}, \xi_{5}\right)\right\}$, and $\preceq^{\prime \prime}=\left\{\left(\xi_{2}, \xi_{3}\right),\left(\xi_{3}, \xi_{4}\right),\left(\xi_{4}, \xi_{5}\right)\right\}$ are all indexed root orderings on $\Xi$ that allow us to derive sign-invariance of the corresponding polynomials above the sample $s$ and the interval I.

Now, we must maintain that an indexed root ordering holds over the underlying cell. First, we will define this property formally. it is not straight forward as we do allow an optimisation whereby the referred root functions may disappear (i.e. not to be defined) over parts of the cell, so long as this does not change the ordering.

Property 5.3. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i}$, $\preceq$ be an indexed root ordering of level $i+1$, and $\preceq^{t}$ be the reflexive and transitive closure of $\preceq$.

The property $\operatorname{ir\_ ord}(\preceq, s)$ holds on $R$ if and only if $\preceq$ matches s and for all $\xi, \xi^{\prime} \in \operatorname{dom}(\preceq)$ it holds that

$$
\xi \preceq^{t} \xi^{\prime} \wedge \xi(s)<\xi^{\prime}(s) \Longrightarrow \theta_{\xi, s}(r)<\theta_{\xi^{\prime}, s}(r) \text { for all } r \in \operatorname{dom}\left(\theta_{\xi, s}\right) \cap \operatorname{dom}\left(\theta_{\xi^{\prime}, s}\right) \cap R
$$


(a) In the simple case all three real root functions are defined over the underlying cell. Only two resultants are necessary to maintain the ordering of the root functions, e.g. $\operatorname{res}_{x_{2}}\left(\xi_{1} \cdot p, \xi_{2} \cdot p\right)$ and $\operatorname{res}_{x_{2}}\left(\xi_{2} \cdot p, \xi_{3} \cdot p\right)$. We conclude the ordering between $\xi_{1}$ and $\xi_{3}$ by transitivity.

(b) It is also possible that some real root functions disappear over the underlying cell. If this were the highest or lowest then it would not change the ordering. Subsequent ones could also disappear in addition without changing the ordering.

$x_{1}$
(c) If a real root function in the middle were to disappear alone (here $\xi_{2}$ ) then the ordering is maintained by the same resultants and transitivity: i.e. it must converge to $\pm \infty$ when disappearing and thus cross a neighbouring root function, which was protected by the resultant.

Figure 6: Different cases for the property $\operatorname{ir} \_\operatorname{ord}(\preceq, s)$ with $\preceq=\left\{\left(\xi_{1}, \xi_{2}\right),\left(\xi_{2}, \xi_{3}\right)\right\}$, noting that we are only depicting real root functions defined at $s$ and not whole varieties of their defining polynomials.
and

$$
\xi \preceq^{t} \xi^{\prime} \wedge \xi(s)=\xi^{\prime}(s) \Longrightarrow \theta_{\xi, s}(r)=\theta_{\xi^{\prime}, s}(r) \text { for all } r \in \operatorname{dom}\left(\theta_{\xi, s}\right) \cap \operatorname{dom}\left(\theta_{\xi^{\prime}, s}\right) \cap R
$$

(for the real root functions $\theta_{\xi, s}, \theta_{\xi^{\prime}, s}$ according to Definition 4.4).
References. ir_ord($\preceq, s)$ Rule 5.9
Note that the semantics of an indexed root expression $\xi$ is only defined in combination with the sample $s$, as this uniquely determines the real root function $\theta_{\xi, s}$; thus, we add $s$ to the signature of the property. To clarify this further: the sample $s$ is only used to identify the referred real root functions and is not necessarily contained in the constructed cell by definition of the above property (although in the following proof rules, this will be required).

The property is maintained by projective delineability and adding resultants for the defining polynomials of a pair of indexed root expressions to the projection. Note that the property is defined in such a way that no leading coefficients need to be projected. We essentially prove that the ordering of root functions ensured by the order-invariance of resultants is transitive. Figure 6 shows the different cases whereby transitivity is maintained for three root functions.

Rule 5.9. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$, $s \in \mathbb{R}^{i}$, and $\preceq$ be an indexed root ordering of level $i+1$. Assume that $\xi$.p is irreducible for all $\xi \in \operatorname{dom}(\preceq)$, and that $\preceq$ matches $s$.

$$
\begin{aligned}
& \operatorname{sample}(s)(R), \text { an_sub }(i)(R), \text { connected }(i)(R), \forall \xi \in \operatorname{dom}(\preceq) . \text { an_proj_del }(\xi \cdot p)(R), \\
& \forall\left(\xi, \xi^{\prime}\right) \in \preceq . \operatorname{ord\_ }_{-} i n v\left(\operatorname{res}_{x_{i+1}}\left(\xi \cdot p, \xi^{\prime} \cdot p\right)\right)(R) \quad \vdash i r_{-} o r d(\preceq, s)(R)
\end{aligned}
$$

In the following, we need to make sure that certain indexed root expressions are well-defined over the underlying subset $R \subseteq \mathbb{R}^{i}$. To do so, we define a property for an indexed root expression stating that its graph is continuous on some connected superset of $R$.

Property 5.4. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$, $s \in \mathbb{R}^{i}$, $\xi$ be an indexed root expression of level $i+1$.
The property well_def $(\xi, s)$ holds on $R$ if and only if $\xi \in \operatorname{ir} \operatorname{Expr}(\xi \cdot p, s)$ and $\operatorname{dom}\left(\theta_{\xi, s}\right) \supseteq R$ for the real root function $\theta_{\xi, s}$ according to Definition 4.4.

References. well_def( $\boldsymbol{\xi}, \boldsymbol{s}$ ) Rule 5.10
For maintaining this property, we do an elaborated case distinction resulting from the definition of projective delineability, and in some cases we need to fall back to making the corresponding polynomial delineable by adding its leading coefficient to the projection. The different cases are illustrated in Figures 7a to 7c. We note that this case distinction does not detect all superfluous leading coefficient as shown in Figure 7d.

$x_{1}$
(a) The real root function that $\operatorname{root}_{x_{2}}\left[p_{1}, 2\right]$ at $s$ refers to is already well-defined by projective delineability of $p_{1}$, since it is neither the first nor last root of $p_{1}$ at d $s$ and thus cannot be the special $\theta_{*}$ function from Definition 4.1.

$x_{1}$
(b) As the leading coefficient of $p_{1}$ is zero at $s$, no indexed root expression refers to $\theta_{*}$. Thus all real root functions defined at $s$ are welldefined over the underlying cell by projective delineability of $p_{1}$.

$x_{1}$
(c) Here $\operatorname{root}_{x_{2}}\left[p_{1}, 1\right]$ refers to $\theta_{*}$ at $s$. Thus, the leading coefficient of $p_{1}$ is added to the projection.
(d) $\operatorname{root}_{x_{2}}\left[p_{1}, 1\right]$ does not refer to $\theta_{*}$ at $s$. However, this cannot be distinguished from the previous case and so a superfluous leading coefficient is added to the projection.

Figure 7: Ensuring well defined real root functions.

Rule 5.10. Let $i \in \mathbb{N}, R \subseteq \mathbb{R}^{i}$, $s \in \mathbb{R}^{i}$, and $\xi$ be an indexed root expression of level $i+1$. Assume that $\xi \in$ $\operatorname{ir} \operatorname{Expr}(\xi . p, s)$.

```
sample(s)(R), an_proj_del(\xi.p)(R), 1<\xi.j< |realRoots (\xi.p(s, \mp@subsup{x}{i+1}{}))|\quad\vdash\mathrm{ well_def( }\xi,s)(R)
sample(s)(R), an_proj_del(\xi.p)(R), ldcf}\mp@subsup{\mp@code{xi+1}}{}{(\xi\cdotp)(s)=0
sample(s)(R), an_proj_del(\xi.p)(R), sgn_inv(ldcf}\mp@subsup{x}{\mp@subsup{x}{i+1}{}}{}(\xi.p))(R)\quad\vdash\mathrm{ well_def( }(,s)(R
```

Now we are ready to define the rule for ensuring the sign-invariance of a polynomial $p$, given a representation I and an indexed root ordering $\preceq$. The latter will be used to ensure that no real root function of $p$ that is defined at the current sample $s$ (i.e. which is "visible" at $s$ by real root isolation) crosses the cell's boundaries. Additionally to that, we make sure that no further root function of $p$ emerges over the underlying cell. Thus, the many case distinctions follow from the fact that we allow the number of roots of a polynomial to vary over the underlying cell, maintaining only that the appearing roots do not cross the cell's boundary. The cases are depicted in Figure 8.

Rule 5.11. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i-1}, p \in \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$, level $(p)=i$, I be a symbolic interval of level $i$, $\preceq$ be an indexed root ordering of level $i$, and $\preceq^{t}$ be the reflexive and transitive closure of $\preceq$.

We choose $l, u$ such that either $\mathrm{I}=($ sector, $l, u)$ or $\mathrm{I}=($ section, $b)$ and $l=u=b$.
Assume that $p$ is irreducible, $\operatorname{ir} \operatorname{Expr}(p, s) \neq \emptyset$, $\xi . p$ is irreducible for all $\xi \in \operatorname{dom}(\preceq)$, $\preceq$ matches $s$, and for all $\xi \in \operatorname{ir} \operatorname{Expr}(p, s)$ it holds either $\xi \preceq^{t} l$ or $u \preceq^{t} \xi$.

Let

$$
\begin{aligned}
& Q:=\operatorname{sample}(s)(R) \wedge \text { an_sub }(i-1)(R) \wedge \text { connected }(i-1)(R) \\
& \wedge \operatorname{repr}(\mathrm{I}, s)(R) \wedge \operatorname{ir} r_{-} \operatorname{ord}(\preceq, s)(R) \wedge \text { an_proj_del}(p)(R), \\
& Q_{l}:=l \neq-\infty \wedge \exists \xi \in \operatorname{irExpr}(p, s) \cdot\left(\xi \preceq^{t} l \wedge \text { well_def }(\xi, s)(R)\right), \\
& Q_{l \_ \text {other }}:=l \neq-\infty \wedge \exists \xi^{\prime} \in \operatorname{dom}(\preceq) \backslash \operatorname{irExpr}(p, s) .\left(\xi^{\prime} \preceq^{t} l \wedge \text { well_def }\left(\xi^{\prime}, s\right)(R) \wedge \text { ord_inv }\left(\operatorname{res}_{x_{i}}\left(p, \xi^{\prime} \cdot p\right)\right)(R)\right) \text {, } \\
& Q_{u}:=u \neq \infty \wedge \exists \xi \in \operatorname{irExpr}(p, s) .\left(u \preceq^{t} \xi \wedge \text { well_def }(\xi, s)(R)\right) \text {, and } \\
& Q_{u \_ \text {other }}:=u \neq \infty \wedge \exists \xi^{\prime} \in \operatorname{dom}(\preceq) \backslash \operatorname{irExpr}(p, s) .\left(u \preceq^{t} \xi^{\prime} \wedge \text { well_def }\left(\xi^{\prime}, s\right)(R) \wedge \text { ord_inv }\left(\operatorname{res}_{x_{i}}\left(p, \xi^{\prime} \cdot p\right)\right)(R)\right) \text {. }
\end{aligned}
$$


(a) The indexed root ordering ensures that real root functions remain outside the cell. In the case where the green polynomial is delineable, this is sufficient to prove sign invariance.

(b) In the case where the green polynomial has a " $\theta_{*}$ " root function over the underlying cell that disappears and reappears, the indexed root ordering does not say anything about " $\theta$ *" on other connected components of its domain. For preventing them to cross the cell's boundary (e.g. the upper right root function), we might need to add the leading coefficients to the projection.

$x_{1}$
(c) However, if we know that there are well-defined root functions of the green polynomial between the cell's boundaries and " $\theta_{*}$ ", the leading coefficient is not needed, as by projective delineability, these intermediate root functions already protect the cell.

(d) Alternatively, an additional resultant with a polynomial with a (well-defined) root function below or above the cell can be added to prevent " $\theta$ *" root functions from crossing the cell's boundary. As this is expensive, this rule is only applied when this additional resultant is added anyway for maintaining the indexed root ordering, for instance if the cell's lower and upper bounds are defined by the same polynomial.

Figure 8: Different cases for maintaining the sign-invariance of a polynomial (graphed here in green) in a cell (here defined by the enclosed region between the orange and magenta polynomial graphs).

$$
\left.\begin{array}{rlrl}
l & =u \wedge p(s, b(s)) & =0, Q &
\end{array}\right) \text { sgn_inv }(p)(R)
$$

We note that this rule hides some of the complexity for an efficient decision: in the $Q_{l}$ and $Q_{u}$ cases, it is wise to choose the closest root to the lower respectively upper bound, as if there is more than one bound below/above the cell, their root functions are well defined on the underlying cell without adding the leading coefficient. The $Q_{l_{\text {_other }}}$ and $Q_{u \_o t h e r ~}$ cases do not add any extra properties if resultants are picked that are added due to the indexed root ordering $\preceq$ anyway.

### 5.8. Connectedness

For maintaining properties (in particular the order-invariance of some polynomials) on higher levels, we need to maintain connectedness. Besides some preconditions, i.e. that the cell's boundaries are described by two welldefined real root functions, we need to ensure that the lower and upper bound of a sector do not cross as illustrated in Figure 9.

Rule 5.12. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i-1}$, I be a symbolic interval of level $i$, $\preceq$ be an indexed root ordering of level $i$, and $\preceq^{t}$ be the reflexive and transitive closure of $\preceq$. Assume that $\preceq$ matches s.

Let $Q:=a n \_s u b(i-1)(R) \wedge$ connected $(i-1)(R) \wedge \operatorname{repr}(I, s)(R)$.

$$
\begin{aligned}
& \vdash \operatorname{connected}(0)\left(\mathbb{R}^{0}\right) \\
& \vdash \operatorname{connected}(i)(R) \\
& \vdash \operatorname{connected}(i)(R) \\
& \vdash \operatorname{connected}(i)(R)
\end{aligned}
$$

$Q, \mathrm{I}=($ sector $, l, u), l \neq-\infty, u \neq \infty, l . p \neq u \cdot p, l \preceq^{t} u, \quad \operatorname{ir} r_{-} \operatorname{ord}(\preceq, s)$
$Q, \mathrm{I}=($ sector $, l, u), l=-\infty \vee u=\infty \vee l \cdot p=u \cdot p$

### 5.9. Generalization of the current sample

We define a mapping for generalizing the sample to the cell on the current level.


Figure 9: Let $\mathrm{I}=\left(\right.$ sector $\left., \xi_{1}, \xi_{2}\right)$, then $\operatorname{repr}\left(\mathrm{I}, s_{1}\right)$ holds in the whole blue area, thus forming a non-connected subset of $\mathbb{R}^{2}$. By adding $\operatorname{res}_{x_{2}}\left(\xi_{1} \cdot p, \xi_{2} \cdot p\right)$ to the projection, the subset is made connected by restricting the underlying cell as indicated by the dashed vertical lines.

Rule 5.13. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}, s \in \mathbb{R}^{i}$, and I be a symbolic interval of level $i$. Assume that $s_{i} \in \operatorname{setOf}\left(s_{[i-1]}, \mathrm{I}\right)$.

$$
\operatorname{sample}\left(s_{[i-1]}\right)(R), \operatorname{repr}\left(\mathrm{I}, s_{[i-1]}\right)(R) \quad \vdash \operatorname{sample}(s)(R)
$$

### 5.10. Cell descriptions

Now, we define a property that states that the cell is described by its representation.
Property 5.5. Let $i \in \mathbb{N}_{>0}, R \subseteq \mathbb{R}^{i}$, and I be a symbolic interval of level $i$.
The property holds(I) holds on $R$ if and only if $R=\operatorname{set} O f\left(R \downarrow_{[i-1]}, \mathrm{I}\right)$.
This property is the only one that cannot be mapped to a set of other properties, i.e. properties of this kind are the assumptions in our proof system.

Given holds(I), we can maintain the weaker property $\operatorname{repr}(\mathrm{I}, s)$. To do so, we need to ensure that the indexed root expressions describing the cell's boundaries always refers to the same root on the underlying cell by making their defining polynomials delineable (such that the referred real root function is defined and the number of roots below the referred root function is constant over the underlying cell).

Rule 5.14. Let $i \in \mathbb{N}_{>0}$, $R \subseteq \mathbb{R}^{i-1}$, $s \in \mathbb{R}^{i-1}$, and I be a symbolic interval of level $i$. Assume that $\mathrm{I} . l \in$ $\operatorname{ir} \operatorname{Expr}(\mathrm{I} . l . p, s)($ if I. $l \neq-\infty)$, I. $u \in \operatorname{ir} \operatorname{Expr}(\mathrm{I} . u . p, s)($ if I. $u \neq \infty)$ respectively I. $b \in \operatorname{irExpr}(\mathrm{I} . b . p, s)$.

Let $Q_{\text {delineable }}(q):=$ an_proj_del $(q)(R) \wedge \operatorname{sgn\_ inv}\left(\operatorname{ldcf}_{x_{i+1}}(q)\right)(R)$.

$$
\begin{array}{ll}
\operatorname{sample}(s)(R), \text { holds }(\mathrm{I})(R), \mathrm{I}=(\text { section }, b), Q_{\text {delineable }}(b . p) & \vdash \operatorname{repr}(\mathrm{I}, s)(R) \\
\operatorname{sample}(s)(R), \text { holds }(\mathrm{I})(R), \mathrm{I}=(\operatorname{sector}, l, u), l=-\infty \vee Q_{\text {delineable }}(l . p), u=\infty \vee Q_{\text {delineable }}(u . p) & \vdash \operatorname{repr}(\mathrm{I}, s)(R)
\end{array}
$$

Note that $\operatorname{repr}(\mathrm{I}, s)$ could be maintained by a weaker property, namely that the roots defining the cell boundaries are well-defined and some analogous guarantee as given by holds(I). For the purpose of describing single cells, this is too weak; however, for other scenarios, this could avoid some more leading coefficients in the projection.

