

Thesis

Hollie

2024-03-15

Contents

Chapter 1

Cylindrical algebraic decompositions with monotone cells

A Cylindrical Algebraic Decomposition (CAD) is a finite partition of n -dimensional space into so-called Cylindrical Cells. The key idea of cylindricity is that cells are arranged in *stacks*. I.e., the projection of any two cells onto dimension $n-1$ is either disjoint or identical.

CAD provides a constructive and computationally tractable proof of the famous Tarski-Seidenberg theorem, which states that semialgebraic sets are closed under projection, or, equivalently, that quantifier elimination is possible over the reals. Indeed, QE is possibly the most famous application of CAD. CAD also has applications in computing topological properties of semialgebraic sets and solving motion-planning problems. However, for these applications, a basic CAD will not suffice. Instead, we may have to introduce some additional constraints to the construction. One such constraint, which is useful in topology, is that every cell in the CAD is a *topologically regular cell*. Put briefly, its boundary should be homeomorphic to a circle. Another useful property is the *frontier condition*. I.e., the boundary of every cell in the decomposition must be a union of some other cells in the decomposition of smaller dimension. This property is necessary to solve motion planning problems.

The goal of this thesis is to develop an algorithm to construct such a CAD. In fact, a CAD satisfying a slightly stronger condition – such that every cell contained in some input set is **monotone**. It has been proved that a **monotone cell** is a **topologically regular cell**. The constructive proof that such a CAD exists is due to (?), and this thesis will aim to extract an algorithm from the formal description. (?) work over the category of sub-pfaffian sets, using an

algorithm due to **TODO** to construct the base cylindrical decomposition. However, the algorithm presented here will work with semialgebraic sets, using the more familiar CAD due to (?). This will allow the tools of computer algebra to be used. In addition, an implementation would not be possible over sub-pfaffian sets, as an oracle is needed to determine whether a set is empty.

1.1 Things to explain

- some background on CAD
 - algorithms (including pfaffian one), implementations and upper bounds
- monotone cells are obtained by refinements, i.e., more cells
 - why might one want to introduce more cells?
- motivation
 - what can we do with monotone cad that we cannot do with ordinary
 - frontier condition: Schwartz and Sharir and Gabrielov and Vorobjov
 - definitions and motivating examples
 - * start with my favourite example in \mathbb{R}^2 . $x^2 + y^2 < 1$. observe that the curves forming the top and the bottom of the cell are not monotone - they are not even quasi-affine. we can see, by considering the curves as a function of x , that they are not strictly increasing in, strictly decreasing in, or independent of x . from our intuition, it's clear that we can split these curves where they “change direction” – in other words, at their local maxima and minima. this is the basic idea of BGV's algorithm.
 - * note that the example of a subset of \mathbb{R}^2 which is QA but not M is a bit trivial and not instructive (two disjoint increasing / decreasing curves). being not connected, it cannot be a CAD cell. I.e., any CAD cell in \mathbb{R}^2 which is quasi-affine is also monotone. furthermore (see the theorem in QA section) any CAD cell of dimension ≤ 1 which is quasi-affine will be monotone.
 - * let's consider a more interesting example in \mathbb{R}^3 . the smashed fragment of bowl. we see that, in the triangle, the function $z = x^2 + y^2$ is strictly increasing in both y and z . however, if we look at the projection onto the x - z plane, it is clear that this cell is not monotone. in particular, if we intersect with the plane $z = 3/4$, it intersects our cell in two curve intervals. being strictly increasing/decreasing/independent of each variable, it suffices to consider the one-dimensional “top” and “bottom” of each cell (see BGV lemma 3.18) and ensure that these curves are monotone. this amounts, again, to finding the points at which they “change direction”. we will see later that this can be reduced to the problem of finding the local maxima and minima of a surface, subject to constraints defining a curve, which is a well-studied problem.

- * finally, let's see some examples of the frontier condition. things are pretty simple in \mathbb{R}^2 . indeed, Lazard proves that a sign-invariant CAD of \mathbb{R}^2 will always satisfy the frontier condition. (my favourite example, compatible and sign-invariant). However, things become more interesting in \mathbb{R}^3 , where blow-up points may occur. Let's consider another rather famous construction – the Whitney umbrella. Construct a CAD sign-invariant with respect to $x^2 - y^2 z = 0$. we get three cells with sign 0 on $x^2 - y^2 z = 0$ – two 2-dimensional sections and one one-dimensional sector above the origin. consider the frontier of one of these sections – $x=y=0, z>0$. (for both) observe that the frontier of both sections coincides and has non-empty intersection with the one-cell, however it's not a union of cells and, with the polynomials we have, it's not possible to refine our one-cell such that the frontier will be a union. i.e., we need to add more polynomials. also observe that, in \mathbb{R}^3 and, in fact, for a set of dimension 2 in any ambient space, blow-up points (cells above which blow-ups occur) will have dimension at most 0.
- * finally, let's consider a definable example (from BGV) of a subset which cannot satisfy the frontier condition in a simple way. observe that S is a semialgebraic set. note, however, that some projection operators, e.g., brown mc and laz (see section TODO) guarantee smooth cells, so such a construction would not occur in this case. However, the construction is cylindrical, so this is a valid CAD. considering the frontier of S , we see that it is not a topologically regular cell. in particular, the blow-up point at the origin. this means that introducing the top and bottom of S as new cells would not be possible, since the bottom of S is not a valid cylindrical cell. First, we would need to refine the bottom of S such that it is a union of valid cells. i.e., we would need to divide it into the two horizontal curve intervals, and the blow-up interval above the origin. Adding these cells directly to the CAD would destroy the cylindrical structure, so a refinement of some of the induced decompositions below it (starting with the decomposition induced by D on \mathbb{R}^1) will be needed. i.e., we would need to refine D' such that it is compatible with $x < 0$, $x = 0$ and $x > 0$. this is the basis of a novel algorithm (see section ...) for satisfying the frontier condition in the general case.
 - existing work: Schwartz, Lazard and Locatelli
- discussing the frontier condition
 - let's proceed with the construction due to BGV. first they observe that the boundary of every 2-dimensional cell is homeomorphic to a circle. thus, a refinement of this boundary into monotone curve intervals and points can be computed. if the boundary intersects a cell, that cell will be 2-dimensional and we can divide it into two monotone 2-dimensional cells and the one-dimensional component of

the frontier. This procedure is simple, but it has the drawback that we need to find the frontier of a 2-dimensional monotone cell. the usual way of doing this is by solving a QE problem with ??? quantifier alternations. Using the singly exp algorithm from BPR, each frontier computation will have complexity ???. This would be unacceptable in practice.