### 5.11. Ordering of properties

We observe that the rules of inference defined in Rules 5.1 to 5.14 are cycle-free in the sense that all properties in the antecedents are smaller than the consequent property according to some ordering $\triangleleft$ :

Definition 5.4. The ordering $\triangleleft i$ defined such that properties of level $i$ are greater than any property of level $i-1$ w.r.t. $\triangleleft$ and such that it satisfies the following partial order of properties of each level $i$ (starting with the greatest element):

1. $\operatorname{ir} \_$ord $(\preceq, s)$ for all $\preceq$ for roots of level $i$ and $s \in \mathbb{R}^{i-1}$
2. well_def $(\xi, s)$ for all $\xi$ of level $i$ and $s \in \mathbb{R}^{i-1}$
3. an_proj_del( $p$ ) for all $p$ of level $i$
4. non_null( $(p)$ for all $p$ of level $i$


Figure 10: Proof rule relationships; irred stands for irreducible.
5. ord_inv $(p)$ for all reducible $p$ of level $i$
6. ord_inv $(p)$ for all irreducible $p$ of level $i$
7. sgn_inv( $p$ ) for all reducible $p$ of level $i$
8. sgn_inv $(p)$ for all irreducible $p$ of level $i$
9. connected $(i)$
10. an_sub(i)
11. sample(s) for all $s \in \mathbb{R}^{i}$ of level $i$
12. $\operatorname{repr}(\mathrm{I}, s)$ for all I of level $i$ and $s \in \mathbb{R}^{i-1}$
13. holds(I) for all I of level $i$

The proof rules introduced in this section are visualised in Figure 10 which shows the relationships between rules of different levels.

## 6. Levelwise construction of a single cell

So far, we presented an abstract proof system, i.e. a set of rules which allow us to construct a proof for signinvariance of some polynomials in a cell. In the following, we will give an algorithm for constructing a single cell in a levelwise manner by specifying how the proof rules are applied. We first give the algorithmic framework and then define heuristics needed to instantiate the algorithm.

### 6.1. Algorithm to construct single cell

Algorithm 1 constructs a single cell. The input to the algorithm is a set of polynomials for which a cell is to be computed on which these polynomials are sign-invariant. This initialises the set of properties $Q$ on Line 1 which is maintained to contain properties that need to be fulfilled by the yet to be chosen representations on the lower levels. I.e. at the beginning of iteration $i$, the data structure $\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{i}\right)$ needs to maintain all properties in $Q$. To do so, Algorithm 1 iteratively calls Algorithm 2 to process the properties and construct the interval on level $i$, taking note of any failure cases due to nullifications that might occur.

Algorithm 3 is a sub-algorithm that replaces a property in $Q$ by a set of properties $Q^{\prime}$ induced by a proof rule if no such set is already contained in $Q$, and otherwise simply removes the property from $Q$. FAIL is returned here if and only if no proof rule is applicable; which is the case only when a polynomial is nullified (i.e. Rule 5.2) and equational constraint projection does not allow us to ignore that fact. The sets $Q^{\prime}$ induced by a proof rule are determined in Line 1 as follows. Assume we want to derive the property non_null( $p$ ) of $R \subseteq \mathbb{R}^{i}$ for some polynomial $p$ of level $i+1$ with respect to the current sample $s \in \mathbb{R}^{i}$. The algorithm would check if the conditions of the proof rules in Rule 5.2 are satisfied, and then determine the corresponding properties of $R$ which form the set $Q^{\prime}$. In the first case, we check whether $p$ is irreducible, $\operatorname{deg}_{x_{i+1}}(p)>1$, and $\operatorname{disc}_{x_{i+1}}(p)(s) \neq 0$ hold; if yes, we add the set $Q^{\prime}=\left\{\operatorname{sample}(s), \operatorname{sgn} \operatorname{inv}\left(\operatorname{disc}_{x_{i+1}}(p)\right)\right\}$ to the set of choices. In the second case, we check whether $p$ is irreducible holds and there exists coefficient $c_{j}$ of $p$ such that $c_{j}(s) \neq 0$ holds; if yes, we add the set $Q^{\prime}=\left\{\right.$ sample $(s)$, sgn_inv $\left.\left(c_{j}\right)\right\}$ to the set of choices.

Algorithm 2 is used to extend the cell construction by a level. It applies the proof rules according to the ordering $\triangleleft$ on the properties from Definition 5.4 until the only properties of level $i$ remaining are the sign-invariance of some irreducible polynomials (note that all smaller properties w.r.t. $\triangleleft$ are provable with only the sample $s$ present except if a nullification occurs). Then, a representation consisting of a symbolic interval I, a set of polynomials $E$ and an indexed root ordering $\preceq$ is chosen. The roles of I and $\preceq$ are already discussed above; the polynomials in $E$ are meant to be excluded from the indexed root ordering, and thus the application of the equational constraint projection rule (Rule 5.8) is enforced. The formal requirements on the representation will be defined in Definition 6.1. Finally, the remaining properties on level $i$ are eliminated by application of the proof rules, until only a property of the form $h o l d s(\mathrm{I})$ is left on the level (which is the greatest property of the current level w.r.t. $\triangleleft$ ).

Theorem 6.1. Let $P \subseteq \mathbb{Q}\left[x_{1}, \ldots, x_{n}\right]$, and $s \in \mathbb{R}^{n}$. The method single_cell in Algorithm 1 either returns a cell data structure $R$ such that all polynomials in $P$ are sign-invariant on $\operatorname{set} O f(R)$ and $s \in \operatorname{set} O f(R)$, or returns FAIL.

Proof. Note that the algorithm will terminate as Algorithm 3 replaces properties by smaller properties according to the ordering as defined in Definition 5.4, there does exist a smallest property, and the CAD projection of a set of polynomials is finite. The correctness follows from the correctness proofs of the rules of inference.

### 6.2. Heuristic choices

Our algorithm has two points where a choice must be made: (1) when applying a rule of inference in Algorithm 3 Line 5 and (2) choosing the representation in Algorithm 2 Line 7.

For the first choice, the rules of inference may define multiple possible property sets for a property that each imply that property and so we have the freedom to choose which to use. We aim to pick the "best" such set, noting that this choice is heuristic (made for example with respect to estimated computational effort, or degrees of polynomials, or number of polynomials / real roots). One could compute each of these sets individually and compare them, or, to avoid superfluous heavy computations of resultants or discriminants, prefer sets which seem easier to compute (which is the approach taken by our implementation). For instance, in Rule 5.2, we always pick a coefficient rather than computing a discriminant. Note that even if sets involving resultants or discriminants are avoided where possible, the algorithm should first check whether one such set is already maintained before computing any new set.

For the second choice, we formalize what we mean by a representation. As already discussed, there might be multiple choices for the cell description I on the current level and the indexed root ordering $\preceq$ for fulfilling the requirements for Rule 5.11. Additionally, if I is a section, we can make use of the equational constraints projection in Rule 5.8. A representation determines all these parameters.

Definition 6.1 (Representation for $\boldsymbol{\Xi})$. Let $i \in \mathbb{N}_{>0}$, $s \in \mathbb{R}^{i}$, and $\Xi$ be a set of indexed root expressions of level $i$ such that $\xi \in \operatorname{ir} \operatorname{Expr}\left(\xi \cdot p, s_{[i-1]}\right)$ for every $\xi \in \Xi$.
$A$ representation for $\Xi$ with respect to $s$ is a tuple ( $\mathrm{I}, E, \preceq$ ) where I is a symbolic interval of level $i, E$ is a set of polynomials of level $i$, and $\preceq$ is an indexed root ordering of level $i$ such that

- $s_{i} \in \operatorname{set} O f\left(s_{[i-1]}, \mathrm{I}\right)$,

```
Algorithm 1: single_cell \((P, s)\)
    Input : finite \(P \subseteq \mathbb{Q}\left[x_{1}, \ldots, x_{n}\right], s \in \mathbb{R}^{n}\)
    Output: single cell data structure R such that \(s \in \operatorname{set} O f(\mathrm{R})\) and all polynomials in \(P\) are sign-invariant on
                    setOf(R); or FAIL
    \(Q:=\left\{\operatorname{sgn\_ inv}(p) \mid p \in P\right\}\)
    for \(i=n, \ldots, 1\) do
        result \(:=\) construct_interval \((Q, s) / /\) Algorithm 2
        if result \(\neq F A I L\) then
                \(\mathrm{I}_{i}, Q:=\) result
        else
                return FAIL
    return \(\left(\mathrm{I}_{1}, \ldots, \mathrm{I}_{n}\right)\)
```

```
Algorithm 2: construct_interval \((Q, s)\)
    Input : \(i \in \mathbb{N}_{>0}\), finite \(Q \subseteq \operatorname{Prop}_{[i]}, s \in \mathbb{R}^{i}\)
    Output: an interval data structure I such that \(\left.s \in \operatorname{setOf(s} s_{[i-1]}, \mathrm{I}\right)\), and a set \(\left.Q^{\prime} \subseteq \operatorname{Prop}\right|_{[i-1]}\) such that for
                any \(R \subseteq \mathbb{R}^{i-1}\) it holds \(\left(\forall q^{\prime} \in Q^{\prime} . q^{\prime}(R)\right) \Longrightarrow(\forall q \in Q . q(\operatorname{setOf}(R, \mathrm{I})))\); or \(F A I L\)
    foreach \(\left.q \in Q\right|_{i}\) where \(q\) is the greatest element with respect to \(\triangleleft\) (from Definition 5.4) and \(q \neq \operatorname{sgn} \_\)inv \((p)\)
    for an irreducible \(p\) do
        \(Q:=\) apply_pre \((Q, q,(s)) / /\) Algorithm 3
        if \(Q=\) FAIL then
            return FAIL
    \(P_{\text {nonnull }}:=\left\{p\left|\operatorname{sgn\_ inv}(p) \in Q\right|_{i}\right.\) s.t. \(\left.p\left(s_{[i-1]}, x_{i}\right) \neq 0\right\}\)
    \(\Xi:=\operatorname{irExpr}\left(P_{\text {nonnull }}, s_{[i-1]}\right)\)
    choose representation ( \(\mathrm{I}, E, \preceq\) ) of \(\Xi\) with respect to \(s\)
    if \(i>1\) then
        foreach \(\left.q \in Q\right|_{i}\) where \(q\) is the greatest element with respect to \(\triangleleft\) (from Definition 5.4) and
            \(q \neq\) holds(I) do
                \(Q:=\) apply_pre( \(Q, q,(s, \mathrm{I}, \preceq)\) ) // Algorithm 3
                if \(Q=F A I L\) then
                    return FAIL
    /* \(\left.Q\right|_{i}\) contains now only holds(I) */
    return \(\mathrm{I},\left(\left.Q \backslash Q\right|_{i}\right)\)
```

```
Algorithm 3: apply_pre( \(Q, q\), parameters)
    Input : \(i \in \mathbb{N}_{>0}\), finite \(\left.Q \subseteq \operatorname{Prop}\right|_{[i]},\left.q \in \operatorname{Prop}\right|_{i} \cap Q\) and either parameters \(=(s)\) or
                parameters \(=(s, \mathrm{I}, \preceq)\) with \(s \in \mathbb{R}^{i}\), symbolic interval I on level \(i\), indexed root ordering \(\preceq\) on
                level \(i\)
    Output: a set \(\left.(Q \backslash\{q\}) \cup Q^{\prime} \subseteq \operatorname{Prop}\right|_{[i]} \backslash\{q\}\) such that for all \(R \subseteq \mathbb{R}^{i}\) it holds \(\left(\forall q^{\prime} \in Q^{\prime} . q^{\prime}(R)\right) \Longrightarrow q(R)\);
                        or \(F A I L\)
    let choices \(\subseteq 2^{\text {Prop }}\) such that each \(Q^{\prime} \in\) choices is a set of properties from a proof rule which can be
    applied according to parameters to derive \(q\) (see explanatory text above for details)
    if choices \(=\emptyset\) then
        return FAIL
    if \(\nexists Q^{\prime} \in\) choices. \(Q^{\prime} \subseteq Q\) then
        choose \(Q^{\prime} \in\) choices
        return \((Q \backslash\{q\}) \cup Q^{\prime}\)
    else
        return \(Q \backslash\{q\}\)
```

- either $p \in E$ or $\operatorname{irExpr}\left(p, s_{[i-1]}\right) \subseteq \operatorname{dom}(\preceq)$ for all $p \in\{\xi \cdot p \mid \xi \in \Xi\}$,
- if $E \neq \emptyset$ then $\mathrm{I}=($ section, $b)$ for some indexed root expression $b$,
- $\preceq$ matches $s_{[i-1]}$, and
- $\xi \preceq^{t}$ I.l or I. $u \preceq^{t} \xi$ for all $\xi \in \Xi$ if I $=$ (sector, $l$, $u$ ) respectively $\xi \preceq^{t}$ I.b or I.b $\preceq^{t} \xi$ for all $\xi \in \Xi$ if $\mathrm{I}=($ section,$b)$.

This definition is quite general by allowing additional indexed root expressions in $\Xi^{\prime}$ compared to the set $\Xi$ describing the roots of the present polynomials. This for example enables heuristics to under-approximate the constructed cell to reduce heavy computations in future work (see Section 9.1).

In the following however, we will assume that $\Xi^{\prime}=\Xi$, that is, I and $\preceq$ will only consider roots from $\Xi$ corresponding to the polynomials for which we need to derive sign-invariance.

Note that we omit the requirements of Rule 5.12 here yet for readability reasons and as the adaption is trivial: If required, we only need to add the pair (I.l, I.u) to the indexed root ordering.

For the symbolic interval I, we minimize the degrees in the main variable of the defining polynomials.
Definition 6.2 (Choice of the symbolic interval). Let $i \in \mathbb{N}_{>0}, P \subseteq \mathbb{Q}\left[x_{1}, \ldots, x_{i}\right]$ be a set of irreducible polynomials of level $i, s \in \mathbb{R}^{i}$ be a sample such that no $p \in P$ is nullified on $s_{[i-1]}$, and $\Xi \subseteq \operatorname{irExpr}\left(P, s_{[i-1]}\right)$.

We define the sets $\Xi_{l o}$ and $\Xi_{\text {up }}$ of the closest lower respectively upper bounds of $s_{i}$ as follows:

$$
\begin{aligned}
& \Xi_{l o}=\left\{\xi \in \Xi \mid \xi\left(s_{[i-1]}\right) \leq s_{i} \wedge \forall \xi^{\prime} \in \Xi .\left(\xi^{\prime}\left(s_{[i-1]}\right) \leq s_{i} \Longrightarrow \xi^{\prime}\left(s_{[i-1]}\right) \leq \xi\left(s_{[i-1]}\right)\right)\right\}, \\
& \Xi_{u p}=\left\{\xi \in \Xi \mid s_{i} \leq \xi\left(s_{[i-1]}\right) \wedge \forall \xi^{\prime} \in \Xi .\left(s_{i} \leq \xi^{\prime}\left(s_{[i-1]}\right) \Longrightarrow \xi\left(s_{[i-1]}\right) \leq \xi^{\prime}\left(s_{[i-1]}\right)\right)\right\}
\end{aligned}
$$

We pick $\xi_{l o} \in \Xi_{l o}$ such that $\operatorname{deg}_{x_{i}}\left(\xi_{l o} . p\right)$ is minimal and $\xi_{u p} \in \Xi_{u p}$ such that $\operatorname{deg}_{x_{i}}\left(\xi_{u p} \cdot p\right)$ is minimal. Additionally, if $\Xi_{l o}=\Xi_{u p}$, then we require $\xi_{l o}=\xi_{u p}$.

We define the lowest degree interval $\mathrm{I}_{\text {ldeg }}$ of $s$ with respect to $\Xi$ as

$$
\begin{array}{rr}
\mathrm{I}_{\text {ldeg }}=\left(\text { section }, \xi_{l o}\right) & \text { if } \xi_{l o}=\xi_{\text {up }}, \\
\mathrm{I}_{\text {ldeg }}=\left(\text { sector }, \xi_{l o}, \xi_{\text {up }}\right) & \text { otherwise } .
\end{array}
$$

For the indexed root ordering $\preceq$, there are two possibilities: aim to make the underlying cell as big as possible; or aim to avoid heavy resultant computations. In theory we could compute the results for all (or all promising) possible indexed root orderings and pick the best one. However, as this is infeasible in practise, we define below several alternative heuristics with different rationales.

To achieve this we employ a somewhat idealistic view on the problem. Recall the property that an indexed root ordering holds is ensured by making resultants order-invariant, which is often a stronger ordering on the roots than required by the picked indexed root ordering. We ignore this fact and base our heuristics on the set of indexed roots without considering common defining polynomials between them (for now). This is further discussed in Section 7.

We start with the following observation: as the given derivation rules always require projective delineability for all polynomials whose sign-invariance is proven using an indexed root ordering, a fixed ordering of all real root functions defined at $s_{[i-1]}$ of such polynomials is guaranteed anyway. Thus, we can restrict the heuristics for the choice of $\preceq$ by computing the ordering on the set $\tilde{\Xi}$ containing for each polynomial only the closest lower and upper roots to $s_{i}$, and extending an ordering $\tilde{\preceq}$ of $\tilde{\Xi}$ to an ordering $\preceq$ on $\Xi$. This is formalized in the following definition.

Definition 6.3 (Choice of the indexed root ordering: reduction). Let $i, P, s, \Xi$ be as in Definition 6.2. Then

$$
\begin{aligned}
\tilde{\Xi}= & \left\{\xi \in \Xi \mid \xi\left(s_{[i-1]}\right) \leq s_{i} \wedge \forall \xi^{\prime} \in \Xi \backslash\{\xi\} \cdot\left(\xi^{\prime} \cdot p=\xi \cdot p \Longrightarrow \neg\left(\xi\left(s_{[i-1]}\right)<\xi^{\prime}\left(s_{[i-1]}\right) \leq s_{i}\right)\right)\right\} \\
& \cup\left\{\xi \in \Xi \mid s_{i} \leq \xi\left(s_{[i-1]}\right) \wedge \forall \xi^{\prime} \in \Xi \backslash\{\xi\} .\left(\xi^{\prime} \cdot p=\xi \cdot p \Longrightarrow \neg\left(s_{i} \leq \xi^{\prime}\left(s_{[i-1]}\right)<\xi\left(s_{[i-1]}\right)\right)\right)\right\} .
\end{aligned}
$$

For any indexed root ordering $\tilde{\preceq}_{X}$ on $\tilde{\Xi}$ matching $s_{[i-1]}$, we define the ordering $\preceq_{X}$ on $\Xi$ matching $s_{[i-1]}$ as

$$
\left.\preceq_{X}=\tilde{\preceq}_{X} \cup\left\{\left(\xi, \xi^{\prime}\right) \mid \xi, \xi^{\prime} \in \Xi \text { such that } \xi \cdot p=\xi^{\prime} \cdot p \wedge \xi\left(s_{[i-1]}\right)<\xi^{\prime}\left(s_{[i-1]}\right)\right)\right\} .
$$

In the following definitions we give different heuristics to choose an indeed root ordering $\tilde{\Omega}_{X}$ which in each case can be extended to a corresponding ordering $\preceq_{X}$.

First, in the section case, the below EQUATIONAL CONSTRAINT heuristic enforces simply the application of the equational constraint rule to all polynomials.

Definition 6.4 (EQUATIONAL CONSTRAINT representation). Let $i, P, s, \Xi, \xi_{l o}, \xi_{u p}, \mathrm{I}_{\text {ldeg }}$ be as in Definition 6.2. If $\xi_{l o}=\xi_{u p}$, we define the EqUational constraint representation as the tuple $\left(\mathrm{I}_{\text {ldeg }}, P, \emptyset\right)$.

Next, the BIGGEST CELL heuristic defines the weakest ordering on the indexed roots according to Definition 6.1 and thus defines the biggest possible underlying cell (under the assumption that this ordering can be realized perfectly, i.e. the resultants have roots only below the crossing of a polynomials' root with a cell boundary) as visualized in Figure 11a. Note that for the section case, the biggest cell heuristic is the EQuational constraint heuristic plus some discriminants and coefficients; thus, the application of BIGGEST CELL only makes sense in the sector case.

Definition 6.5 (BigGest cell representation). Let $i, P, s, \Xi, \xi_{l o}, \xi_{u p}, \mathrm{I}_{\text {ldeg }}$ be as in Definition 6.2.
We define the indexed root ordering $\preceq_{\text {biggest }}$ on $\Xi$ according to

$$
\begin{aligned}
\tilde{\preceq}_{\text {biggest }}= & \left\{\left(\xi, \xi_{l o}\right) \mid \xi \in \tilde{\Xi} \backslash\left\{\xi_{l o}\right\} \text { s.t. } \xi\left(s_{[i-1]}\right) \leq \xi_{l o}\left(s_{[i-1]}\right)\right\} \\
& \cup\left\{\left(\xi_{u p}, \xi\right) \mid \xi \in \tilde{\Xi} \backslash\left\{\xi_{u p}\right\} \text { s.t. } \xi_{u p}\left(s_{[i-1]}\right) \leq \xi\left(s_{[i-1]}\right)\right\}
\end{aligned}
$$

and the BIGGEST CELL representation as the tuple ( $\mathrm{I}_{\text {ldeg }}, \emptyset, \preceq_{\text {biggest }}$ ).
The below LOWEST DEGREE BARRIERS heuristic minimizes the degrees of the defining polynomials (locally per level), and thus also the degree of the computed resultants (under the above assumption) as visualized in Figure 11b. Furthermore, it enforces the equational constraints rule whenever possible.

This heuristic has two motivations. First recall that the polynomial degrees grow doubly exponentially during the CAD projection, see e.g. [5, Table 1]. Second, that the running time of the resultant computation depends quadratically on the degree in the main variable of the input polynomials, see [22].

In the following definition, we use the lexicographical ordering on tuples, that is $t=\left(t_{1}, \ldots, t_{k}\right)<\left(t_{1}^{\prime}, \ldots, t_{k}^{\prime}\right)=t^{\prime}$ holds if $t_{1}<t_{1}^{\prime} \vee\left(t_{1}=t_{1}^{\prime} \wedge t_{2}<t_{2}^{\prime}\right) \vee\left(t_{1}=t_{1}^{\prime} \wedge t_{2}=t_{2}^{\prime} \wedge t_{3}<t_{3}^{\prime}\right) \vee \ldots$

Definition 6.6 (Lowest DEGREE BARRIERS representation). Let $i, P, s, \Xi, \xi_{l o}, \xi_{u p}, \mathrm{I}_{l d e g}$ be as in Definition 6.2.
We assume an injection $o: \Xi \rightarrow \mathbb{N}$ that orders the elements of $\Xi$ such that $o\left(\mathrm{I}_{\text {ldeg }} \cdot b\right)=0$ respectively $o\left(\mathrm{I}_{\text {ldeg. }} \cdot l\right)=$ 0 , o( $\left.\mathrm{I}_{\text {ldeg }} \cdot u\right)=1$ (if both are indexed root expressions), or o $\left(\mathrm{I}_{\text {ldeg }} \cdot l\right)=0$ (if $\mathrm{I}_{\text {ldeg }} \cdot u=\infty$ ), or o $\left(\mathrm{I}_{\text {ldeg }} \cdot u\right)=0$ (if $\left.\mathrm{I}_{\text {ldeg. }} l=-\infty\right)$.

Let $\Xi^{\prime} \subseteq \Xi$. For $\xi \in \Xi^{\prime}$, we define the barrier of $\xi$ w.r.t. $\Xi^{\prime}$, that is a root in $\Xi^{\prime}$ between $\xi\left(s_{[i-1]}\right)$ and $s_{i}$ with minimal degree in the main variable:

$$
\begin{array}{ll}
\text { if } \xi\left(s_{[i-1]}\right) \leq s_{i} & \text { then barrier } \Xi_{\Xi^{\prime}}(\xi)=\underset{\left\{\xi^{\prime} \in \Xi^{\prime} \mid \xi\left(s_{[i-1]}\right) \leq \xi^{\prime}\left(s_{[i-1]}\right) \leq s_{i}\right\}}{\arg \min ^{\arg }}\left(\operatorname{deg}_{x_{i}}\left(\xi^{\prime} \cdot p\right), s_{i}-\xi^{\prime}\left(s_{[i-1]}\right), o\left(\xi^{\prime}\right)\right) ; \\
\text { if } s_{i} \leq \xi\left(s_{[i-1]}\right) & \text { then barrier } \\
\left\{\xi^{\prime}(\xi)=\underset{\left.\xi^{\prime} \in \Xi^{\prime} \mid s_{i} \leq \xi^{\prime}\left(s_{[i-1]}\right) \leq \xi\left(s_{[i-1]}\right)\right\}}{\arg \min }\left(\operatorname{deg}_{x_{i}}\left(\xi^{\prime} \cdot p\right), \xi^{\prime}\left(s_{[i-1]}\right)-s_{i}, o\left(\xi^{\prime}\right)\right) .\right.
\end{array}
$$

$$
\begin{aligned}
& \text { If } \xi_{l o} \neq \xi_{\text {up }} \text {, we define the indexed root ordering } \preceq_{\text {barriers }} \text { on } \Xi \text { according to } \\
& \qquad \begin{aligned}
\underline{\Omega}_{\text {barriers }} & =\left\{\left(\xi, \text { barrier }_{\tilde{\Xi}}(\xi)\right) \mid \xi \in \tilde{\Xi} \text { s.t. } \xi\left(s_{[i-1]}\right)<s_{i} \text { and } \xi \neq \operatorname{barrier}_{\tilde{\Xi}}(\xi)\right\}, \\
& \cup\left\{\left(\operatorname{barrier}_{\tilde{\Xi}}(\xi), \xi\right) \mid \xi \in \tilde{\Xi} \text { s.t. } s_{i}<\xi\left(s_{[i-1]}\right) \text { and } \xi \neq \text { barrier }_{\tilde{\Xi}}(\xi)\right\}
\end{aligned}
\end{aligned}
$$

and the LOWEST DEGREE BARRIERS representation as the tuple ( $\mathrm{I}_{\text {ldeg }}, \emptyset, \preceq_{\text {barriers }}$ ).
If $\xi_{l o}=\xi_{u p}$, we exclude the polynomials with roots around $s_{i}$ which do not qualify as a barrier for some other roots from the indexed root ordering (thus enforcing the application of the equational constraint rule). To do so, for a set of polynomials $P^{\prime} \subseteq P$, we define the set $\left.\tilde{\Xi}\right|_{P^{\prime}}=\left\{\xi \mid \xi \in \tilde{\Xi}\right.$ s.t. $\left.\xi . p \in P^{\prime}\right\}$ and let $P_{e q} \subseteq P$ be the result of the following fixed point computation (i.e. $P_{e q}=P_{j}=P_{j+1}$ for some $j$ ):

$$
\begin{aligned}
P_{0} & =\emptyset \\
P_{j+1} & =P_{j} \cup\left\{p \in P \mid p \neq \mathrm{I}_{l \text { deg }} \cdot b \cdot p\right. \\
& \left.\left.\wedge \exists \xi \in \tilde{\Xi} \backslash \tilde{\Xi}\right|_{P_{j}} .\left(\xi \cdot p=p \wedge \operatorname{barrier}_{\left.\tilde{\Xi} \backslash \tilde{\Xi}\right|_{P_{j}}}(\xi)=\left.\mathrm{I}_{\text {ldeg }} \cdot b \wedge \nexists \xi^{\prime} \in \tilde{\Xi} \backslash \tilde{\Xi}\right|_{P_{j}} . b \operatorname{barrier}_{\left.\tilde{\Xi} \backslash \tilde{\Xi}\right|_{P_{j}}}\left(\xi^{\prime}\right)=\xi\right)\right\} .
\end{aligned}
$$

We define the indexed root ordering $\preceq_{\text {barriers }}$ on $\left.\Xi \backslash \Xi\right|_{P_{e q}}$ according the definition of $\tilde{\preceq}_{\text {barriers }}$ as above but only considering the roots $\left.\tilde{\Xi} \backslash \tilde{\Xi}\right|_{P_{e q}}$ and the LOWEST DEGREE BARRIERS representation as the tuple $\left(\mathrm{I}_{\text {ldeg }}, P_{\text {eq }}, \preceq_{\text {barriers }}\right)$.


Figure 11: Visualization of indexed root ordering heuristics.

The below chain heuristic fixes the total ordering on the roots as visualized in Figure 11c.
Definition 6.7 (ChAIN representation). Let $i, P, s, \Xi, \xi_{l o}, \xi_{u p}, \mathrm{I}_{\text {ldeg }}$ be as in Definition 6.2.
Let $\left\{\xi_{1}, \ldots, \xi_{k}\right\}=\tilde{\Xi}$ s.t. $\xi_{j}\left(s_{[i-1]}\right) \leq \xi_{j+1}\left(s_{[i-1]}\right)$ for all $j \in[1 . . k-1]$.
We define the indexed root ordering $\preceq$ chain on $\Xi$ according to

$$
\underline{\preceq}_{\text {chain }}=\left\{\left(\xi_{j}, \xi_{j+1}\right) \mid j \in[1 . . k-1]\right\}
$$

and the CHAIN representation as the tuple ( $\mathrm{I}_{\text {ldeg }}, \emptyset, \preceq_{\text {chain }}$ ).
Finally, the FULL heuristic fixes the same total ordering as it is the unique transitive closure of the chain heuristic as visualized in Figure 11d. Note that we include this heuristic only for illustrative purposes: it resembles the cells constructed naively by making a full projection without adapting to the behaviour at the sample.

Definition 6.8 (FULL representation). Let $i, P, s, \Xi, \xi_{l o}, \xi_{u p}, \mathrm{I}_{\text {ldeg }}$ as in Definition 6.2.
Let $\left\{\xi_{1}, \ldots, \xi_{k}\right\}=\tilde{\Xi}$ s.t. $\xi_{j}\left(s_{[i-1]}\right) \leq \xi_{j+1}\left(s_{[i-1]}\right)$ for all $j \in[1 . . k-1]$.
We define the indexed root ordering $\preceq_{\text {full }}$ on $\Xi$ according to

$$
{\underset{\underline{q}}{\text { full }}}=\left\{\left(\xi_{j}, \xi_{j^{\prime}}\right) \mid j, j^{\prime} \in[1 . . k-1], j<j^{\prime}\right\}
$$

and the FULL representation as the tuple ( $\mathrm{I}_{\text {ldeg }}, \emptyset, \preceq_{\text {full }}$ ).

## 7. Observations

### 7.1. Ordering heuristics

In this section we will explain the intuition behind the various ordering heuristics. They are designed to optimise desirable characteristics. We acknowledge their heuristic nature, and in particular that the intuition is made under the idealistic view that any indexed root ordering $\preceq$ can be perfectly realized: that is, the resultants calculated only have roots which indicate a crossing of two root functions considered in $\preceq$ when in reality they also have "spurious" roots which do not have relevance for the problem at hand.

From the FULL to the CHAIN heuristic. As mentioned above, both the FULL and the Chain heuristic fix the same ordering on the real root functions over any cell containing the current sample as depicted in Figure 12a.

Obviously, the chain heuristic is more efficient here as it uses a strict subset of the work done by the full heuristic. Unlike the FULL heuristic, the Chain heuristic takes the underlying sample into account, and thus the resulting projection is only valid for a single cell, i.e. locally delineable. The FULL heuristic is independent from the sample and makes the set of polynomials fully delineable.


Figure 12: Root ordering heuristics; the dotted lines indicate for which pairs of polynomials the resultant is added to the projection.

From the ChAIN to the BIGGEST CELL heuristic. The CHAIN heuristic still fixes a stronger ordering on the root functions than necessary. As defined above, the BIGGEST CELL heuristic is the minimal requirement on the ordering to maintain sign-invariance on the constructed cell. This way, we hope that the size of the underlying cell is maximized, as in Figure 12b. Note that while the CHAIN heuristic only takes the sample in one dimension less into account, the BIGGEST CELL heuristic also considers the highest dimension of the sample.

From the BIGGEST CELL heuristic to the LOWEST DEGREE BARRIERS heuristic. The LOWEST DEGREE BARRIERS heuristic minimizes the degrees of the resultants, as illustrated in Figure 12c. The rationale here is that resultant computations are heavy and their complexity depends on the degrees of the input polynomials, and that polynomials with lower degrees have fewer roots. Thus reducing degrees may reduce the case of resultants having real roots that do not actually correspond to relevant points. Note that despite the naming, the LOWEST DEGREE BARRIERS heuristic could in theory lead to bigger cells than the BIGGEST CELL heuristic in certain cases.

Equational constraint projection. The equational constraint projection allows us to leave out discriminants and coefficients of all polynomials except the section-defining polynomial by only adding resultants of all polynomials with the defining polynomial. Thus, we expect this rule to be more efficient than the presented heuristics in most cases. However, we added the possibility for the application of heuristics for cases where the section-defining polynomial has high degree and thus computing resultants with that polynomial is desirable to be avoided.

On the gap between the idealised view and reality. For the LOWEST DEGREE BARRIERS heuristic, this might lead to the computation of a redundant set of resultants regarding the minimal requirement on the indexed root ordering, as shown in Figure 13, where the relation between the first roots of $p_{2}$ and $p_{3}$ (depicted in red) is superfluous but its corresponding resultant is added because all roots are considered individually and not their connection via the defining polynomials.

### 7.2. Comparison with the refinement-based approach

The refinement-based approach to single cell construction [11] saves resultants compared to full CAD projection in the same way as our levelwise variant, by exploiting the transitivity of the induced ordering on real root functions. Complexity-wise, the approaches are the same, as both add up to two resultants per polynomial on a level.

However, the influence on the ordering of constraints on the refinement-based variant means the quality of the constructed cell varies. In the worst case, the resulting ordering corresponds to the chain heuristic: in Figure 12a, this is achieved for the example when the polynomials are merged in the ordering $p_{4}, p_{3}, p_{2}, p_{1}$. Then, the upper cell boundary is updated in every step to a lower boundary. In the best case, the BIGGEST CELL heuristic is achieved: for our example in Figure 12b, this is when the polynomial $p_{1}$ defining the sector's boundary is merged first.


Figure 13: Redundancies in the Lowest Degree barriers heuristic

For the section case, it should also be noted that the levelwise approach can always apply the equational constraints rule as the cell description is known before the projection. The refinement-based approach only starts applying the equational constraints rule when the cell collapses to a section, until then it adds discriminants of all polynomials, which may not actually be needed. The illustrating examples may be used to explain this observation. Consider Figure 12a but with the sample $s$ moved to the upper root of $p_{1}$. When merging polynomials in the ordering $p_{4}, p_{3}, p_{2}, p_{1}$, the polynomials $p_{4}, p_{3}, p_{2}$ are merged as in the sector case until the cell collapses to a section when merging $p_{1}$. When merging $p_{1}$ first, then the section case is identified directly and the reduced projection applied when merging $p_{4}, p_{3}, p_{2}$, meaning the discriminants and coefficients for those polynomials need not be added.

### 7.3. Potential for non-connected "cell" descriptions

Recall that we aim to compute a single cell on which the input polynomials are sign-invariant, but McCallum's CAD projection theory uses the stronger property of order-invariance. It has been observed before that this can allow for small optimisations at the top layer. For example, when using equational constraint projection we must take discriminants if the projection is in a middle layer (to ensure order-invariance is provided which is a hypothesis of the next lifting) but can avoid these at the top layer as we need never lift over that [23].

The visualisation of our proof rules in Figure 10 alerted us to another such optimisation, which if enacted has some strange consequences. Note that a cell being connected at level $i$ is required for it to be order-invariant, but not sign-invariant. Thus we need not ensure connectivity at the top level, i.e. we could avoid taking resultants of the upper and lower bound polynomials to satisfy Rule 5.12. Without connectedness it would not be accurate to describe what we are constructing as a cell, rather it is a semi-algebraic set (see also Figure 9). However, it is still describing a portion of space on whose points the respective polynomials are all sign-invariant. Thus, in the context of the MCSAT search which we discuss in the next section, the set is still describing a portion of space on whose points the constraints are all unsatisfiable for the same reason and so its negation is still a valid explanation clause to further that search.

We note that the resultants saved by this optimisation may still have to be computed later, if the cell is used in further propagation. However, this will not always be the case and so often this optimisation may save computation. Discovery of this optimisation illustrates the advantages of the proof system presentation used in this paper.

### 7.4. Factorization of polynomials and cell size

To ensure the output of CAD is correct we must compute a square-free basis of the current set of polynomials $P$ (i.e. a set of square-free polynomials without common factors which define the same varieties as $P$ ) before the application of a projection operator.

The approach of the rules presented above is to fully factorize each polynomial, resulting in what is called the finest square-free basis. This is a fairly standard choice made in CAD implementations as the effort of computing a full factorization pays off compared to the heavy resultant, discriminant, and real root isolation computations, which are all simpler for smaller polynomials.

When building a full CAD the choice of square-free basis does not effect the decomposition computed, just the time taken to compute it. However, for the single-cell construction, this makes a difference also in the size of the resulting cell, specifically the cell can be larger if we factor. This can easily be observed by considering


Figure 14: By perturbing a problem making a reducible polynomial irreducible, the problem gets harder.