- instead, we try to turn our attention to Lazard, who presents as construction in \mathbb{R}^3 . since his cells are already assumed to be topologically regular and blow-up points are assumed to be 0-dimensional, it is sufficient to find “bad cells” (cells above which some polynomial vanishes identically) and try to introduce new polynomials which refine the one-dimensional curve above the bad cell and complete the frontier condition. He does this in a much different way to BGV, who are working in the definable (semi-pfaffian) case, by using properties of polynomial ideals. I don’t completely understand what he is doing, but it has something to do with considering the zariski closure of the one-dimensional cells (at the extrema of the monotone cells) with the bad point removed. the zariski closure as we approach this point is part of the frontier of the set, and is the new point we need to add to satisfy the frontier condition.
- i then conjecture, since we have only 2d sets with 0d blow-ups, that we could extend this to higher dimension. in particular, we can assume that only one polynomial will blow-up at any one time (otherwise the blow-up points would have higher dimension) and work our way up the stack of polynomials above a 0-cell until we find one with a blow-up. we now have a version of Lazard’s ideal, but our 0d point lies in a higher dimension. so we can do the same again – consider the zariski closure of the 1d cell approaching the bad point and get the required refinement. the problem here would be proving this correct. i.e., will it find all of the blow-up points? will it even work at all, or is there something he assumed that is only true in the 3d case?
- finally we should present the novel frontier algorithm and make it very clear how we stop infinite returns to higher dimension. the bit about constructing the chain of cad and then reconstructing it the other way up is important, as the refinements we do on the way back up result in the final one (the refinement of the initial cad) satisfying the frontier condition. we finish off by observing that these weren’t refinements at all, since the refined cads actually coincide with the cad’s in the chain that we had in the first place.
- in the algorithm, change semi-monotone to monotone sector

Chapter 2

Introduction

Based on the work of Basu, Gabrielov and Vorobjov (?).

Chapter 3

Background

3.0.1 Sets definable over O-minimal structures

We will be working mostly in the category *Semialgebraic Set* and sometimes in the more general category of sets definable over an O-minimal structures. A semialgebraic subset of \mathbb{R}^n is the set of points in \mathbb{R}^n satisfying a Boolean combination of polynomial equations and inequalities with coefficients in \mathbb{R} .

Definition 3.1. The semialgebraic subsets of \mathbb{R}^n are the smallest class SA_n of subsets of \mathbb{R}^n satisfying the following properties.

1. If $f \in \mathbb{R}[x_1, \dots, x_n]$, then $\{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) = 0\}$ and $\{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) > 0\}$ are elements of SA_n .
2. If $A, B \in SA_n$, then $A \cup B$, $A \cap B$ and $\mathbb{R}^n \setminus A$ are elements of SA_n .

Definition 3.2. Let $X \subset \mathbb{R}^n$ and $Y \subset \mathbb{R}^m$. A mapping $f : X \rightarrow Y$ is called semialgebraic if its graph

$$\{(\mathbf{x}, f(\mathbf{x})) \in \mathbb{R}^{n+m} \mid \mathbf{x} \in X\}$$

is a semialgebraic set.

A semialgebraic set can be represented as either a quantifier-free or first-order Boolean formula.

Definition 3.3. A quantifier-free Boolean formula is obtained by the following rules:

1. If $f \in \mathbb{R}[x_1, \dots, x_n]$ then $f = 0$ and $f > 0$ are quantifier-free Boolean formulas,
2. if F and G are quantifier-free Boolean formulas then $F \wedge G$, $F \vee G$ and $\neg F$ are quantifier-free Boolean formulas.

Definition 3.4. A first-order Boolean formula is obtained by rules 1 and 2 from Definition 3.1 and

3. If F is a first-order Boolean formula and y is a variable ranging over \mathbb{R} , then $\exists y F$ and $\forall y F$ are first-order Boolean formulas.

A fundamental result in semialgebraic geometry is the Tarski-Seidenberg Theorem.

Theorem 3.1. *Let $S \subset \mathbb{R}^{n+1}$ and $\text{proj}_{\mathbb{R}^n} : \mathbb{R}^{n+1} \rightarrow \mathbb{R}^n$ be the projection map onto the first n co-ordinates. Then $\text{proj}_{\mathbb{R}^n}(S) \subset \mathbb{R}^n$ is a semialgebraic set.*

One important consequence of this theorem is that quantifier elimination is possible over real numbers. In other words, a subset of \mathbb{R}^n defined by a first-order formula with quantifiers is semialgebraic and can be represented by a quantifier-free Boolean formula. One useful application of this theorem is that the closure, $\text{cl}((S))$, of any semialgebraic set $S \subset \mathbb{R}^n$ is also semialgebraic.

Proposition 3.1. *Every semialgebraic subset of \mathbb{R}^n is the (not necessarily disjoint) union of finitely many semialgebraic subsets of the kind*

$$\{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) = 0 \wedge g_1(\mathbf{x}) > 0 \wedge \cdots \wedge g_k(\mathbf{x}) > 0\}$$

where $k \in \mathbb{N}$ and $f, g_1, \dots, g_k \in \mathbb{R}[x_1, \dots, x_n]$.

This follows from the fact that the class of finite unions of this kind satisfy properties listed in Definition 3.1.

We now define the O-minimal structures over the reals.

Definition 3.5. A structure expanding the real closed field R is a collection

$$S := (S^n)_{n \in \mathbb{N}}$$

where each S^n is a set of subsets of the affine space R^n satisfying the following conditions.

1. All algebraic subsets of R^n are in S^n .
2. S^n is a Boolean subalgebra of the powerset of R^n .
3. If $A \in S^n$ and $B \in S^m$, then $A \times B \in S^{n+m}$.
4. Let $\text{proj} : R^{n+1} \rightarrow R^n$ be the projection onto the first n coordinates. If $X \in S^{n+1}$, then $\text{proj}(X) \in S^n$.

Definition 3.6. A structure S satisfying the properties of a definable set (listed in Definition 3.5) and

5. The elements of S^1 consist of the finite unions of points and open intervals of R .

Definition 3.7. A map $f : A \rightarrow \mathbb{R}^m$, where $A \subset \mathbb{R}^n$, is called *definable* if its graph $G \subset \mathbb{R}^{n+m}$ is a definable set.

Remark. Since definable sets are closed under projection, it can be easily deduced that A is a definable set.

Remark. For convenience, we will call a definable map $f : \mathbb{R}^n \rightarrow \mathbb{R}$ a definable function.

It is easy to see that semialgebraic sets satisfy the properties of sets definable in an O-minimal structures over the reals. Another example is the sub-pfaffian sets.

Definition 3.8. (?, Definition 2.1) A Pfaffian chain of order $r \geq 0$ and degree $\alpha \geq 1$ in an open domain $G \subset \mathbb{R}^n$ is a sequence of analytic functions

$$f_1, \dots, f_r$$

in G satisfying the differential equations

$$df_j(\mathbf{x}) = \sum_{i=1}^n g_{ij}(\mathbf{x}, f_1(\mathbf{x}), \dots, f_j(\mathbf{x})) dx_i$$

for $1 \leq j \leq r$, where $g_{ij}(x_1, \dots, x_n, y_1, \dots, y_j)$ are polynomials of degree not greater than α .

A function $f(\mathbf{x}) = P(\mathbf{x}, f_1(\mathbf{x}), \dots, f_r(\mathbf{x}))$, where $P(\mathbf{x}, f_1(\mathbf{x}), \dots, f_r(\mathbf{x}))$ is a polynomial of degree not greater than $\beta \geq 1$ is called a Pfaffian function of order r and degree (α, β) . Note that f is only defined in the domain G , where all functions f_1, \dots, f_r are analytic, even if f itself can be extended as an analytic function in a larger domain.

Examples of Pfaffian functions include the exponential, logarithmic, reciprocal and trigonometric functions, as well as the polynomials.

Definition 3.9. (?, Definition 2.7) A set $X \subset \mathbb{R}^n$ is called semi-Pfaffian in an open domain $G \subset \mathbb{R}^n$ if it consists of points in G satisfying a Boolean combination F of some atomic equations and inequalities $f = 0, g > 0$, where f, g are Pfaffian functions having a common Pfaffian chain defined in G .

X is called restricted in G if $\text{cl}(\cdot)X$ is contained in G .

X is called basic if the Boolean combination F is a conjunction.

Definition 3.10. (?, Definition 2.8) A set $X \subset \mathbb{R}^n$ is called sub-Pfaffian in an open domain $G \subset \mathbb{R}^n$ if

$$X := \text{proj}_{\mathbb{R}^n}(Y),$$

where $Y \subset \mathbb{R}^m$ is a semi-Pfaffian set and \mathbb{R}^n is a subspace of \mathbb{R}^m .

Definition 3.11. (?, 2.9) Let $I^k := [-1, 1]^k$ be the closed cube in an open domain $G \subset \mathbb{R}^k$. $X \subset I^n$ is called restricted sub-Pfaffian if $X := \text{proj}_{\mathbb{R}^n}(Y)$, where $Y \subset I^{n+m}$ is a restricted semi-Pfaffian set.

The restricted sub-pfaffian sets form a Boolean subalgebra. Restricted sub-pfaffian sets are clearly closed under taking finite unions and intersections. The property that the compliment of restricted sub-pfaffian set is restricted sub-pfaffian is a particular case of Gabrielov's compliment theorem (?). The restriction to the closed cube ensures that the restricted sub-Pfaffian sets are definable in an O-minimal structure. Furthermore, ? present an algorithm for constructing a cylindrical decomposition compatible with a restricted sub-Pfaffian set. Note that these properties and results may not be applied to (restricted) semi-Pfaffian sets. Since polynomials are a type of Pfaffian function, semialgebraic sets are a particular case of restricted sub-Pfaffian sets.

3.0.2 Monotone cells

It is clear from the definition that a set definable in an O-minimal structure is also a definable set and a semialgebraic set is definable in an O-minimal structure. In this section, we will introduce, closely following ?, some properties of definable sets which are also applicable to semialgebraic sets.

Definition 3.12. (?, Definition 2.1) Let $L_{j,c} := \{(x_1, \dots, x_n) \mid x_j = c\}$ for some $1 \leq j \leq n$ and $c \in \mathbb{R}$. Each intersection of the kind

$$S := L_{j_1, c_1} \cap \dots \cap L_{j_m, c_m}$$

where $0 \leq j \leq m$, $1 \leq j_1 < \dots < j_m \leq n$ and $c_1, \dots, c_m \in \mathbb{R}$ is called an affine coordinate subspace of \mathbb{R}^n .

Definition 3.13. (?, Definition 2.2) Let $\mathbf{f} = (f_1, \dots, f_k) : X \rightarrow \mathbb{R}^k$ be a bounded continuous map defined on an open, bounded, non-empty subset $X \subset \mathbb{R}^n$ having the graph $Y \subset \mathbb{R}^{n+k}$.

\mathbf{f} is called quasi-affine if, for any coordinate subspace $L \subset \mathbb{R}^{n+k}$, the restriction $\text{proj}_L|_Y$ is injective if and only if $\text{proj}_L(Y)$ is n -dimensional.

Definition 3.14. (?, Definition 2.3) Let $\mathbf{f} : X \rightarrow \mathbb{R}^k$ be a bounded, continuous, quasi-affine map defined on an open, bounded, non-empty subset $X \subset \mathbb{R}^n$ having the graph $Y \subset \mathbb{R}^{n+k}$. \mathbf{f} is called monotone if, for each affine coordinate subspace $S \subset \mathbb{R}^{n+k}$, the intersection $Y \cap S$ is connected.

Definition 3.15. (?, Definition 2.10) Let $X \subset \text{span}\{x_1, \dots, x_n\}$ be a monotone cell and $\mathbf{f} : X \rightarrow \text{span}\{y_1, \dots, y_k\}$ be a continuous map having graph Y . If Y is a monotone cell, then \mathbf{f} is called monotone on X .

Definition 3.16. (?, Definition 2.5) $Y \subset \text{span}\{x_1, \dots, x_n\}$ is called a monotone cell if it is the graph of a monotone map $\mathbf{f}: X \rightarrow H$ where $H \subset \text{span}\{x_1, \dots, x_n\}$ and $X \subset \text{span}\{x_1, \dots, x_n\} \setminus H$.

Example 3.1.

1. Observe that any quasi-affine map defined on a connected subset and having the graph in \mathbb{R}^2 is also monotone. This quasi-affine map will fail to be monotone if the set it is defined on is not connected.
2. Let $\Delta := \{0 < x < 1, y > 0, x + y < 1\} \subset \mathbb{R}^2$ and $\varphi(x, y) = x^2 + y^2$. Let $Y \subset \mathbb{R}^3$ be the graph of function ϕ on Δ . Observe that $\phi|_{\Delta}$ is a quasi-affine map. However, $y \cap \{z = 3/4\}$ is not connected, hence $\phi|_{\Delta}$ is not a monotone function.