Example 3.1. Here polynomial $p_{1} \cdot p_{2} \cdot p_{3}$ is already square-free. Using this directly in the one-cell construction algorithm without factorization as a whole would simply result in computing the discriminant of the polynomial (and some coefficients): the discriminant must have as factors all the cross resultants of $p_{1}, p_{2}, p_{3}$ by definition. So in this case, no improvement over a full projection is achieved, and we would find the smaller cell from Figure 1. However, if we factor to consider the set $\left\{p_{1}, p_{2}, p_{3}\right\}$ instead then we can obtain the larger cell from Figure 1.

The limits of factorization. We note that this described gain in size of the computed cell from factorization does not mean we build optimal cells for all problems where there is a geometric separation. Figure 14a shows an example with two circles and a linear polynomial for which the constructed cell we ideally build is the inside of the circle defined by $p_{1}$. Even if we were originally presented with $p_{1} \cdot p_{2}$, factorisation would allow a one-cell algorithm to ignore the intersections of $p_{2}$ and $p_{3}$ to construct the entire inner circle. But consider the similar example from Figure 14b, in which we have perturbed the problem to consider the irreducible polynomial $p_{1} \cdot p_{2}+1$. Graphically, the two problems seem to be similar, and the delineation of the roots are identical. But in the second case the single cell construction cannot treat the two ovals separately: we must consider the irrelevant intersection and thus build a smaller than ideal cell.

This shows some of the limits of the single cell construction by way of a well observed truth in computer algebra: problems which appear similar to human beings (i.e. when viewed geometrically) are not always equally hard algebraically.

## 8. Experimental evaluation

The presented proof rules are, due to their generality, potentially applicable with only small additions to a variety of problems and algorithms related to non-linear arithmetic, such as quantifier elimination by CAD [16], various CAD optimisations e.g. [34, 17], non-uniformly cylindrical decompositions for quantifier elimination [8, 9], the generation of explanations when using cylindrical algebraic coverings for a traditional SMT solver [2], and the generation of explanations in MCSAT [27, 26, 11]. As the latter was the main motivation for our work, the evaluation of this paper will focus on generating theory explanations in MCSAT for non-linear arithmetic.

### 8.1. Generating explanations for MCSAT

Recall the description of MCSAT in Section 1.2. We are interested in when MCSAT resolves theory conflicts. I.e. when there is a set of constraints $C$ in real variables $x_{1}, \ldots, x_{n}, x_{n+1}$ that should be satisfied according to the Boolean model and an assignment $s:\left\{x_{1}, \ldots, x_{n}\right\} \rightarrow \mathbb{R}$ such that $s$ cannot be extended to a value for $x_{n+1}$ satisfying $C$. The task then is to exclude a cell around $s$ that generalizes this conflict, i.e. a region cell where the reason for unsatisfiability of $C$ is invariant.

This reason of unsatisfiability is maintained when all input polynomials are sign-invariant on the generalized cell. To achieve this, we could do a full McCallum projection step, obtaining a set of properties of one level below allowing to construct a cell around $s$.

However, this is already too strong, as we need the set of input polynomials $P=\{p \mid(p \sim 0) \in C\} \subset$ $\mathbb{Q}\left[x_{1}, \ldots, x_{n}, x_{n+1}\right]$ to be delineable over a cell containing the current sample $s \in \mathbb{R}^{n}$ for maintaining the desired property. We achieve this by determining the indexed root expression of the real roots of the set $\{p \in$ factors $(P) \mid \operatorname{level}(p)=n+1\}$ over $s$ in $x_{n+1}$ and ordering them such that $\xi_{1}(s) \leq \xi_{2}(s) \leq \ldots \leq \xi_{k}(s)$. Finally, we ensure that all lower level factors $\{p \in$ factors $(P) \mid \operatorname{level}(p)<n+1\}$ are sign-invariant, check that each polynomial is not nullified (if not, we stop), make each polynomial individually delineable and add the resultants of the pair of polynomials $\left(\xi_{j} . p, \xi_{j+1} \cdot p\right)$ for $j \in[1 . . k-1]$. Thus, the input of the presented one-cell algorithm is given as

$$
\begin{aligned}
Q= & \{\text { sgn_inv }(p) \mid p \in \operatorname{factors}(P), \operatorname{level}(p)<n+1\} \\
& \cup\left\{\text { an_proj_del }(p), \text { sgn_inv }\left(\operatorname{ldcf}_{x_{n+1}}(p)\right) \mid p \in \operatorname{factors}(P), \operatorname{level}(p)=n+1\right\} \\
& \cup\left\{\text { ord_inv }\left(\operatorname{res}_{x_{n+1}}\left(\xi_{j} \cdot p, \xi_{j+1} \cdot p\right)\right) \mid j \in[k-1]\right\} .
\end{aligned}
$$

This approach is similar to the chain heuristic for indexed root orderings presented in Definition 6.7.
Note that this approach could be embedded nicely into our system as a proof rule, taking over some optimizations. Furthermore, there are alternative approaches for elimination in the first levels, i.e. by computing a covering of unsatisfying intervals of input constraints. These possibilities are part of our plans for future work.

Further note that in MCSAT, conflicts might also depend on previously computed cells which are expressed by conjunctions of extended constraints where a variable is compared with an indexed root expression; these constraints can also occur in the input. To handle an extended constraint $x_{n+1} \sim \operatorname{root}_{x_{i+1}}[p, j]$ with $p \in \mathbb{Q}\left[x_{1}, \ldots, x_{n+1}\right]$, we simply add $p$ to the set $P$ of input polynomials.

### 8.2. Implementation

For the evaluation of the presented algorithm, we employ the SMT-RAT [1, 18] solver, which provides an MCSAT engine allowing the combination of multiple explanation backends. Several incomplete and complete methods are combined in the sense that these backends are called sequentially until one returns an explanation.

Currently available backends are the Fourier-Motzkin variable elimination (FM) [26], interval constraint propagation (ICP) [28], virtual substitution (VS) [3], the complete model-based CAD cell construction algorithm from NLSAT [27] using Collin's projection operator as well as the refinement-based single cell construction algorithm [11]. Furthermore, SMT-RAT employs a fully dynamic activity-based variable ordering heuristic for scheduling theory variable assignments and Boolean decisions [39].

All variants of the presented levelwise one-cell construction algorithm are implemented as backends in SMTRAT. This is a preliminary version not fully exploiting the power of all proof rules, in particular, it is not checked whether a property is already implied by some properties in the projection. Most notably, the presence of resultants (the second and third respectively the third and fourth case) is not checked for in Rule 5.11 ; thus more leading coefficients than necessary are added to the projection set.
For the evaluation, we compare the following solver variants:
NL The model-based projection using Collin's operator from $N L S A T$ [27] as a complete explanation backend. I.e. to use when one of the following variants which are all based on McCallum projection hits a nullification which they cannot handle (see Definition 2.2).

OC-* The refinement-based one-cell construction algorithm [11]. We use the same MCSAT embedding as described above. Furthermore, the refinement-based method is able to return an explanation when a polynomial is nullified in the sector case in some special cases which our levelwise approach cannot handle yet; for better comparability, these special cases are excluded from the following tests. In case of failure, the complete explanation from $N L S A T$ is called. To further specify the algorithm's behaviour and make it reproducible, we implemented heuristics for the order of merging of initial polynomials. These specify the $*$ in the variant name as follows.

ASC The merge-operation is called on the initial polynomials in ascending order by their total degree.
DSC The merge-operation is called on the initial polynomials in descending order by their total degree.
LW_*_* The new levelwise one-cell construction algorithm with different heuristics applied in the section and sector case. In case of failure, the complete explanation from $N L S A T$ is called. The first $*$ in the variant name dictates the employed heuristic for the section case and the second for the sector case. Possible substitutions for the stars are as follows.

|  | solved | sat | unsat | with single cell | without single cell |
| :---: | :---: | :---: | :---: | :---: | :---: |
| NL | 9057 | 4518 | 4539 |  |  |
| OC-ASC | 9316 | 4605 | 4711 | 4194 | 5122 |
| OC-DSC | 9316 | 4605 | 4711 | 4194 | 5122 |
| LW-EQ-BC | 9308 | 4598 | 4710 | 4186 | 5122 |
| LW-EQ-CH | 9311 | 4602 | 4709 | 4189 | 5122 |
| LW-EQ-LDB | 9304 | 4595 | 4709 | 4182 | 5122 |
| VB-LW | 9403 | 4650 | 4753 | 4281 | 5122 |
| VB | $\mathbf{9 4 3 5}$ | $\mathbf{4 6 7 1}$ | $\mathbf{4 7 6 4}$ | $\mathbf{4 3 1 3}$ | 5122 |
| LW-CH-CH | 9305 | 4600 | 4705 | 4183 | 5122 |
| LW-LDB-LDB | 9296 | 4585 | 4711 | 4174 | 5122 |
| NL+ | 9535 | 4699 | 4836 |  |  |
| OC-ASC+ | 9797 | 4788 | 5009 | 615 | 9182 |
| LW-EQ-BC+ | 9795 | 4786 | 5009 | 613 | 9182 |
| LW-EQ-CH+ | 9797 | 4787 | 5010 | 615 | 9182 |
| LW-EQ-LDB+ | 9795 | 4786 | 5009 | 613 | 9182 |
| VB-LW+ | 9797 | 4787 | 5010 | 615 | 9182 |
| VB+ | $\mathbf{9 8 0 0}$ | $\mathbf{4 7 9 0}$ | $\mathbf{5 0 1 0}$ | $\mathbf{6 1 8}$ | 9182 |
| Total | $11552^{*}$ | $5069^{*}$ | $5379^{*}$ |  |  |
| * For 1104 instances, it is not known whether they are satisfiable or unsatisfiable. |  |  |  |  |  |
|  |  |  |  |  |  |

Table 1: Details on the instances solved by each solver: the number solved (first column), the number of satisfiable and unsatisfiable solved instances (the second and third columns) and the number of solved instances where the solver made at least one call to the single-cell construction and the number that did not require any such call to solve (the final two columns). Note that the last column is, for the first block of solvers, the number of instances that may be solved with Boolean reasoning and construction of sample points alone, i.e. without any theory calls. For the second block of solvers this also includes the instances that are solved using the additional incomplete theory backends.

EQ EqUATIONAL CONSTRAINT heuristic is applied (only for section case).
BC BIGGEST CELL heuristic is applied (only for sector case).
$\mathbf{C H}$ Chain heuristic is applied.
LDB LOWEST DEGREE BARRIERS heuristic is applied.
[solver] + with [solver] being one of the solver variants above, uses the FM-, ICP- and VS-based backends serially, in this order, before resorting to [solver]. Furthermore, we apply general preprocessing to the input before calling the main solver [18].

All variants are executed on the SMT-LIB benchmark library [4] for quantifier-free non-linear real arithmetic, abbreviated as QF_NRA. This set contains 11552 problem instances.

The machine used for testing has four 2.1 GHz AMD Opteron CPUs with 12 Cores each. In the created test series, each instance was solved with 15 minutes timeout and 6 GB of memory.

For reproducibility, the implementation which generated the following results is available at https://doi.org/ $10.5281 /$ zenodo. 5764569 .

### 8.3. Results

Examined solvers. First of all, we observed that OC-ASC solves as many instances as OC-DSC but needs slightly less time; thus, we omit OC-DSC. Furthermore, we observe that LW-EQ-CH and LW-EQ-LDB solve more instances than LW-CH-CH and LW-LDB-LDB. In our basic implementation, saving leading coefficients and discriminants pays off compared to the other heuristics. Thus for further examination, we focus on the LW-EQ-* variants which always use the equational constraints projection in the section case. A brief summary of solved instances of all solvers can be seen in Table 1. Furthermore, from now on, VB-LW (respectively VB-LW+) is the virtual best of the LW-EQ-* (respectively $\mathrm{LW}-\mathrm{EQ}-*+$ ) solvers. VB and $\mathrm{VB}+$ are the corresponding virtual bests with respect to $\mathrm{LW}-\mathrm{EQ}-*$ and OC-ASC.

|  | LW-EQ-BC | LW-EQ-BC+ | VB-LW | VB-LW+ |
| :--- | :---: | :---: | :---: | :---: |
| solved instances | 9308 | 9795 | 9403 | 9797 |
| mean running time (s) | 4.36 | 5.22 | 5.56 | 5.26 |
| sum calls to explanation | 83628 | 58826 | 107964 | 58953 |
| of them: sum calls to SC | $83628(100.0 \%)$ | $1740(2.96 \%)$ | $107964(100.0 \%)$ | $1849(3.14 \%)$ |
| of them: sum calls to NL | $15394(18.41 \%)$ | $117(0.2 \%)$ | $19412(17.98 \%)$ | $119(0.2 \%)$ |
| solved instances with single cell | 4186 | 613 | 4281 | 615 |
| mean running time (s) | 9.13 | 20.34 | 11.7 | 21.81 |
| sum calls to explanation | 83628 | 10363 | 107964 | 10490 |
| of them: sum calls to SC | $83628(100.0 \%)$ | $1740(16.79 \%)$ | $107964(100.0 \%)$ | $1849(17.63 \%)$ |
| of them: sum calls to NL | $15394(18.41 \%)$ | $117(1.13 \%)$ | $19412(17.98 \%)$ | $119(1.13 \%)$ |

Table 2: Comparison of four exemplary solvers: firstly, the subset of all solved instances and secondly the subset of all solved instances where at least one call to the single cell construction was made are considered. For each, the statistics describing the total number of instances, the mean running time, the sum of constructed cells as well as the sum of attempts to generate a cell using single cell construction or NLSAT backend are given. For the last two, these sums are included in the previous one; their share of the total number of constructed cells is given. Note that the explanation calls involving single cell are the ones not being solved by the additional backends and the calls involving NLSAT are the ones where the single cell construction failed.


Figure 15: Performance profile (running times).

General observations. Before we compare the different approaches, we make some general comments on the results based on exemplary solvers in Table 2.

As already observed in Table 1 and confirmed by Table 2, large parts of the benchmark set are relatively easy. Around half of the benchmarks do not involve a single explanation call; and when enabling the additional incomplete backends, $79 \%$ of the benchmarks can be solved without a single call to the single cell construction. Considering the total number of explanation calls made, only $3.14 \%$ of them use the single cell construction for the VB-LW+ solver, meaning that even for the problems where single cell is needed, it is only needed rarely.

The fail rate of the levelwise backend (the cases where a nullification occurs) is smaller on the VB-LW+ solver ( $6 \%$ ) than on the VB-LW solver ( $17.98 \%$ ). That means, that nullifications are more probable on the simple parts of the problem.

It should also be noted that VB-LW+ needs significantly less explanation calls than VB-LW; which means, that the incomplete backends have explanations of higher quality.

To summarize, we can only make meaningful statements on our heuristics based on the solver variants without the additional backends, as otherwise, there are too few calls to the single cell construction in solved instances. The possible reasons for this are twofold. On the one hand, our procedure and its implementation may not yet be suitable to solve the harder instances in the benchmark set. On the other hand, the benchmark set might not contain enough interesting or diverse benchmarks for an evaluation.


Figure 16: Commonly solved instances.

Overall results. All solvers are depicted in the performance profile in Figure 15.
First of all, the NL and NL+ solvers perform significantly worse than the single cell variants, which justifies the investigation of this new levelwise cell construction approach.

The refinement-based approach OC-ASC as well as LW-EQ-* perform similarly. Among the levelwise variants, the LW-EQ-CH solves the most; however, these differences are not significant and depend on the implementation and heuristics chosen in the MCSAT solver. Considering OC-ASC+ and LW-EQ-*+, these differences vanish even more when combining them with incomplete methods handling simple sub-problems. These results are summarized in Table 1.

Virtual best and orthogonality of heuristics. VB-LW performs significantly better than the LW-EQ-* solvers. This means that although the number of solved instances is similar for all heuristics, each heuristic solves instances that the others do not solve. Thus these heuristics are orthogonal to some degree. We depict the number of instances solved by differing combinations of solvers in Figure 16. Note that the same holds for VB, meaning that OC-ASC is orthogonal to LW-EQ-* as well.

Note that the VB-LW+ solver does not solve significantly more instances than any of the LW-EQ-*+ solvers. That means the differences of the heuristics only become noticeable in the simple parts of the instances. An explanation could be that harder parts of instances require heavy resultant computations which could quickly shift the instance to unsolvable within the timeout.

Now focusing on the "pure" solvers without additional backends, we observe that on simple instances the refinement based approach OC-ASC and the virtual best of the levelwise approaches VB-LW behave similar on simple instances while they are more "orthogonal" on harder instances, as indicated by Figure 17a.

The SMT-LIB benchmark set is split up into families which each have a similar structure. We note that all variants LW-EQ-* and OC-ASC solve roughly the same amount of benchmarks in each family individually, as seen in Table 3, while the virtual best solvers do solve more. That means that the orthogonality of the variants is split up onto all families.

Number of constructed cells. Figure 18 depicts a performance profile where we consider the number of constructed cells (that is, the number of times the explanation function is called by MCSAT) instead of the running time. We can clearly see that the solvers that solve more instances tend to compute fewer cells during a run. That is, the quality of the explanations is better. Again, considering the refinement based approach and the levelwise approaches, differences are only significant without the other backends. In particular, the virtual best solver clearly solves more instances with fewer cells. Figure 17b makes this even more clear by comparing the number of cells constructed by VB-LW and OC-ASC for every instance.

Comparison of heuristics. For further comparison of the three heuristics, we collected two more statistics for each instance: the dimension of constructed cells (i.e. the number of levels considered during its construction), the maximum degree in the main variable of the polynomials occurring in the computation, and the size of the


Figure 17: Scatter plots comparing VB-LW and OC-ASC on the individual instances.

|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 20 | 11 |
| :--- | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| OC-ASC | 25 | 0 | 0 | 86 | 9 | 2 | 26 | 6 | 2180 | 8 | 6913 |
| LW-EQ-BC | 22 | 0 | 0 | 89 | 9 | 2 | 22 | 6 | 2170 | 8 | 6918 |
| LW-EQ-CH | 25 | 0 | 0 | 88 | 9 | 3 | 22 | 5 | 2174 | 8 | 6914 |
| LW-EQ-LDB | 22 | 0 | 0 | 88 | 9 | 2 | 22 | 7 | 2170 | 8 | 6913 |
| VB-LW | 26 | 0 | 0 | 90 | 9 | 4 | 25 | 7 | 2249 | 8 | 6918 |
| VB | 27 | 0 | 0 | 94 | 9 | 4 | 28 | 7 | 2267 | 8 | 6923 |
| Total | 405 | 9 | 69 | 135 | 63 | 821 | 61 | 20 | 2752 | 45 | 7006 |

Table 3: Solved instances by solver and family. Benchmark families: (1) 20161105-Sturm-MBO (2) 20161105-Sturm-MGC (3) 20170501-Heizmann-UltimateInvariantSynthesis (4) 20180501-Economics-Mulligan (5) 2019-ezsmt (6) LassoRanker (7) UltimateAutomizer (8) hong (9) hycomp (10) kissing (11) meti-tarski (12) zankl.
computed projection (i.e. number of resultants, discriminants and coefficients). A summary of these statistics is shown in Table 4.