** TODO: Notation 2.4 about $\text{span}\{x_1, \dots, x_n\}$ seems quite complicated. Why is the complicated explanation needed. **

** TODO 2 figure 1 shows examples of monotone (and not monotone) cells **

Proposition 3.2. (?, Theorem 1) *Every monotone cell is a topologically regular cell.*

The following properties of monotone cells are used in the proof of ?, Theorem 3.20.

Corollary 3.1. (?, Corollary 7, Theorem 11) *Let $X \subset \mathbb{R}^n$ be a monotone cell, then*

1. $X \cap \{x_i < c\}$, $X \cap \{x_i = c\}$ and $X \cap \{x_i > c\}$, for every $1 \leq i \leq n$ and $c \in \mathbb{R}$, are either empty or monotone cells.
2. Let $Y \subset X$ be a monotone cell such that $\dim(Y) = \dim(X) - 1$ and $\text{fr}((Y)) \subset \text{fr}((X))$. Then $X \setminus Y$ is a disjoint union of two monotone cells.

Proposition 3.3. (?, Theorem 10) *Let $X \subset \mathbb{R}^n$ be a monotone cell. Then $\text{proj}_L(X)$, for any affine coordinate subspace $L \subset \mathbb{R}^n$ is also a monotone cell.*

Remark. (?, Remark 2.11) Let $Y \subset \mathbb{R}^n$ be a monotone cell and L be a coordinate subspace such that $\text{proj}_L|_Y$ is injective. Then Y is the graph of a monotone map defined on $\text{proj}_L(Y)$, by (?, Theorem 7 and Corollary 5).

3.0.3 Cylindrical Algebraic Decompositions

A section cell C is the graph, in \mathbb{R}^n , of a continuous definable function $h: C' \rightarrow \mathbb{R}$, where C' is a cylindrical cell in \mathbb{R}^{n-1} .

In reality, we are given a set of *level- n projection factors* and C is a real root of a polynomial $f \in \mathbb{Z}[x_1, \dots, x_n]$, viewed as a polynomial in $\mathbb{Z}[x_1, \dots, x_{n-1}][x_n]$. By the construction of CAD, f has k real roots at every point $\mathbf{c} \in C'$.

In some situations, the former representation (with functions) is more convenient (than the latter, with roots).

Observe that it is easy to convert from the function representation to the root representation. Indeed, let $C \subset \mathbb{R}^n$ be the graph of

$$h : C' \rightarrow \mathbb{R}$$

where $C' \subset \mathbb{R}^{n-1}$ is an (i_1, \dots, i_{n-1}) -cell in \mathbb{R}^{n-1} . For every $\mathbf{c} \in C'$, the graph C can be written as a root of

$$f := h(\mathbf{c}) - x_n \in \mathbb{Z}[x_1, \dots, x_n]$$

The specific root of f such that $x_n = h(\mathbf{c})$ can be isolated, by Thom's lemma, by adding some sign conditions on partial derivatives of g .

Now let $f \in \mathbb{Z}[x_1, \dots, x_{n-1}][x_n]$ be a polynomial in x_n and $C' \subset \mathbb{R}^{n-1}$ be a (i_1, \dots, i_{n-1}) -cell of \mathbb{R}^{n-1} . Then the cylinder $C' \times \mathbb{R}$ ("above" C') contains a family of section cells

$$C := \{(\mathbf{c}, x_n) \mid \mathbf{c} \in C', f(\mathbf{c}, x_n) = 0\}.$$

The family C consists of k connected semialgebraic sets, corresponding to the k section cells on which $f = 0$. Suppose we want to represent one of these section cells $C \in C$ as the graph of $h : C' \rightarrow \mathbb{R}$. By Thom's lemma, we can write

$$C = \{(\mathbf{c}, x_n) \mid \mathbf{c} \in C', f(\mathbf{c}, x_n) = 0, g_1(\mathbf{c}, x_n) > 0, \dots, g_k(\mathbf{c}, x_n)\}$$

where g_1, \dots, g_k are partial derivatives of f .

C is a semialgebraic set, therefore h is certainly a definable function. But, is h a polynomial? The answer is no. This can be seen using a simple example.

Example 3.2. Let $D \subset \mathbb{R}^2$ be the unit disc, a semialgebraic set defined by a quantifier-free Boolean formula

$$\{x^2 + y^2 < 1\}.$$

Construct a CAD of \mathbb{R}^2 compatible with D in which D is a single sector cell. Observe that the top of D ,

$$D_T = \{-1 < x < 1, y > 0, x^2 + y^2 = 0\}$$

and

$$D' := \text{proj}_{\mathbb{R}^1}(D) = \{-1 < x < 1\}.$$

It is clear that D_T is a semialgebraic (and therefore a definable) set, so there exists a definable function

$$h : D' \rightarrow \mathbb{R}$$

such that D_T is the graph of $h|_{D'}$. Indeed, let

$$h(x) = +\sqrt{1-x^2}.$$

This is clearly not a polynomial.

In addition, it's clear that not every definable set is the graph of a definable map. Indeed, consider

$$C := \{x^2 + y^2 = 1\}$$

and

$$C' := \text{proj}_{\mathbb{R}^1}(C) = \{-1 \leq x \leq 1\}.$$

Both C and C' are clearly definable, but there is no map from $C' \rightarrow \mathbb{R}$ having C as its graph. Indeed, at $x = 0$, this map should have images -1 and 1 , which contradicts the definition of a map. However, in the case of a cylindrical section cell, the map h always exists by definition.

Therefore, when working with cylindrical section cells, the map $h : C' \rightarrow \mathbb{R}$ may be used, but algorithms will work solely with polynomials $f \in \mathbb{Z}[x_1, \dots, x_{n-1}][x_n]$ from the level- n projection factor set.

3.1 Cylindrical algebraic decomposition

? introduced both the concept of cylindrical algebraic decomposition along with an algorithm to construct one such that each cell has constant sign on a set of input polynomials $\mathbf{F} \subset \mathbb{R}[x_1, \dots, x_n]$.

Before describing Collins' algorithm, we will first define, closely following ?, a cylindrical cell and cylindrical decomposition. Note that ? work in the broader class of definable sets.