First note that the virtual best has the lowest value or close to the lowest value in all categories, which means that they might indicate the performance of a solver to some degree. However, interpretation needs to be careful, as the differences are relatively small, confirming the previous observations. Further note that we do consider fewer instances as in previous analysis. However, we do have two minor observations to make here.

Regarding the heuristic LW-EQ-LDB: its idea was to minimize the degrees of the polynomials in the projection, however, the average maximum degree is slightly higher than for the other two heuristics. LW-EQ-BC needs slightly less projection steps, but is similar to the other solvers in terms of number of cells created. This means that the size of the cells are similar, although again, the intention was to produce the biggest possible cells. To emphasize, in the analysis in Figure 18 which is based on all solved instances (including the ones where the NLSAT backend was used as fallback), it needs more cells than LW-EQ-LDB.

To summarize, this simple analysis can not confirm that the ideas behind the different heuristics take effect on the benchmarks or prove different behaviour on all benchmarks. However, as stated above, on individual benchmarks, they do behave differently, as shown by the performance of the virtual best solver.

## 9. Conclusions and future work

### 9.1. Future work

The formulation of the presented proof system allows heuristic to influence the shape of the constructed cells. The experimental evaluation shows potential for further development of those. Furthermore, for some applications such as incremental linearization [15], under-approximations of the constructed cell might be beneficial if they can


Figure 18: Performance profile (number of cells).

|  | mean \#cells <br> per inst. | mean dim. <br> of cells | mean max <br> deg. per inst. | mean \#res. <br> per inst. | mean \#disc. <br> per inst. | mean \#coeff. <br> per inst. |
| :---: | :---: | :---: | :---: | :---: | :---: | :---: |
| LW-EQ-BC | 4.38 | 8.59 | $\mathbf{6 . 5 1}$ | 8.95 | 13.94 | 13.62 |
| LW-EQ-CH | 4.39 | 8.52 | 6.52 | 9.07 | 14.04 | 13.80 |
| LW-EQ-LDB | 4.39 | $\mathbf{8 . 4 8}$ | 6.57 | 9.07 | 14.05 | 13.88 |
| VB-LW | $\mathbf{4 . 3 2}$ | 8.49 | 6.52 | $\mathbf{8 . 7 7}$ | $\mathbf{1 3 . 7 5}$ | $\mathbf{1 3 . 4 8}$ |

Table 4: Statistics on the 3226 instances solved by all three solvers with at least one call to the single cell construction and without a call to the NLSAT explanation (single cell construction always succeeds): mean number of cells per instance, mean dimension of constructed cells, mean maximum degree of polynomials occurring in an instance, mean number of resultants/discriminants/coefficients in the projection per instance.
be computed more efficiently; more concretely, the resultant computations get trivial if the lower and upper bounds are replaced by one or more linear polynomials, resulting in boxes or polyhedra.

The proof system could be applied in the future in contexts other than MCSAT, such as the cylindrical algebraic coverings method [2] or quantifier elimination algorithms such as $N u C A D[8,9]$.

By relying on the theory of McCallum projection, the presented proof system is incomplete. Recently, there has been progress on the Lazard projection operator $[29,35,36,12,37,38]$, which is complete while maintaining the advantages of the McCallum operator. Thus, it is promising to extend our framework to exploit the Lazard projection.

Finally, the presented set of inference rules allows to produce fine-grained proof graphs, which could enable the certification and external automated verification of results.

### 9.2. Conclusion

We introduced the new concept of levelwise single cell construction, motivated by maintaining the savings of the existing refinement based approach of Brown [11] while allowing for more flexibility and new optimisations.

The theoretical part of this paper consists of the introduction of the novel notion of projective delineability and the presentation of a proof system in order to enable fine-grained projections based on a given sample. To demonstrate the possibilities of the proof system, we gave a simple algorithm which builds upon this proof system as well as several heuristics for the application of the given rules. Finally, we gave a qualitative evaluation as well as some notable observations.

We evaluated our algorithm by an implementation applied to explanation generation in MCSAT. We showed that our basic heuristics yield different performances for different instances indicating that there is room for further algorithmic development, as well as a more elaborated implementation of the proof rules.

Importantly, our proof system allows for a wide range of improvements through experimentation with heuristics, approximations and on theoretical matters as well as extensions to other algorithms than the single cell construction.

## Acknowledgements

Jasper Nalbach was supported by the DFG RTG 2236 UnRAVeL. James Davenport and Matthew England were supported by the EPSRC DEWCAD Project, Pushing Back the Doubly-Exponential Wall of Cylindrical Algebraic Decomposition (Grant References EP/T015748/1 and EP/T015713/1).

## References

[1] SMT-RAT, a toolbox for strategic and parallel satisfiability modulo theories solving. URL: https://github. com/smtrat/smtrat.
[2] Erika Ábrahám, James Davenport, Matthew England, and Gereon Kremer. Deciding the Consistency of Non-Linear Real Arithmetic Constraints with a Conflict Driven Search Using Cylindrical Algebraic Coverings. Journal of Logical and Algebraic Methods in Programming, 119:100633, 2021. URL: https://doi.org/10. 1016/j.jlamp. 2020.100633.
[3] Erika Ábrahám, Jasper Nalbach, and Gereon Kremer. Embedding the virtual substitution method in the model constructing satisfiability calculus framework (work-in-progress paper). In M. England and V. Ganesh, editors, Proceedings of the 2nd International Workshop on Satisfiability Checking and Symbolic Computation (SC ${ }^{2}$ 2017), number 1974 in CEUR Workshop Proceedings, 2017. URL: http://ceur-ws.org/Vol-1974/.
[4] Clark Barrett, Aaron Stump, Cesare Tinelli, et al. The SMT-LIB standard: Version 2.0. In Proceedings of the 8th International Workshop on Satisfiability Modulo Theories (Edinburgh, England), volume 13, page 14, 2010.
[5] Russell Bradford, James H. Davenport, Matthew England, Scott McCallum, and David Wilson. Truth table invariant cylindrical algebraic decomposition. Journal of Symbolic Computation, 76:1-35, 2016. URL: http: //dx.doi.org/10.1016/j.jsc.2015.11.002.
[6] Christopher W. Brown. Improved projection for cylindrical algebraic decomposition. Journal of Symbolic Computation, 32(5):447-465, 2001. URL: https://doi.org/10.1006/jsco.2001.0463.
[7] Christopher W. Brown. Constructing a single open cell in a cylindrical algebraic decomposition. In Proceedings of the 38th International Symposium on Symbolic and Algebraic Computation, ISSAC '13, pages 133-140. ACM, 2013. URL: https://doi.org/10.1145/2465506. 2465952.
[8] Christopher W. Brown. Open non-uniform cylindrical algebraic decompositions. In Proceedings of the 2015 International Symposium on Symbolic and Algebraic Computation, ISSAC '15, pages 85-92. ACM, 2015. URL: https://doi.org/10.1145/2755996.2756654.
[9] Christopher W. Brown. Projection and quantifier elimination using non-uniform cylindrical algebraic decomposition. In Proceedings of the 2017 ACM on International Symposium on Symbolic and Algebraic Computation, ISSAC '17, pages 53-60. ACM, 2017. URL: https://doi.org/10.1145/3087604.3087651.
[10] Christopher W. Brown and Glenn C. Daves. Applying machine learning to heuristics for real polynomial constraint solving. In A. Bigatti, J. Carette, J.H. Davenport, M. Joswig, and T. de Wolff, editors, Mathematical Software - ICMS 2020, volume 12097 of Lecture Notes in Computer Science, pages 292-301. Springer International Publishing, 2020. URL: https://doi.org/10.1007/978-3-030-52200-1_29.
[11] Christopher W. Brown and Marek Košta. Constructing a single cell in cylindrical algebraic decomposition. Journal of Symbolic Computation, 70:14-48, 2015. URL: https://doi.org/10.1016/j.jsc.2014.09.024.
[12] Christopher W. Brown and Scott McCallum. Enhancements to lazard's method for cylindrical algebraic decomposition. In F. Boulier, M. England, T.M. Sadykov, and E.V. Vorozhtsov, editors, Computer Algebra in Scientific Computing, volume 12291 of Lecture Notes in Computer Science, pages 129-149. Springer International Publishing, 2020. URL: https://doi.org/10.1007/978-3-030-60026-6_8.
[13] Bruno Buchberger, George E. Collins, Rudiger Loos, and Rudolf Albrecht, editors. Computer algebra: symbolic and algebraic computation (2nd ed.). Springer-Verlag New York, Inc., New York, NY, USA, 1983.
[14] Bob F. Caviness and Jeremy R. Johnson. Quantifier Elimination and Cylindrical Algebraic Decomposition. Texts \& Monographs in Symbolic Computation. Springer-Verlag, 1998. URL: https://doi.org/10.1007/ 978-3-7091-9459-1.
[15] Alessandro Cimatti, Alberto Griggio, Ahmed Irfan, Marco Roveri, and Roberto Sebastiani. Invariant checking of NRA transition systems via incremental reduction to LRA with EUF. In International Conference on Tools and Algorithms for the Construction and Analysis of Systems (TACAS 2017), pages 58-75. Springer, 2017. URL: https://doi.org/10.1007/978-3-662-54577-5_4.
[16] George E. Collins. Quantifier elimination for real closed fields by cylindrical algebraic decomposition. In Proceedings of the 2nd GI Conference on Automata Theory and Formal Languages, pages 134-183. SpringerVerlag (reprinted in the collection [14]), 1975. URL: https://doi.org/10.1007/3-540-07407-4_17.
[17] George E. Collins and Hoon Hong. Partial cylindrical algebraic decomposition for quantifier elimination. Journal of Symbolic Computation, 12:299-328, 1991. URL: https://doi.org/10.1016/S0747-7171(08)80152-6.
[18] Florian Corzilius, Gereon Kremer, Sebastian Junges, Stefan Schupp, and Erika Ábrahám. SMT-RAT: An open source C ++ toolbox for strategic and parallel SMT solving. In International Conference on Theory and Applications of Satisfiability Testing, pages 360-368. Springer, 2015. URL: https://doi.org/10.1007/ 978-3-319-24318-4_26.
[19] D.A. Cox, J.B. Little, and D.B. O'Shea. Ideals, Varieties and Algorithms. Springer-Verlag, 2006.
[20] James H. Davenport and Joos Heintz. Real quantifier elimination is doubly exponential. Journal of Symbolic Computation, 5(1-2):29-35, 1988. URL: https://doi.org/10.1016/S0747-7171 (88)80004-X.
[21] Leonardo de Moura and Dejan Jovanović. A model-constructing satisfiability calculus. In R. Giacobazzi, J. Berdine, and I. Mastroeni, editors, Verification, Model Checking, and Abstract Interpretation (Proc. VMCAI 2013), pages 1-12. Springer Berlin Heidelberg, 2013. URL: https://doi.org/10.1007/978-3-642-35873-9_ 1.
[22] Lionel Ducos. Optimizations of the subresultant algorithm. Journal of Pure and Applied Algebra, 145(2):149163, 2000. URL: https://doi.org/10.1016/S0022-4049(98)00081-4.
[23] Matthew England, Russell Bradford, and James H. Davenport. Cylindrical algebraic decomposition with equational constraints. Journal of Symbolic Computation, 100:38-71, 2020. URL: https://doi.org/10. 1016/j.jsc.2019.07.019.
[24] Israel M. Gelfand, Mikhail M. Kapranov, and Andrei V. Zelevinsky. Discriminants, Resultants, and Multidimensional Determinants. Springer Berlin, 1994.
[25] Dejan Jovanović. Solving nonlinear integer arithmetic with MCSAT. In Proceedings of the 18th International Conference on Verification, Model Checking, and Abstract Interpretation (VMCAI 2017), volume 10145 of Lecture Notes in Computer Science, pages 330-346. Springer Verlag, 2017. URL: https://doi.org/10.1007/ 978-3-319-52234-0_18.
[26] Dejan Jovanovic, Clark Barrett, and Leonardo De Moura. The design and implementation of the model constructing satisfiability calculus. In 2013 Formal Methods in Computer-Aided Design, pages 173-180. IEEE, 2013. URL: https://doi.org/10.1109/FMCAD.2013.7027033.
[27] Dejan Jovanovic and Leonardo de Moura. Solving non-linear arithmetic. In B. Gramlich, D. Miller, and U. Sattler, editors, Automated Reasoning: 6th International Joint Conference (IJCAR), volume 7364 of Lecture Notes in Computer Science, pages 339-354. Springer, 2012. URL: https://doi.org/10.1007/978-3-642-31365-3_ 27.
[28] Gereon Kremer. Cylindrical Algebraic Decomposition for Nonlinear Arithmetic Problems. PhD dissertation, RWTH Aachen University, 2019. URL: https://publications.rwth-aachen.de/record/792185.
[29] Daniel Lazard. An improved projection for cylindrical algebraic decomposition. In C.L. Bajaj, editor, Algebraic Geometry and its Applications: Collections of Papers from Abhyankar's 60th Birthday Conference, pages 467476. Springer Berlin, 1994. URL: https://doi.org/10.1007/978-1-4612-2628-4_29.
[30] Haokun Li and Bican Xia. Solving Satisfiability of Polynomial Formulas By Sample-Cell Projection, March 2020. arXiv:2003. 00409.
[31] Scott McCallum. An improved projection operation for cylindrical algebraic decomposition. PhD thesis, University of Wisconsin-Madison, 1985.
[32] Scott McCallum. An improved projection operation for cylindrical algebraic decomposition of threedimensional space. Journal of Symbolic Computation, 5(1-2):141-161, 1988. URL: https://doi.org/10. 1016/S0747-7171 (88) 80010-5.
[33] Scott McCallum. An improved projection operation for cylindrical algebraic decomposition. In Quantifier Elimination and Cylindrical Algebraic Decomposition, pages 242-268. Springer, 1998. URL: https://doi. org/10.1007/978-3-7091-9459-1_12.
[34] Scott McCallum. On projection in CAD-based quantifier elimination with equational constraint. In Proceedings of the 1999 international symposium on Symbolic and algebraic computation, pages 145-149, 1999. URL: https://doi.org/10.1145/309831.309892.
[35] Scott McCallum and Hoon Hong. On using Lazard's projection in CAD construction. Journal of Symbolic Computation, 72:65-81, 2016. URL: https://doi.org/10.1016/j.jsc.2015.02.001.
[36] Scott McCallum, Adam Parusińiski, and Laurentiu Paunescu. Validity proof of Lazard's method for CAD construction. Journal of Symbolic Computation, 92:52-69, 2019. URL: https://doi.org/10.1016/j.jsc. 2017.12.002.
[37] Akshar Nair, James H. Davenport, and Gregory Sankaran. On benefits of equality constraints in lex-least invariant CAD. In J. Abbott and A. Griggio, editors, Proceedings of the 4 th Workshop on Satisfiability Checking and Symbolic Computation (SC ${ }^{2}$ 2019), number 2460 in CEUR Workshop Proceedings, 2019. URL: http: //ceur-ws.org/Vol-2460/.
[38] Akshar Nair, James H. Davenport, and Gregory Sankaran. Curtains in CAD: Why are they a problem and how do we fix them? In A. Bigatti, J. Carette, J.H. Davenport, M. Joswig, and T. de Wolff, editors, Mathematical Software - ICMS 2020, volume 12097 of Lecture Notes in Computer Science, pages 17-26. Springer International Publishing, 2020. URL: https://doi.org/10.1007/978-3-030-52200-1_2.
[39] Jasper Nalbach, Gereon Kremer, and Erika Ábrahám. On variable orderings in MCSAT for non-linear real arithmetic. In J. Abbott and A. Griggio, editors, Proceedings of the 4 th Workshop on Satisfiability Checking and Symbolic Computation (SC ${ }^{2}$ 2019), number 2460 in CEUR Workshop Proceedings, 2019. URL: http: //ceur-ws.org/Vol-2460/.
[40] Alfred Tarski. A Decision Method For Elementary Algebra And Geometry. RAND Corporation, Santa Monica, CA (reprinted in the collection [14]), 1948.

## Appendix A. Technical theorems on projection

In the following theorem, we prove analytic projective delineability of a polynomial locally in some neighbourhood around a point.

Theorem A.1. Let $p \in \mathbb{R}\left[x_{1}, \ldots, x_{n}\right]$ be square-free and $R$ be a connected analytic submanifold in $\mathbb{R}^{n-1}$ in which $\operatorname{disc}_{x_{n}}(p)$ is order-invariant and $p$ is non-nullified. For any point $r$ in $R$, there exists a neighbourhood of $r$ such that the intersection of that neighbourhood and $R$ is an analytic submanifold (which we will refer to as $N_{r}$ ) and for which either

1. $\operatorname{ldcf}_{x_{n}}(p)$ is sign-invariant on $N_{r}$, or
2. $\operatorname{ldcf}_{x_{n}}(p)$ is zero at $r$ but is non-zero at all points in $N_{r}$ arbitrarily close to $r$, in which case:
(a) realRoots $(p) \cap\left(N_{r} \times \mathbb{R}\right)$ consists of the graphs of $k+1$ analytic functions $\theta_{1}, \ldots, \theta_{k}, \theta_{*}$ where the domain of each $\theta_{1}, \ldots, \theta_{k}$ is $N_{r}$ and the domain of $\theta_{*}$ is $N_{r} \backslash$ realRoots $\left(\operatorname{ldcf}_{x_{n}}(p)\right)$, and for all paths $\sigma:[0,1] \rightarrow R$ such that $\operatorname{ldcf}_{x_{n}}(p)$ is zero at $\sigma(1)$ and non-zero for all other points on the path, $\lim _{t \rightarrow 1-} \theta_{*}(\sigma(t))= \pm \infty$;
(b) there are integer constants $m_{1}, \ldots, m_{k}, m_{*}$ such that for all $r^{\prime} \in N_{r}$, the multiplicity of $\theta_{i}\left(r^{\prime}\right)$ as a root of $p\left(r_{1}^{\prime}, \ldots, r_{n-1}^{\prime}, x_{n}\right)$ is $m_{i}$, and for all $r^{\prime} \in N_{r} \backslash$ realRoots $\left(\operatorname{ldcf}_{x_{n}}(p)\right)$, the multiplicity of $\theta_{*}\left(r^{\prime}\right)$ as a root of $p\left(r_{1}^{\prime}, \ldots, r_{n-1}^{\prime}, x_{n}\right)$ is $m_{*}$;
(c) for all $r^{\prime} \in N_{r}, \theta_{1}\left(r^{\prime}\right)<\cdots<\theta_{k}\left(r^{\prime}\right)$, and for all $r^{\prime} \in N_{r} \backslash \operatorname{realRoots}\left(\operatorname{ldcf}_{x_{n}}(p)\right)$, either $\theta_{k}\left(r^{\prime}\right)<\theta_{*}\left(r^{\prime}\right)$ or $\theta_{*}\left(r^{\prime}\right)<\theta_{1}\left(r^{\prime}\right) ;$ and
(d) $p$ is order-invariant in the p-sections $\theta_{1}, \ldots, \theta_{k}$ over $R$ and has constant order at every point of the graph of $\theta_{*}$ over $R \backslash \operatorname{realRoots}\left(\operatorname{ldcf}_{x_{n}}(p)\right)$.