Definition 3.17. Cylindrical cells are defined by induction on $n \geq 1$. Each cylindrical cell C is a connected definable subset of \mathbb{R}^n with an *index*

$$(i_1, \dots, i_n) \in \{0, 1\}^n.$$

- When $n = 1$, a (0)-cell (section cell) is a point $c \in \mathbb{R}$ and a (1)-cell (sector cell) is an open interval: (a, b) , $(-\infty, b)$, (a, ∞) , $(-\infty, \infty)$ where $a, b \in \mathbb{R}$.
- When $n > 1$, suppose that (i_1, \dots, i_{n-1}) -cells are already defined and let C' be one of these cells. An $(i_1, \dots, i_{n-1}, 0)$ -cell (section cell) is the graph

of a continuous definable function $f : C' \rightarrow \mathbb{R}$. An $(i_1, \dots, i_{n-1}, 1)$ -cell (sector cell) is a subset of the cylinder $C' \times \mathbb{R}$, either

$$\begin{aligned} & \{(\mathbf{x}, t) \mid \mathbf{x} \in C', f(\mathbf{x}) < t < g(\mathbf{x})\}, \\ & \{(\mathbf{x}, t) \mid \mathbf{x} \in C', -\infty < t < g(\mathbf{x})\}, \\ & \{(\mathbf{x}, t) \mid \mathbf{x} \in C', f(\mathbf{x}) < t < \infty\}, \\ & \{(\mathbf{x}, t) \mid \mathbf{x} \in C', t \in \mathbb{R}\} \end{aligned}$$

where $f, g : C' \rightarrow \mathbb{R}$ are continuous definable functions such that $f(\mathbf{x}) < g(\mathbf{x})$ for all $\mathbf{x} \in C'$.

Let $S \subset \mathbb{R}^n$ be a definable subset of \mathbb{R}^n and define the *frontier* of S

$$\text{fr}(S) := \text{cl}(S) \setminus S,$$

where $\text{cl}(S)$ is the closure of S in the Euclidean topology.

If $C \subset \mathbb{R}^n$ is a cylindrical cell, then it will be convenient to partition $\text{fr}(C)$ into three subsets: the top (C_T), bottom (C_B) and side-wall (C_W).

First let C be a sector $(i_1, \dots, i_{n-1}, 1)$ -cell and let $C' := \text{proj}_{\mathbb{R}^{n-1}}(C)$. If C is bounded from below by a continuous function $f : C' \rightarrow \mathbb{R}$, then C_B is the graph of f . Similarly, if C is bounded from above by the graph of a continuous function $g : C' \rightarrow \mathbb{R}$, then C_T is the graph of g . It is clear that the top and bottom of every cylindrical sector cell (if non-empty) are always cylindrical section cells.

Now let C be an $(i_1, \dots, i_k, 0, \dots, 0)$ -cell, where $i_k = 1$, and $C' := \text{proj}_{\mathbb{R}^k}(C)$. Since C' is a sector cell, C'_B and C'_T are cylindrical section cells. The bottom C_B of C is the pre-image of C'_B by the projection map $\text{proj}_{\mathbb{R}^k}|_{\text{cl}(C)}$ and the top C_T of C is the pre-image of C'_T by the projection map $\text{proj}_{\mathbb{R}^k}|_{\text{cl}(C)}$. It may be simpler to think of C_B as the definable set $\text{cl}(C) \cap (C'_B \times \mathbb{R}^{n-k+1})$ and C_T as the definable set $\text{cl}(C) \cap (C'_T \times \mathbb{R}^{n-k+1})$. It is important to note that C_B and C_T may not be cylindrical cells. Indeed, they may fail to be graphs of continuous functions (see Example 3.3).

For both section and sector cells, let the side wall, C_W , be $\text{fr}(C) \setminus (C_B \cup C_T)$. There is no requirement that C_W will be (a union of) cylindrical cells.

Example 3.3. In \mathbb{R}^3 , let $C' := \{(x, y) \mid -1 < x < 1, |x| < y < 1\}$ and $\phi(x, y) = |x/y|$ so that C is the graph of $\phi|_{C'}$. Consider

$$C_B := \{(x, y, z) \mid -1 < x < 1, y = |x|, z = 1\} \cup \{(0, 0, z) \mid 0 \leq z \leq 1\}.$$

C_B is not the graph of a continuous function due to the blow-up point of ϕ at the origin.

We now define the cylindrical decomposition.

Definition 3.18. A cylindrical decomposition is defined by induction on $n \geq 0$.

- When $n = 0$, the unique cylindrical decomposition of \mathbb{R}^0 is the unique point in \mathbb{R}^0 .
- When $n > 0$: Let D be a partition of \mathbb{R}^n into cylindrical cells. Define D' to be the set of all projections $C' := \text{proj}_{\mathbb{R}^{n-1}}(C)$ for all C in D . D is a cylindrical decomposition of \mathbb{R}^n if D' is a cylindrical decomposition of \mathbb{R}^{n-1} . We call D' the *decomposition of \mathbb{R}^{n-1} induced by D .

Less formally, a partition D of \mathbb{R}^n into definable subsets is cylindrical if the projection of any two cells of D onto \mathbb{R}^{n-1} is either disjoint or coincides.

Definition 3.19. Let $S \subset \mathbb{R}^n$ be a definable set. A cylindrical decomposition D is compatible with S if every cell C of D is either a subset of S or disjoint from S .

Definition 3.20. A cylindrical decomposition E is called a refinement of D if E is compatible with every cell of D . In other words, every cell of D is a union of cells in the refinement E .

Remark. (? , Remark 3.8) Let D be a cylindrical decomposition of \mathbb{R}^n and C be a cylindrical cell of D such that $c := \text{proj}_{\mathbb{R}^1}(C)$ is a single point. Then it follows immediately from the definition that D is compatible with the hyperplane $\{x_1 = c\}$ and the set of all cells of D contained in this hyperplane form a cylindrical decomposition E of \mathbb{R}^{n-1} . Moreover, any refinement of E is also a refinement of D .

We now present some useful results proved by ? relating to desirable properties of cylindrical decompositions.

Definition 3.21. Let $S \subset \mathbb{R}^n$ be a definable set. A cylindrical decomposition, compatible with S , D is called monotone with respect to S if every cell $C \subset S$ of D is monotone.

Definition 3.22. Let D be a cylindrical decomposition of \mathbb{R}^n and C be a cell of D . C satisfies the frontier condition in D if $\text{fr}((\cdot)C)$ is a union of cells of D of smaller dimension. If it is unambiguous that C is a cell of D , then we just say that C satisfies the frontier condition. If every cell in D satisfies the frontier condition, then the cylindrical decomposition D satisfies the frontier condition.