To illustrate what this theorem is saying, consider the polynomial $p:=x y^{3}+16 y^{2}-32$ with variable ordering $x \prec y$. We take $x=0$ as the chosen point $r$, and we take $R$ to be an open interval on the $x$-axis containing $r$. The pictures below show the curve defined by the variety of $p$, and the situation over $R$ and how the $\theta_{1}, \theta_{2}$ and $\theta_{*}$ from the theorem relate to the picture.


The particular feature to keep in mind is the discontinuity of $\theta_{*}$ at the point $x=0$, and how, as we move through the discontinuity from negative $x$ to positive $x, \theta_{*}$ switches from being below $\theta_{1}$ and $\theta_{2}$ to being above them. Note also that if $R$ were expanded either right or left, we would violate the requirement that the $\operatorname{disc}_{y}(p)$ is order-invariant.

Proof of Theorem A.1. If $\operatorname{ldcf}_{x_{n}}(p)$ is sign-invariant in $N_{r}$, then the theorem holds. The interesting case is when $\operatorname{ldcf}_{x_{n}}(p)$ is not sign-invariant in $N_{r}$, no matter how small we make $N_{r}$. If this is the case, then there must be points in $R$ at which $\operatorname{ldcf}_{x_{n}}(p)$ vanishes that are arbitrarily close to $r$.

Choose a real number $\alpha$ such that all roots of $p\left(r_{1}, \ldots, r_{n-1}, x_{n}\right)$ are greater than $\alpha$, and choose $N_{r}$ to be sufficiently small so that $p$ has no zeros in $N_{r} \times \mathbb{R}$ that intersect with the hyperplane $x_{n}=\alpha$. Since $p$, being a polynomial, is continuous and non-zero at $\left(r_{1}, \ldots, r_{n-1}, x_{n}\right)$, such an $N_{r}$ exists. Let $p^{*}=p\left(x_{1}, \ldots, x_{n-1}, x_{n}-\alpha\right)$. Note that the discriminant is invariant under this transformation (see for example [24]) so disc $\operatorname{din}_{x_{n}}(p)=\operatorname{disc}_{x_{n}}\left(p^{*}\right)$. The mapping that shifts $x_{n}$ by $\alpha$ is a homeomorphism that maps the roots of $p$ to the roots of $p^{*}$. So we can study the roots of $p$ by understanding the roots of $p^{*}$. But for $p^{*}$ we are guaranteed that $p^{*}\left(r_{1}, \ldots, r_{n-1}, 0\right) \neq 0$. In fact, over $N_{r}$, no zero of $p^{*}$ lies in the hyperplane $x_{n}=0$. To put another way, the constant coefficient with respect to $x_{n}$ of $p^{*}$ does not vanish on $N_{r}$.

Let $d$ be the degree of $p^{*}$ in $x_{n}$. We now consider the polynomial $\bar{p}=x_{n}^{d} p^{*}\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)$. We note that, over $N_{r}$, the transformation from $p$ to $\bar{p}$ defines an analytic homeomorphism from $N_{r} \times(\mathbb{R} \backslash\{0\})$ to itself that maps the zeros of $\bar{p}$ off the $x_{n}=0$ hyperplane to the zeros of $p$ (none of which, recall, lie on the $x_{n}=\alpha$ hyperplane). Specifically, the homeomorphism is given by the mapping $(x, y) \mapsto(x, 1 / y-\alpha)$. The transformation leaves the discriminant unchanged as well, i.e. $\operatorname{disc}_{x_{n}}(\bar{p})=\operatorname{disc}_{x_{n}}\left(p^{*}\right)$. This also follows from [24], but is also addressed explicitly as Lemma 8.1 of [6]. Thus, the discriminant of $\bar{p}$ is order-invariant in $R$. Moreover, its leading coefficient, which is the same as the constant coefficient (in $x_{n}$ ) of $p^{*}$, is non-zero throughout $N_{r}$. This means that $\bar{p}$ is not nullified anywhere in $N_{r}$ and its leading coefficient (in $x_{n}$ ) is sign-invariant on $N_{r}$. Thus, by the Brown-McCallum projection, $\bar{p}$ is analytically delineable over $N_{r}$. It must have a section that passes through the point $\left(r_{1}, \ldots, r_{n-1}, 0\right)$, since its constant coefficient (in $x_{n}$ ) vanishes at $r$. So let the sections of $\bar{p}$ be $\bar{\theta}_{1} \prec \cdots \prec \bar{\theta}_{k}$ with multiplicities $\bar{m}_{1}, \ldots, \bar{m}_{k}$, and let $i$ be the index such that $\bar{\theta}_{i}$ is the section passing through $\left(r_{1}, \ldots, r_{n-1}, 0\right)$. Note that it must be unique because delineability guarantees non-intersecting sections. Since none of the remaining sections pass through $\left(r_{1}, \ldots, r_{n-1}, 0\right)$ we can refine $N_{r}$ to $N_{r}^{\prime}$ that is sufficiently small so that none of the remaining sections intersect the hyperplane $x_{n}=0$ over $N_{r}^{\prime}$.

So, over $N_{r}^{\prime}$ the $\bar{p}$-section $\bar{\theta}_{j}$, where $j \neq i$, maps to $p$-section $\theta_{j}$. Moreover, since $\theta_{i}$ is the only $\bar{p}$-section that crosses the $x_{n}=0$ hyperplane, the graphs of $\bar{\theta}_{1}, \ldots, \bar{\theta}_{i-1}$ lie in $N_{r}^{\prime} \times \mathbb{R}_{<0}$, and the graphs of $\bar{\theta}_{i+1}, \ldots, \bar{\theta}_{k}$ lie in $N_{r}^{\prime} \times \mathbb{R}_{>0}$. This means these $p$-sections are non-intersecting and ordered as follows $\theta_{i-1}<\cdots<\theta_{0}<\theta_{k}<\ldots \theta_{i+1}$. Finally, consider a connected region $M$ in $N_{r}^{\prime}$ on which the leading coefficient of $p$ or, equivalently, the constant coefficient of $\bar{p}$ is non-zero. Section $\bar{\theta}_{i}$ either lies entirely above the $x_{n}=0$ hyperplane on this region, in which case it maps via the homeomorphism to a section over $M$ that lies above section $\theta_{i+1}$, or entirely below $x_{n}=0$, in which case it maps to section over $M$ that lies below section $\theta_{i-1}$. So the $\theta_{*}$ required by the theorem is the image $\bar{\theta}_{i}$ via the homeomorphism restricted to the points in $N_{r}^{\prime}$ at which the trailing coefficient of $\bar{p}$ is non-zero, which is equivalent to $N_{r}^{\prime} \backslash \operatorname{realRoots}\left(\operatorname{ldcf}_{x_{n}}(p)\right)$. Note that because $\overline{\theta_{i}}$ passes through $x_{n}=0$ at precisely the points that map back to points in realRoots $\left(\operatorname{ldcf}_{x_{n}}(p)\right), \lim _{t \rightarrow 1} \theta_{*}(\sigma(t))= \pm \infty$. Thus points 2.a and 2.c are proven.

Next we show that the multiplicities of the sections are constant, i.e. that point $2 . b$ of the theorem holds. We will show that the multiplicities of sections of $\bar{p}$ carry over to the sections of $p^{*}$, which suffices since the sections of $p^{*}$ are just shifts of the sections of $p$. Let $M_{d}$ be the map of a polynomial in $x_{n}$ of degree at most $d$ to a polynomial
of degree at most $d$ in $x_{n}$ given by $M_{d}\left(h\left(x_{n}\right)\right)=x_{n}^{d} h\left(1 / x_{n}\right)$. Note that $M_{d}\left(p^{*}\right)=\bar{p}$ and $M_{d}(\bar{p})=p^{*}$. First, consider point $r^{\prime}$ in $N_{r}^{\prime} \backslash \operatorname{realRoots}\left(\operatorname{ldcf}_{x_{n}}(p)\right)$. In this case,

$$
\bar{p}\left(r_{1}^{\prime}, \ldots, r_{n-1}^{\prime}, x_{n}\right)=\gamma \prod_{i=1}^{r}\left(x-\beta_{i}\right)^{e_{i}}
$$

where the $\beta_{i}$ s are the distinct zeros (over $\mathbb{C}$ ) and the $e_{i}$ s are the multiplicities of those zeros, and $\gamma$ is some non-zero constant. Hence we have

$$
\begin{aligned}
p^{*}\left(r_{1}^{\prime}, \ldots, r_{n-1}^{\prime}, x_{n}\right) & =M_{d}\left(\bar{p}\left(r_{1}^{\prime}, \ldots, r_{n-1}^{\prime}, x_{n}\right)\right. \\
& =M_{d}\left(\gamma \prod_{i=1}^{r}\left(x-\beta_{i}\right)^{e_{i}}\right) \\
& =\gamma \prod_{i=1}^{r} M_{d}\left(x-\beta_{i}\right)^{e_{i}}=\gamma \prod_{i=1}^{r}\left(1-\beta_{i} x\right)^{e_{i}}
\end{aligned}
$$

and we see that the multiplicities of the roots are preserved. Thus, we are left with the case $r^{\prime} \in \operatorname{realRoots}\left(\operatorname{ldcf}_{x_{n}}(p)\right)$. In this case, $p^{*}\left(r_{1}^{\prime}, \ldots, r_{n-1}^{\prime}, x_{n}\right)=q^{*}\left(x_{n}\right)$ for some polynomial $q^{*} \in \mathbb{R}\left[x_{n}\right]$ of degree less than $d$ (because the leading coefficient, and possibly some others, vanish at $r^{\prime}$ ). The leading coefficient of $\bar{p}$ does not vanish at $r^{\prime}$, but the trailing coefficient does. In fact, the coefficients of the degree zero up to $m_{*}-1$ all vanish at $r^{\prime}$. So we have $\bar{p}\left(r_{1}^{\prime}, \ldots, r_{n-1}^{\prime}, x_{n}\right)=x_{n}^{m_{*}} \cdot \bar{q}\left(x_{n}\right)$, where $\bar{q}\left(x_{n}\right)=M_{d-m_{*}}\left(p^{*}\left(x_{n}\right)\right)$. So the roots of $\bar{q}\left(x_{n}\right)$ are $\theta_{1}\left(r^{\prime}\right), \ldots, \theta_{k}\left(r^{\prime}\right)$ with multiplicities $m_{1}, \ldots, m_{k}$, and these roots and multiplicities carry over to $q^{*}$, which can be proven with the same argument used in the previous case.

To finish we need to justify the claim of order-invariance for the sections $\theta_{1}, \ldots, \theta_{k}$ and for $\theta_{*}$. Lemma A. 1 (below) shows that the orders of $p$ and $\bar{p}$ are the same at associated points. (Note that Theorem 2.1 of [32] essentially shows that order is preserved under general analytic coordinate transformation, which includes Lemma A. 1 as a special case.) Given that, we note that polynomial $\bar{p}$ is order-invariant in the sections $\bar{\theta}_{1}, \cdots, \bar{\theta}_{k}$ and $\overline{\theta_{*}}$ (by McCallum's theorem on projection), and when we map these sections back to zeros of $p$, the orders of $p$ in $\theta_{1}, \ldots, \theta_{k}$ and $\theta_{*}$ will be the same as the orders of $\bar{p}$ in $\bar{\theta}_{1}, \cdots, \bar{\theta}_{k}$ and $\bar{\theta}_{*}$, which concludes the proof.

Lemma A.1. Let $p \in \mathbb{R}\left[x_{1}, \ldots, x_{n}\right]$ be of degree $d$ in $x_{n}$. Let $r \in \mathbb{R}^{n}$ with $r_{n} \neq 0$ be a point at which $p$ is non-zero. Define the function $f$ as $f(p)=x_{n}^{d} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)$, and the function $g(r)=\left(r_{1}, \ldots, r_{n-1}, 1 / r_{n}\right)$, noting that $f(p) \in \mathbb{R}\left[x_{1}, \ldots, x_{n}\right]$ is of degree $d$ in $x_{n}$, and $g(r)$ is a point in $\mathbb{R}^{n}$ with non-zero $n$th coordinate. The order of $p$ at $r$ is the same as the order of $f(p)$ at $g(r)$.

Proof. We will show that $\operatorname{ord}_{r}(p) \leq \operatorname{ord}_{g(r)}(f(p))$. Since $f(f(p))=p$ and $g(g(r))=r$, and $f(p)$ and $g(r)$ satisfy the hypotheses of the theorem, we may reverse the roles of $p$ and $f(p)$, and $r$ and $g(r)$ and get that ord $g(r)(f(p)) \leq$ $\operatorname{ord}_{r}(p)$.

Let $\bar{p}=f(p)$ and $\bar{r}=g(r)$. We will use the usual multi-index notation for partial derivation so that $\partial^{\omega}=$ $\frac{\partial^{\omega_{1}}}{\partial x_{1}^{\omega_{1}}} \cdots \frac{\partial^{\omega_{n}}}{\partial x_{n}^{\omega_{n}}}$. We show by induction on $\|\omega\|_{1}$ that

$$
\begin{equation*}
\partial^{\omega} \bar{p}=\sum_{\omega^{\prime} \in \mathbb{N}^{n} \text { s.t. }\left\|\omega^{\prime}\right\|_{1}<\|\omega\|_{1}} c_{\omega^{\prime}} \partial^{\omega^{\prime}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)+c^{*} x_{n}^{d^{*}} \partial^{\omega} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right) \tag{A.1}
\end{equation*}
$$

where $c_{\omega^{\prime}}$ is a Laurent polynomial in $x_{n}, c^{*} \in \mathbb{R}$ and $d^{*} \in \mathbb{Z}$. Note that substitution of $\bar{r}$ into (A.1) is always well-defined because only $x_{n}$ appears in denominators and $\bar{r}_{n}$ is non-zero. If $p$ has order $t+1$ at $r$, then for any multi-index $\omega$ such that $\|\omega\|_{1} \leq t$, (A.1) shows us that $\partial^{\omega} \bar{p}$ is a sum of terms of the form $\left(\partial^{\rho} p\right)\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)$, where $\|\rho\|_{1} \leq t$, which evaluates at $\bar{r}$ to exactly $\left(\partial^{\rho} p\right)(r)$. Therefore, all terms of (A.1) vanish and we get that $\partial^{\omega} \bar{p}(\bar{r})$ is zero.

So what remains to be proven is (A.1), which we prove by induction on $t=\|\omega\|_{1}$. When $t=0$, we have that $\bar{p}=x_{n}^{d} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)$, which is of the form required by (A.1) - the $c_{\omega^{\prime}}$ s are all zero, $c^{*}=1$ and $d^{*}=d$. So assume (A.1) holds for all values up to and including $t$ and consider a multi-index $\omega$ such that $\|\omega\|_{1}=t$. Let $\mu_{i}$ be the $n$-vector that is 1 in the $i$ th component and zero everywhere else. We will show that for any $i$, (A.1) holds for $\partial^{\omega+\mu_{i}}$.

First assume $i<n$. By induction we have

$$
\begin{align*}
\partial^{\omega+\mu_{i}} \bar{p} & = \\
\frac{\partial}{\partial x_{i}} \partial^{\omega} \bar{p} & =\frac{\partial}{\partial x_{i}}\left(\sum_{\omega^{\prime}} \mid\left\|\omega^{\prime}\right\|_{1}<\|\omega\|_{1} c_{\omega^{\prime}} \partial^{\omega^{\prime}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)+c^{*} x_{n}^{d^{*}} \partial^{\omega} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)\right) \\
& =\sum_{\omega^{\prime}} \mid\left\|\omega^{\prime}\right\|_{1}<\|\omega\|_{1} c_{\omega^{\prime}} \partial^{\omega^{\prime}+\mu_{1}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)+c^{*} x_{n}^{d^{*}} \partial^{\omega+\mu_{1}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)  \tag{A.2}\\
& =\sum_{\omega^{\prime}} \mid\left\|\omega^{\prime}\right\|_{1}<\left\|\omega+\mu_{1}\right\|_{1} c_{\omega^{\prime}} \partial^{\omega^{\prime}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)+c^{*} x_{n}^{d^{*}} \partial^{\omega+\mu_{1}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)
\end{align*}
$$

which is of the required form. Second assume $i=n$. By induction we have

$$
\begin{align*}
\partial^{\omega+\mu_{n}} \bar{p}= & \\
\frac{\partial}{\partial x_{n}} \partial^{\omega} \bar{p}= & \frac{\partial}{\partial x_{n}}\left(\sum_{\omega^{\prime}} \mid\left\|\omega^{\prime}\right\|_{1}<\|\omega\|_{1} c_{\omega^{\prime}} \partial^{\omega^{\prime}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)+c^{*} x_{n}^{d^{*}} \partial^{\omega} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)\right) \\
= & \sum_{\omega^{\prime}} \left\lvert\,\left\|\omega^{\prime}\right\|_{1}<\|\omega\|_{1}\left(\left(\frac{d}{d x_{n}} c_{\omega^{\prime}}\right) \partial^{\omega^{\prime}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)+c_{\omega^{\prime}}\left(\partial^{\omega^{\prime}+\mu_{n}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)\right) \frac{-1}{x_{n}^{2}}\right)\right. \\
& + \\
& c^{*} d^{*} x_{n}^{d^{*}-1} \partial^{\omega} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)+c^{*} x_{n}^{d^{*}} \partial^{\omega+\mu_{n}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right) \frac{-1}{x_{n}^{2}} \tag{A.3}
\end{align*}
$$

In the final line of (A.3) we see that the last term, $-c^{*} x_{n}^{d^{*}-2} \partial^{\omega+\mu_{n}} p\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)$, is the only order $t+1$ derivative term, and it has the proper form. All other terms are of the form $c \partial^{\omega^{\prime}}\left(x_{1}, \ldots, x_{n-1}, 1 / x_{n}\right)$, where $c$ is a Laurent polynomial in $x_{n}$ and $\left\|\omega^{\prime}\right\|_{1}<t+1$. Since Laurent polynomials are closed under addition, this sum has the required form.

Theorem A.2. Let $p_{1}, p_{2} \in \mathbb{R}\left[x_{1}, \ldots, x_{i}\right]$ of level $i$. Let $R$ be a connected analytic submanifold in $\mathbb{R}^{i-1}$ on which $p_{1}$ and $p_{2}$ are degree-invariant and non-nullified, and on which $p_{r}=\operatorname{res}_{x_{i}}\left(p_{1}, p_{2}\right)$ is order-invariant. Let $\theta_{1}$ be a real root function of $p_{1}$ and $\theta_{2}$ be a real root function of $p_{2}$, both defined on $R$. If there is a point $r \in R$ at which the values of $\theta_{1}$ and $\theta_{2}$ differ, then the values of $\theta_{1}$ and $\theta_{2}$ differ throughout $R$.

Proof. We prove this theorem by contradiction. Suppose there are points at which $\theta_{1}$ and $\theta_{2}$ have the same value. Choose such a point $r^{\prime} \in R$ and path $\pi:[0,1] \rightarrow R$ such that $\pi(0)=r, \pi(1)=r^{\prime}$ and for all $x \in(0,1)$ we have $\theta_{1}(\pi(x)) \neq \theta_{2}(\pi(x))$. Note that we are guaranteed to be able to find such $r^{\prime}$ and $\pi$ by the continuity of $\theta_{1}$ and $\theta_{2}$.

Let $U$ be a neighbourhood of $r^{\prime}$ and $V$ a neighbourhood of the origin, both within $\mathbb{R}^{i-1}$. Denote by $d$ the dimension of $R$. We now refer to the concept of a coordinate system, as described in [31]. By Theorem 2.2.1 of [31], there is a coordinate system $\phi$ mapping $U$ to $V$ such that $R \cap U=\left\{x \in U \mid \phi_{d+1}(x)=0, \ldots, \phi_{i-1}(x)=0\right\}$. In other words, the image of $R \cap U$ is the set of points in $V$ with zeros for the last $n-1-d$ coordinates, which we will refer to as $S$. We view $p_{1}$ and $p_{2}$ in this new coordinate system as $\overline{p_{1}}=p_{1}\left(\phi^{-1}\left(x_{1}, \ldots, x_{i-1}\right), x_{i}\right)$ and $\overline{p_{2}}=p_{2}\left(\phi^{-1}\left(x_{1}, \ldots, x_{i-1}\right), x_{i}\right)$, respectively. Note that for any $a \in U, p_{1}$ evaluated at $a$ gives the same univariate polynomial in $x_{i}$ as $\overline{p_{1}}$ evaluated at $\phi(a)$ (as is also true for $p_{2}$ ). In particular, since $p_{1}$ and $p_{2}$ are degree-invariant in $R, \overline{p_{1}}$ and $\overline{p_{2}}$ are degree-invariant in $S$, moreover, the degrees of $p_{1}$ and $\overline{p_{1}}$ are the same, as are the degrees of $p_{2}$ and $\overline{p_{2}}$. Because $\phi^{-1}$ only acts on the coefficients of $p_{1}$ and $p_{2}$ (as polynomials in $x_{i}$ ), and because the degrees remain unchanged, the resultant construction commutes with respect to $\phi^{-1}$. In other words, recalling that $p_{r}=\operatorname{res}_{x_{i}}\left(p_{1}, p_{2}\right)$, we have $p_{r} \circ \phi^{-1}=\operatorname{res}_{x_{i}}\left(\overline{p_{1}}, \overline{p_{2}}\right)$. Moreover, by the "Remark" on p .45 of [31], the order of $p_{r}$ at point $a$ is the same as the order of $p_{r} \circ \phi^{-1}$ at $\phi(a)$, for any $a \in U$. This means that $p_{r}$ is order-invariant in $U \cap R$ if and only if $p_{r} \circ \phi^{-1}$ is order-invariant in $S$. We will show that the order-invariance of $\operatorname{res}_{x_{i}}\left(\overline{p_{1}}, \overline{p_{2}}\right)$ in $S$ contradicts the assumption that $\theta_{1}\left(r^{\prime}\right)=\theta_{2}\left(r^{\prime}\right)$.

For convenience, let $f=\theta_{1} \circ \phi^{-1}$ and $g=\theta_{2} \circ \phi^{-1}$, noting that $f$ and $g$ are themselves analytic functions and thus proper real-root functions for $\overline{p_{1}}$ and $\overline{p_{2}}$, and that they agree in value at $\phi\left(r^{\prime}\right)$. The function $f$ defines a root of $\overline{p_{1}}$ on $S$, not on all of $V$. Define $f^{*}\left(y_{1}, \ldots, y_{i-1}\right)=f\left(y_{1}, \ldots, y_{s}, 0, \ldots, 0\right)$, which extends the domain of $f$ to all of $V$, and define $g^{*}$ analogously. Note that $f^{*}$ and $g^{*}$ are analytic functions of $V$. [This is actually the key to this proof. Had we tried in the same way to extend $\theta_{1}$ and $\theta_{2}$ from functions of $R$ to functions of $U$, we would not have been guaranteed to get analytic functions.] By division we get $\overline{p_{1}}=\left(x-f^{*}\right) q_{1}+\overline{p_{1}}\left(f^{*}\right)$. The remainder, $\overline{p_{1}}\left(f^{*}\right)$, is zero throughout $S$ because $f$, which $f^{*}$ extends, is a root function of $\overline{p_{1}}$ on $S$. However, since the value of $f^{*}$ is independent of $y_{s+1}, \ldots, y_{i-1}$, the remainder $\overline{p_{1}}\left(f^{*}\right)$ is actually zero throughout $V$. Thus, $\overline{p_{1}}$ factors as $\overline{p_{1}}=\left(x-f^{*}\right) q_{1}$ over $V$. By the same logic, we get the factorization $\overline{p_{2}}=\left(x-f^{*}\right) q_{2}$. Note that the coefficients of $q_{1}$ are all polynomial in $f^{*}$, so all coefficients are analytic functions of $V$.

It's shown in [13], Theorem 1, Chapter Computing in Algebraic Extensions, that res $(A, B)=a_{m}^{n} \prod_{i=1}^{m} B\left(r_{i}\right)$, where $m$ is the degree of $A, a_{m}$ is the leading coefficient of $A$, the $r_{i}$ s are the $m$ complex roots of $A$, and $n$ is the degree of $B$. It easily follows that
$\operatorname{res}(A, B C)=\operatorname{res}(A, B) \cdot \operatorname{res}(A, C)$, from which we get:

$$
\operatorname{res}_{x_{i}}\left(\left(x_{i}-f^{*}\right) q_{1},\left(x_{i}-g^{*}\right) q_{2}\right)=\operatorname{res}_{x_{i}}\left(x_{i}-f^{*}, x_{i}-g^{*}\right) \cdot \operatorname{res}_{x_{i}}\left(q_{1}, x_{i}-g^{*}\right) \cdot \operatorname{res}_{x_{i}}\left(x_{i}-f^{*}, q_{2}\right) \cdot \operatorname{res}_{x_{i}}\left(q_{1}, q_{2}\right)
$$

For any analytic function, if there are points of order $k$ and also of order $k+j$, the set of points of order at least $k+j$ is a closed subset of the set of all points of order at least $k$. This means that if the order of a product of analytic functions is constant in $S$ then the orders of each of the individual factors must be constant as well. Thus, since the order of $\operatorname{res}_{x_{i}}\left(\overline{p_{1}}, \overline{p_{2}}\right)$ is constant in $S$, the order of $\operatorname{res}_{x_{i}}\left(x_{i}-f^{*}, x_{i}-g^{*}\right)$, which we note is $f^{*}-g^{*}$, is
constant as well. There are points in $R$ arbitrarily close to $r^{\prime}$ at which $\theta_{1}$ and $\theta_{2}$ are unequal. Let $r^{\prime \prime}$ be such a point, sufficiently close such that $\phi\left(r^{\prime \prime}\right) \in S$. Then at $\phi\left(r^{\prime \prime}\right)$, the functions $f^{*}$ and $g^{*}$ are unequal. Thus the order of $\operatorname{res}_{x_{i}}\left(x_{i}-f^{*}, x_{i}-g^{*}\right)$ is zero at $\phi\left(r^{\prime \prime}\right)$, which means its order is zero everywhere in $S$. So $f^{*} \neq g^{*}$ for every point in $S$. However, since $\theta_{1}$ and $\theta_{2}$ are equal at $r^{\prime}, f^{*}$ and $g^{*}$ are equal at $\phi\left(r^{\prime}\right)$, which is a contradiction.

## B. Correctness of the theorems of projective delineability

Lemma 4.1. Its proof relies on the more technical Theorem A. 1 from Appendix A.
Proof. If $\operatorname{ldcf}_{x_{i+1}}(p)$ is sign-invariant in $R$, then $p$ is analytically delineable on $R$ by Theorem 3.1 in [6] and Theorem 2 of [33]. Thus, we assume that $\operatorname{ldcf}_{x_{i+1}}(p)$ is not sign-invariant in $R$. By Theorem A.1, for any point $r$ in $R$, there is a neighbourhood $N_{r}$ around $r$ such that all the properties required for analytic projective delineability hold in $N_{r} \cap R$. Since $R$ is connected, this suffices to prove the lemma. To see why, consider two points in $R, r$ and $r^{\prime}$. Since $R$ is connected there is a path (a continuous function with domain $[0,1]$ ) $L:[0,1] \rightarrow R$ such that $L(0)=r$ and $L(1)=r^{\prime}$. According to Theorem A.1, in some neighbourhood containing $L(0)$ the definition of projective delineability is satisfied, i.e. there is some $k$ and root functions $\theta_{1}, \ldots, \theta_{k}, \theta_{*}$ with all the desired properties. We need to prove that the root functions can be extended to cover the entire path while maintaining those properties. Let $c$ be a real number in the range $(0,1)$ such that $\theta_{1}, \ldots, \theta_{k}, \theta_{*}$ can be extended over the subpath $L^{\prime}=\{L(\beta) \mid \beta \in[0, c)\}$. By Theorem A. 1 there is a neighbourhood $N_{L(c)}$ in which the properties required for projective delineability hold, i.e. there is some $k^{\prime}$ and root functions $\theta_{1}^{\prime}, \ldots, \theta_{k^{\prime}}^{\prime}, \theta_{*}^{\prime}$ with all the desired properties. However, because $N_{L(c)} \cap L^{\prime} \neq \emptyset, k^{\prime}=k$ and the functions $\theta_{1}^{\prime}, \ldots, \theta_{k^{\prime}}^{\prime}, \theta_{*}^{\prime}$ agree with $\theta_{1}, \ldots, \theta_{k}, \theta_{*}$ on the path up to $L(c)$. Moreover, choosing a value $c^{\prime}$ greater than $c$ such that $L\left(c^{\prime}\right) \in N_{L(c)}$, they provide an extension of those functions through $L^{\prime \prime}=\left\{L(\beta) \mid \beta \in\left[0, c^{\prime}\right)\right\}$. Thus, the functions we require can be extended to cover the whole path $L$ from $r$ to $r^{\prime}$, maintaining all the properties required for projective delineability. In fact, the same argument shows that $p$ will be order-invariant in the sections defined by the root functions.

## Lemma 4.2.

Proof. If $\operatorname{ldcf}_{x_{i+1}}(p)(r)=0$ for all $r \in R$, then there is no $\theta_{*}$ root function for the definition of projective delineable, meaning the definition collapses to that of ordinary delineability. Otherwise $\operatorname{ldcf}_{x_{i+1}}(p)(r) \neq 0$ for all $r \in R$, which means that there is a $\theta_{*}$ root function, but it is defined everywhere on $R$. As $R$ is connected, the definition of projective delineability collapses once again to that of ordinary delineability.

## Lemma 4.3.

Proof. If $p$ is analytically projectively delineable on $R \downarrow_{[i-1]}$, then by Theorem A.1, $p$ is order-invariant in each $R \downarrow_{[i-1]} \times \theta_{1}(s), \ldots, R \downarrow_{[i-1]} \times \theta_{k}$ and each $R \downarrow_{[i-1]} \times \theta_{*}^{\prime}$ where $\theta_{*}^{\prime}$ is a restriction of $\theta_{*}$ to a connected component of its domain. As $R$ is a subset of one of these sets by definition, $p$ is order-invariant on $R$.

## Lemma 4.4.

Proof. The hypotheses state that $\theta_{1}$ and $\theta_{2}$ are real root functions over $R$, which means that neither can be $\theta_{*}$ in Definition 4.1, the section that is not defined over the whole of $R$, and that both are continuous and analytic. If the two functions are identical over $R$, then the theorem holds. So assume they are not, i.e. assume there is a point $r \in R$ at which the two functions have different values. We will show that the two functions differ everywhere in $R$ which, because they are continuous, proves the theorem.

Suppose, by way of contradiction, that there are points in $R$ at which $\theta_{1}$ and $\theta_{2}$ are equal. Choose such a point, $s$, and path $L:[0,1] \rightarrow R$ such that $L(0)=r, L(1)=s$ and for all $x \in[0,1)$ we have $\theta_{1}(L(x)) \neq \theta_{2}(L(x))$. Note that we are guaranteed to be able to find such $s$ and $L$ by the continuity of $\theta_{1}$ and $\theta_{2}$. Choose a value $\gamma \in \mathbb{R}$ that, over $s$, is not a zero of $p_{1}$ neither of $p_{2}$. Let $N$ be a neighbourhood of $s$ over which neither $p_{1}$ nor $p_{2}$ are zero at $\gamma$. Let $U$ be the connected component of $N \cap R$ that contains $s$. There must be a value $\mu \in(0,1)$ such that $\{L(x) \mid x \in(\mu, 1]\} \subseteq U$. Set $r^{\prime}=L((\mu+1) / 2)$, noting that $\theta_{1}\left(r^{\prime}\right) \neq \theta_{2}\left(r^{\prime}\right)$. Let $p_{1}^{*}=$ $p_{1}\left(x_{1}, \ldots, x_{i}, x_{i+1}-\gamma\right)$ and $p_{2}^{*}=p_{2}\left(x_{1}, \ldots, x_{i}, x_{i+1}-\gamma\right)$. By [24], the resultant is invariant under this transformation. Let $\overline{p_{1}}=x_{i+1}^{d_{1}} p_{1}^{*}\left(x_{1}, \ldots, x_{i}, 1 / x_{i+1}\right)$ and $\overline{p_{2}}=x_{i+1}^{d_{2}} p_{2}^{*}\left(x_{1}, \ldots, x_{i}, 1 / x_{i+1}\right)$, where the $d_{1}=\operatorname{deg}_{x_{i+1}}\left(p_{1}\right)$ and $d_{2}=$ $\operatorname{deg}_{x_{i+1}}\left(p_{2}\right)$. The resultant is unchanged by this transformation, once again by [24], so $\operatorname{res}_{x_{i+1}}\left(\overline{p_{1}}, \overline{p_{2}}\right)$ is orderinvariant in $U$. Moreover, since neither $p_{1}$ nor $p_{2}$ is nullified in $U, \overline{p_{1}}$ and $\overline{p_{2}}$ are non-nullified in $U$. The proof of Theorem A. 1 examines this same transformation, and shows that the root functions of $p_{1}$ and $p_{2}$ map to the root functions of $\overline{p_{1}}$ and $\overline{p_{2}}$ (with, of course, some special caveats for the $\theta^{*}$ section, which don't apply to $\theta_{1}$ and $\theta_{2}$ ).

The leading coefficients of $\overline{p_{1}}$ and $\overline{p_{2}}$ are non-vanishing in $U$, and thus $\overline{p_{1}}$ and $\overline{p_{2}}$ are degree-invariant. Let $\overline{\theta_{1}}$ and $\overline{\theta_{2}}$ be the root functions of $\overline{p_{1}}$ and $\overline{p_{2}}$ corresponding to $\theta_{1}$ and $\theta_{2}$. By Theorem A. 2 The fact that $\overline{\theta_{1}}$ and $\overline{\theta_{2}}$ are unequal at $r^{\prime}$ means that they are unequal throughout $U$. However, this implies they are unequal at $s$, which is a contradiction.

## C. Correctness of the proof system

Lemma C.1. Rule 5.1 is correct.
Note that throughout this section, when proving the correctness of a mapping, we implicitly use the assumptions from the mapping's definition in the proofs.

Proof. Let $R \subseteq \mathbb{R}^{i}$ be a connected analytic submanifold such that $s \in R$. If $p$ is not nullified on any point in $R$ and $p$ is order-invariant on $R$, then $p$ is analytically projectively delineable on $R$ by Lemma 4.1.

Lemma C.2. Rule 5.2 is correct.
Proof. The first rule follows from the definition of the discriminant by the Sylvester matrix. For the second rule, observe that $p$ is nullified on any point $r \in R$ if and only if $c_{j}(r)=0$ for all $j \in[m]$. Thus, statement follows immediately. See also [11, Lemma 5].

Lemma C.3. Rule 5.3 is correct.
Proof. Trivial.
Lemma C.4. Rule 5.4 is correct.
Proof. Let $R \subseteq \mathbb{R}^{i}$. Note that if $p$ is reducible, then $p \notin$ factors $(p)$. If $p$ is reducible and every $q \in \operatorname{factors}(p)$ is order-invariant (sign-invariant) on $R$, then $p$ is order-invariant (sign-invariant) on $R$ by [31, Lemma 3.2.2].

Lemma C.5. Rule 5.5 is correct.
Proof. Let $R \subseteq \mathbb{R}^{i}$. If $p(r) \neq 0$ for all $r \in R$, then $p$ is order-invariant on $R$ as an immediate consequence of the definition of order-invariance.

Let $R \subseteq \mathbb{R}^{i}$ be a cell such that $R \downarrow_{[i-1]}$ is a connected analytic submanifold and $s \in R$. If $p(r)=0$ for all $r \in R$ and $p$ is analytically projectively delineable on $R \downarrow_{[i-1]}$, then $p$ is order-invariant by Lemma 4.3.