We saw in Example ??exm:top-bottom-not-cylindrcilac) that the top and bottom of a 2-dimensional cylindrical section cell, even in \mathbb{R}^3 , need not be a union of cylindrical cells. However, it is clear from the cylindricity property that, if C is a 2-dimensional section cell satisfying the frontier condition in a decomposition D , then the top and bottom of C are single cylindrical cells. If C is a

2-dimensional section cell, then it is said to satisfy the strong frontier condition if the one-dimensional components of its side-wall are 1-dimensional section cells. This condition may be impossible, even in \mathbb{R}^3 , as Example 3.4 demonstrates.

Example 3.4. Consider the two cylindrical cells

$$V := \{(x, y, z) \mid x > y > 0, z > 0, y = xz\}$$

and

$$W := \{(x, y, z) \mid x > y > 0, z > 0, y = 2xz\}$$

in \mathbb{R}^3 . Any cylindrical decomposition compatible with V and W is also compatible with the two intervals

$$I_1 = \{(0, 0, z) \mid 0 \leq z \leq 1/2\}, I_2 = \{(0, 0, z) \mid 1/2 \leq z \leq 1\}$$

and the point

$$v = (0, 0, 1/2).$$

Observe that $I = \{(0, 0, z) \mid 0 \leq z \leq 1\}$ is the only one-dimensional component of the side-wall of V and I_1 is the only one-dimensional component of the side-wall of W . In order to obtain the strong frontier condition, we need to partition V into (at least) three cylindrical cells V', V'', V''' , one of which, say V'' , will be one-dimensional. V'' should have endpoint v (i.e., $\text{cl}((V'') \cap I) = v$). The tangent at the origin of $c := \text{proj}_{\mathbb{R}^2}(V'')$ is $1/2$. The pre-image of the projection map $W' := \text{proj}_{\mathbb{R}^2}^{-1}|_{\text{cl}(W)}(c)$ would satisfy the condition $\text{cl}(W') \cap I = v'$, where $v' = (0, 0, 1/4)$. However, v' must be a zero-dimensional cell of the decomposition, which means we need to perform another refinement of I . An infinite sequence of refinements, introducing a new 0-dimensional cell $(0, 0, 1/2^k)$, $k \in \mathbb{N}_{>0}$, will take place. Thus, a decomposition in which the side-walls of V and W are single 1-dimensional cells cannot exist.

Proposition 3.4. (*?, Lemma 3.3*) Let C be an $(i_1, \dots, i_{k-1}, 0, i_{k+1}, \dots, i_n)$ -cell. Then $C' := \text{proj}_{\text{span}\{x_1, \dots, x_{k-1}, x_{k+1}, \dots, x_n\}}(C)$ is a cylindrical $(i_1, \dots, i_{k-1}, i_{k+1}, \dots, i_n)$ -cell and C is the graph of a monotone map $f : C' \rightarrow \text{span}\{x_k\}$.

Proposition 3.5. (*?, Lemma 3.4*) Let $C \subset \mathbb{R}^n$ be a 2-dimensional cylindrical cell which is the graph of a quasi-affine map. Then the side-wall $W \subset \text{fr}(C)$ has exactly two connected components, each of which is either a point or a closed curve interval.

3.1.1 Constructing a CAD

Remark. A cylindrical algebraic decomposition (CAD) is a cylindrical decomposition in which every cell is a semialgebraic set. While the results of ? are proved for the wider class of cylindrical decompositions compatible with definable sets, we will restrict ourselves to CADs, allowing us to take advantage of properties of polynomials.

Definition 3.23. Let $\mathbf{F} := (f_1, \dots, f_s) \subset \mathbb{Z}[x_1, \dots, x_n]$ be a set of polynomials with integer coefficients. A cylindrical algebraic decomposition D is called sign-invariant with respect to \mathbf{F} (or simply \mathbf{F} -invariant) if every cell C of D has constant sign (either > 0 , < 0 or $= 0$) on each polynomial in \mathbf{F} . Here, \mathbf{F} is called the set of input polynomials.

Remark. The fundamental theorem of algebra implies that for the coefficients of \mathbf{F} , we may choose any ring $K \subset \mathbb{A}$. \mathbb{Z} has been chosen for convenience, but some authors (e.g., ?) choose \mathbb{Q} instead.

Let $\mathbf{F} \subset \mathbb{Z}[x_1, \dots, x_n]$ be a set of polynomials with integer coefficients. We will now describe the algorithm proposed by ?, closely following ?, for constructing an \mathbf{F} -invariant CAD of \mathbb{R}^n .

Collins' algorithm uses the idea of projections and lifting. More precisely, successive projections of the set of input polynomials to smaller and smaller dimensions are taken, until a set of univariate polynomials is obtained. The CAD of \mathbb{R}^1 is constructed by isolating the roots of these univariate polynomials, with section cells being the roots and sector cells being the open intervals in between them. The lifting phase then works recursively: given a collection of cells in \mathbb{R}^k and a set of polynomials in \mathbb{R}^{k+1} , the roots of these polynomials within each cell C are isolated. The section cells in the cylinder $C \times \mathbb{R}$ are the roots of these polynomials, while the sector cells are the bands in between the roots. The projection operation was designed such that the cylindrical property holds. I.e., above C , each polynomial has a constant number of roots and the roots of no two polynomials crosses one another. This second property is called delineability.

Let us begin the description of the algorithm by considering a single polynomial $f \in \mathbb{Z}[x_1, \dots, x_n]$. We want to partition \mathbb{R}^{n-1} into connected semialgebraic sets C' such that, for all $\mathbf{x} \in C'$, $f(\mathbf{x}, x_n)$ has constant degree and a constant number of roots.

Proposition 3.6. (*?, Proposition 2.16*) Let $f \in \mathbb{Z}[x_1, \dots, x_n]$ and, $C' \subset \mathbb{R}^{n-1}$ be a connected semialgebraic set and $k \leq d \in \mathbb{N}$ such that, for every point $\mathbf{x} \in C'$, the univariate polynomial $f(\mathbf{x}, x_n) \in \mathbb{A}[x_n]$ has degree d and exactly k distinct (complex) roots in C' . Then there are $\ell \leq k$ distinct semialgebraic functions

$$\psi_1, \dots, \psi_\ell : C' \rightarrow \mathbb{R}$$

such that, for every $\mathbf{x} \in C'$, the set of real roots of $f(\mathbf{x}, x_n)$ is exactly

$$\{\psi_1(\mathbf{x}), \dots, \psi_\ell(\mathbf{x})\}.$$

Moreover, the multiplicity of each of these roots is constant.