Lemma C.6. Rule 5.6 is correct.
Proof. Let $R \subseteq \mathbb{R}^{i}$ such that $s \in R$.
Assume $p$ is (analytically) projectively delineable on a connected superset $R^{\prime} \supseteq R$ and $\operatorname{realRoots}\left(p\left(s, x_{i+1}\right)\right)=\emptyset$, then by Definition 4.1 either $p\left(r, r^{\prime}\right) \neq 0$ for all $r \in R^{\prime}, r^{\prime} \in \mathbb{R}$ or the graph of $p$ over $R^{\prime}$ is given by a function $\theta_{*}: R^{\prime} \backslash\left\{r \in \mathbb{R}^{i} \mid \operatorname{ldcf}_{x_{i+1}}(p)(r)=0\right\} \rightarrow \mathbb{R}$.

If $\operatorname{ldcf}_{x_{i+1}}(p)(s) \neq 0$, then the second statement cannot hold, thus $p$ is delineable on $R^{\prime}$ without any real roots on $R^{\prime}$. Otherwise, if $\operatorname{ldcf}_{x_{i+1}}(p)$ is sign-invariant on $R^{\prime}$, by $\operatorname{ldcf}_{x_{i+1}}(p)(s)=0$ follows $\operatorname{ldcf}_{x_{i+1}}(p)(r)=0$ for all $r \in R^{\prime}$ consequently $\theta_{*}$ is not defined on $R^{\prime}$; and again, it follows that $p$ is delineable on $R^{\prime}$ without any real roots on $R^{\prime}$.

If $p$ is delineable on $R^{\prime}$ without any real roots on $R^{\prime}$, then $p$ is sign-invariant on $R^{\prime} \times \mathbb{R}$.
Lemma C.7. Rule 5.7 is correct.
Proof. As the endpoints of I are described by analytic functions, the statement follows immediately from [31, Theorem 2.2.3 and Theorem 2.2.4].
Lemma C.8. Rule 5.8 is correct.
Proof. Let $R \subseteq \mathbb{R}^{i}$ such that its projection $R \downarrow_{[i-1]}$ on $\mathbb{R}^{i-1}$ is a connected analytic submanifold. If $b$. $p$ is analytically projectively delineable on $R \downarrow_{[i-1]}$ and $\operatorname{ldcf}_{x_{i}}(b . p)$ is sign-invariant on $R \downarrow_{[i-1]}$, then by Lemma 4.2, b.p is analytically delineable on $R \downarrow_{[i-1]}$. Furthermore, as $\operatorname{repr}(\mathrm{I}, s)$ holds in $R$, it holds $b . p(r)=0$ for all $r \in R$ and that $R$ is connected.

Now, if $p=b . p$, then $p$ is trivially sign-invariant on $R$. Otherwise, by the order-invariance of $\operatorname{res}_{x_{i}}(b . p, p)$ on $R$ and [34, Theorem 2.2], $p$ is sign-invariant on $R$.

Lemma C.9. Rule 5.9 is correct.

Proof. Let $\preceq$ be an indexed root ordering on some $\Xi=\operatorname{dom}(\preceq)$ for some $s \in \mathbb{R}^{i}$. Let $R \subseteq \mathbb{R}^{i}$ be a connected analytic submanifold such that $s \in R, \xi \cdot p$ is analytically projectively delineable on $R$ for all $\xi \in \Xi$ and $\operatorname{res}_{x_{i+1}}\left(\xi \cdot p, \xi^{\prime} . p\right)$ is order-invariant on $R$ for all $\xi \preceq \xi^{\prime}$.

For $\xi \preceq \xi^{\prime}$, let $S:=\operatorname{dom}\left(\theta_{\xi, s}\right) \cap \operatorname{dom}\left(\theta_{\xi^{\prime}, s}\right) \cap R$. Either $\xi . p=\xi^{\prime} . p$, then the statement holds by definition of projective delineability (Definition 4.1), or $\operatorname{res}_{x_{i+1}}\left(\xi \cdot p, \xi^{\prime} \cdot p\right)$ is order-invariant on $R \supseteq S$, then statement follows immediately by Lemma 4.4.

So now let $\xi \preceq^{t} \xi^{\prime}$ (where $\preceq^{t}$ is the transitive closure of $\preceq$ ), then there exist $\xi_{1}, \ldots, \xi_{k}$ such that $\xi=\xi_{1} \preceq \ldots \preceq$ $\xi_{k}=\xi^{\prime}$. Here we will prove that the statements hold by induction, with the base case in the preceding paragraph.

Assume the inductive hypothesis that the statement holds for $\xi_{1}$ and $\xi_{j}, j<k$. Let $S:=\operatorname{dom}\left(\theta_{\xi_{1}, s}\right) \cap$ $\operatorname{dom}\left(\theta_{\xi_{j+1}, s}\right) \cap R$ and $S^{\prime}:=\operatorname{dom}\left(\theta_{\xi_{j}, s}\right) \cap S$. Note that the statements hold for $\xi_{j}$ and $\xi_{j+1}$ on $S^{\prime}$. If $S=S^{\prime}$, then they also hold for $\xi_{1}$ and $\xi_{j+1}$ on $S$. Finally, assume for a contradiction that $S^{\prime} \subset S$. Then $\theta_{\xi_{j}, s}$ is the " $\theta_{*}$ " root function of $\xi_{j} . p$, and thus tends to $-\infty$ or $\infty$ at the borders of $S^{\prime}$. Hence by continuity it intersects either with $\theta_{\xi_{1}, s}$ in the first case or $\theta_{\xi_{j+1}, s}$ in the second case before it disappears. The first case is excluded by induction hypothesis. The second case is excluded by the order-invariance of $\operatorname{res}_{x_{i+1}}\left(\xi_{j} . p, \xi_{j+1} \cdot p\right)$ on $R \supseteq S$. Thus, this is a contradiction and the statement also holds for $\xi_{1}$ and $\xi_{j+1}$. The induction is complete.

Lemma C.10. Rule 5.10 is correct.
Proof. Let $R \subseteq \mathbb{R}^{i}, s \in R$ and $R^{\prime} \supseteq R$ be a cell such that $p$ is analytically projectively delineable on $R^{\prime}$.
By Definition 4.2, the variety of $p$ on $R^{\prime}$ can be described by analytic $\theta_{1}, \ldots, \theta_{k}, \theta_{*}$ where $\operatorname{dom}\left(\theta_{j}\right)=R^{\prime}$ for all $j \in[1 . . k]$ and $\operatorname{dom}\left(\theta_{*}\right) \subseteq R^{\prime}$. If $\xi$ is not the first or last root of $p\left(s, x_{i+1}\right)$, then $\theta_{\xi, s}=\theta_{j}$ for some $j \in[k]$. If $\operatorname{ldcf}_{x_{i+1}}(p)(s)=0$, then $\theta_{*}$ is not defined at $s$, thus $\theta_{\xi, s}=\theta_{j}$ for some $j \in[1 . . k]$.

Otherwise, if additionally $\operatorname{ldcf}_{x_{i+1}}(p)$ is sign-invariant on $R^{\prime}$, then the variety of $p$ on $R^{\prime}$ can be described by $\theta_{1}, \ldots, \theta_{k}$ where $\operatorname{dom}\left(\theta_{j}\right)=R^{\prime}, j \in[1 . . k]$, thus $\theta_{\xi, s}=\theta_{j}$ for some $j \in[1 . . k]$.

Lemma C.11. Rule 5.11 is correct.
Proof. Let $R \subseteq \mathbb{R}^{i}$ such that $R \downarrow_{[i-1]}$ is a connected analytic submanifold, $p$ is analytically projectively delineable on $R \downarrow_{[i-1]}, s \in R \downarrow_{[i-1]}, \operatorname{repr}(\mathrm{I}, s)$ holds on $R$ and $\operatorname{ir\_ ord}(\preceq, s)$ holds on $R \downarrow_{[i-1]}$.

First note that as $\operatorname{repr}(\mathrm{I}, s)$ holds on $R \downarrow_{[i-1]}$, it follows that if $l \neq-\infty$, then $\theta_{l, s}$ exists on $R \downarrow_{[i-1]}$, and if $u \neq \infty$, then $\theta_{u, s}$ exist on $R \downarrow_{[i-1]}$.

- If $p(s, b(s))=0$, then there exists $\xi \in \operatorname{realRoots}\left(p\left(s, x_{i}\right)\right)$ such that $\xi(s)=b(s)$. By repr $(\mathrm{I}, s)$ the root function $\theta_{b, s}$ exists over $R \downarrow_{[i-1]}$. We show that $\theta_{\xi, s}$ exists on $R \downarrow_{[i-1]}$ (according to Definition 4.4) and $\theta_{\xi, s}=\theta_{b, s}$ on $R \downarrow_{[i-1]}$ :
We prove the first by contradiction: Assume that for the domain $S$ of $\theta_{\xi, s}$ it holds $S \cap R \downarrow_{[i-1]} \subset R \downarrow_{[i-1]}$. As $p$ is projectively delineable, then $\theta_{\xi, s}$ is the $\theta_{*}$ function and thus tends to $-\infty$ or $\infty$ at at least one of the boundaries of $S$. By continuity, this would imply $\theta_{\xi, s}(r) \neq \theta_{b, s}(r)$ for some $r \in S \subset R \downarrow_{[i-1]}$, which is a contradiction to the assumption that $\operatorname{ir} \_\operatorname{ord}(\preceq, s)$ holds on $R \downarrow_{[i-1]}$.
Thus, $\theta_{\xi, s}$ exists on $R \downarrow_{[i-1]}$, and by the property $\operatorname{ir\_ ord}(\preceq, s)$ it follows that $\theta_{\xi, s}=\theta_{b, s}$ on $R \downarrow_{[i-1]}$.
Finally, as $\operatorname{repr}(\mathrm{I}, s)$ holds on $R$, it follows $\left(r, \theta_{\xi, s}(r)\right) \in R$ for all $r \in R \downarrow_{[i-1]}$; in other words, $p$ is sign-invariant on $R$.
- If $l \neq u$ or $p(s, b(s)) \neq 0$, then:
- If $\operatorname{ldcf}_{x_{i}}(p)$ is sign-invariant on $R \downarrow_{[i-1]}$, then by Lemma $4.2 p$ is delineable on $R \downarrow_{[i-1]}$. Thus, there exist real root functions $\theta_{1}, \ldots, \theta_{k}: R \downarrow_{[i-1]} \rightarrow \mathbb{R}$ of $p$ on $R \downarrow_{[i-1]}$ covering all real roots of $p$ on $R \downarrow_{[i-1]}$; and as $R \downarrow_{[i-1]}$ is connected, the properties $\operatorname{ir}_{-} \operatorname{ord}(\preceq, s)$ and $\operatorname{repr}(I, s)$ on $R \downarrow_{[i-1]}$ and the note above yield $\left(r, \theta_{j}(r)\right) \notin R$ for all $r \in R \downarrow_{[i-1]}$.
- Now assume $l \neq-\infty$ and $u \neq \infty$ and $\operatorname{ldcf}_{x_{i}}(p)$ is not sign-invariant on $R \downarrow_{[i-1]}$. By analytic projective delineability of $p$, there exist real root functions $\theta_{1}, \ldots, \theta_{k}: R \downarrow_{[i-1]} \rightarrow \mathbb{R}$ of $p$ on $R \downarrow_{[i-1]}$ and a function $\theta_{*}: R \downarrow_{[i-1]} \backslash$ realRoots $\left(\operatorname{ldcf}_{x_{i}}(p)(r)\right) \rightarrow \mathbb{R}$ covering all real roots of $p$ on $R \downarrow_{[i-1]}$.
As $R \downarrow_{[i-1]}$ is connected, the properties $\operatorname{ir}_{-} \operatorname{ord}(\preceq, s)$ on $R \downarrow_{[i-1]}$ and $\operatorname{repr}(\mathrm{I}, s)$ on $R$ and the note above yield immediately $\left(r, \theta_{j}(r)\right) \notin R$ for all $r \in R \downarrow_{[i-1]}$ and $j \in[1 . . k]$. It remains to show that $\left(r, \theta_{*}(r)\right) \notin R$ for all $r \in R \downarrow_{[i-1]} \backslash \operatorname{realRoots}\left(\operatorname{ldcf}_{x_{i}}(p)(r)\right)$. We split $R \downarrow_{[i-1]}$ up into maximal cells in which $\operatorname{ldcf}_{x_{i}}(p)$ is sign-invariant. Let $S$ be any such cell, and note that $S$ is an analytic submanifold by [31, Theorem 2.2.3 and Theorem 2.2.4] (see also Rule 5.7). We need to prove that either $S \cap \operatorname{dom}\left(\theta_{*}\right)=\emptyset,\left.\theta_{*}\right|_{S} \leq \theta_{l, s}$ or $\theta_{u, s} \leq\left.\theta_{*}\right|_{S}$ over $S$. Assume $S \cap \operatorname{dom}\left(\theta_{*}\right) \neq \emptyset$.
* If there exists a $\xi \in \operatorname{ir} \operatorname{Expr}(p, s)$ with $\xi \preceq^{t} l$ (i.e. $\left.\xi(s) \leq l(s)\right)$ such that $\theta_{\xi, s}$ exists on $R \downarrow_{[i-1]}$ (according to Definition 4.4), then $\theta_{\xi, s}=\theta_{j}$ for some $j \in[1 . . k]$ and $\theta_{j} \leq \theta_{l, s}$ on $R \downarrow_{[i-1]}$. By projective delineability, either $\left.\theta_{*}\right|_{S}<\theta_{j} \leq \theta_{l, s}$ or $\theta_{j}<\left.\theta_{*}\right|_{S}$ on $S$.
* Analogously, if $\xi \in \operatorname{ir} \operatorname{Expr}(p, s)$ with $u \preceq^{t} \xi$ such that such that $\theta_{\xi, s}$ exists on $R \downarrow_{[i-1]}$, there exists a $j \in[1 . . k]$ such that either $\theta_{u, s} \leq \theta_{j}<\left.\theta_{*}\right|_{S}$ or $\left.\theta_{*}\right|_{S}<\theta_{j}$ on $S$.
* If there exists a $\xi^{\prime} \in \operatorname{dom}(\preceq) \backslash \operatorname{ir} \operatorname{Expr}(p, s)$ with $\xi^{\prime} \preceq^{t} l$ (i.e. $\left.\xi^{\prime}(s) \leq l(s)\right)$ such that $\theta_{\xi^{\prime}, s}$ exists on $R \downarrow_{[i-1]}$ (according to Definition 4.4), then $\theta_{\xi^{\prime}, s} \leq \theta_{l, s}$ on $R \downarrow_{[i-1]}$. If additionally $\operatorname{res}_{x_{i}}\left(p, \xi^{\prime} \cdot p\right)$ is order-invariant on $R \downarrow_{[i-1]}$ and $\theta_{*}$ tends to $-\infty$ at a boundary of $S$, then by Lemma 4.4 and continuity it holds either $\theta_{*}<\theta_{l, s}$ or $\theta_{*}=\theta_{l, s}$ on $S$.
* Analogously, if there exists a $\xi^{\prime} \in \operatorname{dom}(\preceq) \backslash \operatorname{ir} \operatorname{Expr}(p, s)$ with $u \preceq^{t} \xi^{\prime}$ such that $\theta_{\xi^{\prime}, s}$ exists on $R \downarrow_{[i-1]}$, $\operatorname{res}_{x_{i}}\left(p, \xi^{\prime} \cdot p\right)$ is order-invariant on $R \downarrow_{[i-1]}$ and $\theta_{*}$ tends to $\infty$ at a boundary of $S$, then it holds either $\theta_{u, s}<\theta_{*}$ or $\theta_{u, s}=\theta_{*}$ on $S$.
The above cases in combination with the fact that $\theta_{*}$ either tends to $-\infty$ or $\infty$ at a boundary of $S$ (but not both) yields that $\left.\theta_{*}\right|_{S} \leq \theta_{l, s}$ or $\theta_{u, s} \leq\left.\theta_{*}\right|_{S}$. This concludes our proof.

Lemma C.12. Rule 5.12 is correct.
Proof. Let $R \subseteq \mathbb{R}^{i}$ such that $R \downarrow_{[i-1]}$ is a connected analytic submanifold and let $s \in R \downarrow_{[i-1]}$. Let I be a symbolic interval of level $i$ such that its defining polynomials are not nullified on $s$ and $\operatorname{repr}(\mathrm{I}, s)$ holds on $R$.

Consider the case $\mathrm{I}=($ sector, $l, u), l \neq-\infty, u \neq \infty, l . p \neq u . p$. As $\operatorname{repr}(\mathrm{I}, s)$ holds, $\theta_{l, s}$ and $\theta_{u, s}$ are well-defined on $R \downarrow_{[i-1]}$. As $l \preceq^{t} u$ and $\operatorname{ir\_ ord}(\preceq, s)$ holds on $R \downarrow_{[i-1]}$, it either holds $\theta_{l, s}<\theta_{u, s}$ on $R \downarrow_{[i-1]}$ or $\theta_{l, s}=\theta_{u, s}$ on $R \downarrow_{[i-1]}$. Moreover, by assumption $\operatorname{repr}(\mathrm{I}, s)$ holds on $R$ and $R \downarrow_{[i-1]}$ is connected, thus $R$ is connected (noting that empty sets are connected).

The other cases are similar but simpler.
Lemma C.13. Rule 5.13 is correct.
Proof. This follows from the definitions.
Lemma C.14. Rule 5.14 is correct.
Proof. Let $R \subseteq \mathbb{R}^{i}$ and $R^{\prime} \supseteq R$ be connected such that $p$ is projectively delineable on $R^{\prime}$. Now let $p$ be a polynomial. If $\operatorname{ldcf}_{x_{i+1}}(p)$ is sign-invariant on $R^{\prime}$, by Lemma $4.2 p$ is delineable on $R^{\prime}$. Thus, the graph of $p$ over $R^{\prime}$ is described by real root functions $\theta_{1}<\ldots<\theta_{k}$ on $R^{\prime}$. It follows that for any indexed root expression $\xi$ with $\xi . k \leq k$ it holds $\operatorname{dom}\left(\theta_{\xi, s}\right)=R^{\prime}$ and $\theta_{\xi, s}=\xi$ on $R^{\prime}$.

Now, the result follows immediately from the definition of holds(I) and the previous statement applied on the cell's boundaries.


[^0]:    * Corresponding author

    Email addresses: nalbach@cs.rwth-aachen.de (Jasper Nalbach), abraham@cs.rwth-aachen.de (Erika Ábrahám), philippe.specht@rwth-aachen.de (Philippe Specht), wcbrown@usna.edu (Christopher W. Brown), j.h.davenport@bath.ac.uk (James H. Davenport), matthew.england@coventry.ac.uk (Matthew England)