Proof. The argument relies on the “continuity of roots”:

Fix $\mathbf{c} \in C'$ and let z_1, \dots, z_k be the distinct roots of $f(\mathbf{c}, x_n)$ with multiplicities m_1, \dots, m_k . Choose $\varepsilon > 0$ small enough that the open discs $D(z_i, \varepsilon)$ ($z_i \in \mathbb{C}$ (center z_i , radius ε)) are disjoint. If $\mathbf{b} \in C'$ is sufficiently close to \mathbf{c} , then the polynomial $f(\mathbf{b}, x_n)$ has exactly m_i roots, counted with multiplicities, in $D(z_i, \varepsilon)$ for $1 \leq i \leq k$.

Since $f(\mathbf{b}, x_n)$ has k distinct complex roots, and $d = m_1 + \dots + m_k$ complex roots counted with multiplicities, it follows that each $D(z_i, \varepsilon)$ contains exactly one root, denoted ζ_i , of $f(\mathbf{b}, x_n)$ with multiplicity m_i . If z_i is real, then ζ_i is real, otherwise the complex conjugate of ζ_i would be another root of $f(\mathbf{b}, x_n)$ in the disc $D(z_i, \varepsilon)$. If z_i is complex then ζ_i is also complex, since the conjugation of every point in $D(z_i, \varepsilon)$ lies outside $D(z_i, \varepsilon)$, forming another disc $D(\bar{z}_i, \varepsilon)$. It follows that, if $\mathbf{b} \in C'$ is close enough to \mathbf{c} , then $f(\mathbf{c}, x_n)$ has the same number of real roots as $f(\mathbf{b}, x_n)$. Since C' is connected, $f(\mathbf{x}, x_n)$ has the same number of real roots at every point $\mathbf{x} \in C'$, say ℓ . Define $\psi_i(\mathbf{x}) : C' \rightarrow \mathbb{R}$ to be the continuous (by making ε small enough) semialgebraic function sending $\mathbf{x} \in C'$ to the i -th (in ascending order) real root of $f(\mathbf{x}, x_n)$. It follows from the connectedness of C' that each ψ_i has constant multiplicity. Observe that the graph of each ψ_i can be expressed by a first-order Boolean formula, using existential quantifiers to express each of the ℓ roots of $f(\mathbf{x}, x_n)$ and an equality condition to pick out the i -th root. It follows that the graph of each function ψ_i is a semialgebraic set. \square

To extend this result to more than one polynomial, we also need to ensure that the graphs corresponding to each root never intersect.

Proposition 3.7. (*?, Proposition 2.18*) *Let $f, g \in \mathbb{Z}[x_1, \dots, x_n]$, $C' \subset \mathbb{R}^{n-1}$ such that, for all $\mathbf{x} \in C'$, the degree and number of roots of $f(\mathbf{x}, x_n)$ and $g(\mathbf{x}, x_n)$ is constant and the degree of the GCD of $f(\mathbf{x}, x_n)$ and $g(\mathbf{x}, x_n)$ is constant. Let $\phi, \psi : C' \rightarrow \mathbb{R}$ be continuous semialgebraic functions such that $f(\mathbf{x}, \phi(\mathbf{x})) = 0$ and $g(\mathbf{x}, \psi(\mathbf{x})) = 0$ for all $\mathbf{x} \in C'$. If there exists $\mathbf{c} \in C'$ such that $\phi(\mathbf{c}) = \psi(\mathbf{c})$, then $\phi(\mathbf{x}) = \psi(\mathbf{x})$ for all $\mathbf{x} \in C'$.*

Proof. We use the same method of proof as in the previous proposition. For an arbitrary element $\mathbf{c} \in C'$, let $z_1 = \phi(\mathbf{c}) = \psi(\mathbf{c}), \dots, z_k$ be the distinct roots in \mathbb{C} of the product

$$f(\mathbf{c}, x_n)g(\mathbf{c}, x_n)$$

(recall that $fg = 0$ if either $f = 0$ or $g = 0$). Let m_i (resp p_i) be the multiplicity of z_i as a root of $f(\mathbf{x}, x_n)$ (resp. $g(\mathbf{x}, x_n)$) where multiplicity zero indicates that z_i is not a root. The degree of $\gcd(f(\mathbf{x}, x_n), g(\mathbf{x}, x_n))$ is $\min(m_1, p_1) + \dots + \min(m_k, p_k)$ and each z_i has multiplicity $\min(m_i, p_i)$ as a root of this GCD. Choose $\varepsilon > 0$ small enough that the discs $D(z_i, \varepsilon)$ are disjoint. For each $\mathbf{b} \in C'$ close enough to \mathbf{c} , each disc contains a root of multiplicity m_i of $f(\mathbf{b}, x_n)$ and a root of multiplicity p_i of $g(\mathbf{b}, x_n)$. Since the degree of the GCD of f and g (evaluated at any $\mathbf{x} \in C'$) is equal to $\min(m_1, p_1) + \dots + \min(m_k, p_k)$, this GCD must have one root of multiplicity $\min(m_i, p_i)$ in each disc $D(z_i, \varepsilon)$ such

that $\min(m_i, p_i) > 0$. In particular, it follows that $\phi(\mathbf{b}) = \psi(\mathbf{b})$. Since C' is connected, this equality holds for all $\mathbf{x} \in C'$. \square

Now we have algebraic conditions on the polynomials in the set of level- $(n-1)$ projection polynomials $\mathbf{F}' = \text{proj}(\mathbf{F})$. We now describe how these properties can be satisfied and, as such, how the set \mathbf{F}' can be constructed. We will need the concept of the principal subresultant coefficient of two polynomials f and g .

Definition 3.24. Let $f, g \in \mathbb{Z}[x_1, \dots, x_{n-1}][x_n]$ be polynomials, with degrees d and e respectively, such that

$$\begin{aligned} f(x_n) &= a_d x_n^d + \dots + a_1 x_n + a_0 \\ g(x_n) &= b_e x_n^e + \dots + b_1 x_n + b_0. \end{aligned}$$

Then the Sylvester matrix associated to f and g is the $(d+e) \times (d+e)$ -matrix defined as follows.

$$S_{f,g}(f, g) = \begin{pmatrix} a_d & \dots & a_0 & 0 & \dots & \dots & 0 \\ 0 & a_d & \dots & a_0 & 0 & \dots & 0 \\ \dots & 0 & a_d & \dots & a_0 & \dots & 0 \\ b_e & \dots & b_0 & 0 & \dots & \dots & 0 \\ 0 & b_e & \dots & b_0 & 0 & \dots & 0 \\ \dots & 0 & b_e & \dots & b_0 & \dots & 0 \end{pmatrix}.$$

Definition 3.25. Let $f, g \in \mathbb{Z}[x_1, \dots, x_{n-1}][x_n]$ be polynomials with degrees d and e respectively. The principal subresultant coefficient of order j of f and g , denoted $\text{psrc}_j(f, g)$, is the determinant of the $(m+n-2j) \times (m+n-2j)$ -matrix obtained from the Sylvester matrix associated to f and g by deleting the first and last j rows and columns.

Let $f, g \in \mathbb{Z}[x_1, \dots, x_{n-1}][x_n]$ and $X \subset \mathbb{Z}[x_1, \dots, x_n]$.

- f has a constant number of complex roots over X if

$$\text{psrc}_k(f, \partial f / \partial x_n),$$

for $1 \leq k \leq \deg(f)$, is either zero or nonzero everywhere in X .

- f and g have constant GCD in X if

$$\text{psrc}_k(f, g),$$

for $1 \leq k \leq \min(\deg(f), \deg(g))$, is either zero or nonzero and $\deg(f) = \deg(g)$ everywhere in X .

- If, for the leading term of f (or g) vanishes at some points in $\mathbb{Z}[x_1, \dots, x_{n-1}]$, we need to take the principal subresultant coefficients of the reductum of f (or g). (? , pp36)

From these properties of the principal subresultant coefficient, we can define a projection operator.

Definition 3.26. Consider $f \in \mathbb{Z}[x_1, \dots, x_n]$ as a univariate polynomial in x_n . I.e.,

$$f(x_n) = a_d x_n^d + \dots + a_1 x_n + a_0$$

where $a_d, \dots, a_0 \in \mathbb{Z}[x_1, \dots, x_{n-1}]$. Let $\text{lc } f = a_d$ denote the leading coefficient of f and $\text{red}(f) = a_{d-1} x_n^{d-1} + \dots + a_1 x_n + a_0$ be the reductum of f .

Let $\mathbf{F} \subset \mathbb{Z}[x_1, \dots, x_n]$. We define the projection operator, $\text{proj}(\mathbf{F})$ as follows:

- Let $d := \deg(f_i)$. If $d > 1$, then $\text{proj}(f_1, \dots, f_i, \dots, f_s)$ contains

$$\text{psrc}_k(f_i, \partial f_i / \partial x_n)$$

for all $1 \leq k \leq d$.

- Let $d := \min(\deg(f_i), \deg(f_j))$. If $d > 0$, then $\text{proj}(f_1, \dots, f_i, \dots, f_j, \dots, f_s)$ contains

$$\text{psrc}_k(f_i, f_j)$$

for all $1 \leq k \leq d$.

- If $\deg(f_i) > 0$ and $\text{lc}(f_i)$ is non-constant, then $\text{proj}(f_1, \dots, f_i, \dots, f_s)$ contains

$$\text{lc}(f_i) \text{ and } \text{proj}(f_1, \dots, \text{red}(f_i), \dots, f_n).$$

Theorem 3.2. (*?, Theorem 2.19*) Let $\mathbf{F} \subset \mathbb{Z}[x_1, \dots, x_n]$ be a family of polynomials and let C' be a connected $\text{proj}(\mathbf{F})$ -invariant semialgebraic subset of \mathbb{R}^{n-1} – an (i_1, \dots, i_{n-1}) -cell. Then there exist continuous definable functions

$$\psi_1, \dots, \psi_\ell : C' \rightarrow \mathbb{R}$$

such that for all $\mathbf{x} \in C'$, the set $\{\psi_i(\mathbf{x}), \dots, \psi_k(\mathbf{x})\}$ coincides with the real roots of polynomials in \mathbf{F} , defining the $(i_1, \dots, i_{n-1}, 0)$ -cells in the cylinder $C' \times \mathbb{R}$ and $\{-\infty < t < \psi_1(\mathbf{x}), \dots, \psi_i(\mathbf{x}) < t < \psi_{i+1}(\mathbf{x}), \dots, \psi_k(\mathbf{x}) < t < \infty\}$ are the $(i_1, \dots, i_{n-1}, 1)$ -cells in the cylinder $C' \times \mathbb{R}$.

This follows from the results previously proved. Applying Theorem @??thm:cad-lift), we are able to pass from a $\text{proj}_{\mathbb{R}^{k-1}}(\mathbf{F})$ -invariant CAD of \mathbb{R}^{k-1} to a $\text{proj}_{\mathbb{R}^k}(\mathbf{F})$ -invariant CAD of \mathbb{R}^k . Thus, by iterating the projection operation until we obtain a set of polynomials in $\mathbb{Z}[x_1]$, isolating their roots, and then iteratively applying Theorem 3.2\$, we are able to obtain an \mathbf{F} -invariant CAD of \mathbb{R}^n .

3.1.2 Bounds and variations

We have described how to construct a CAD of \mathbb{R}^n which is sign-invariant with respect to a set of polynomials. We now present the bounds, along with some variations, based ?.

Proposition 3.8. (*?, Algorithm 11.2*) *Let $\mathbf{F} := f_1, \dots, f_s$ be set of polynomials in $\mathbb{Z}[x_1, \dots, x_n]$ with maximum degree d . There is an algorithm, taking \mathbf{F} as input, which produces an \mathbf{F} -invariant CAD D of \mathbb{R}^n . The complexity of the algorithm is*

$$(sd)^{O(1)^n}.$$

This is also an upper bound on the number of cells in D , number of polynomials defining cells and their degrees.

Observe that these bounds are doubly exponential in the number of variables. Indeed, ? give an example of a CAD in which a doubly exponential number of cells is obtained. In practice, we can obtain a CAD with fewer cells, e.g., by constructing a truth invariant or partial CAD.

Proposition 3.9. *Let S_1, \dots, S_k be a finite collection of semialgebraic subsets of \mathbb{R}^n defined by quantifier-free Boolean formulas F_1, \dots, F_k respectively. Together, these formulas include s different polynomials in $\mathbb{Z}[x_1, \dots, x_n]$ with maximum degree d . There is an algorithm, taking $\{F_1, \dots, F_k\}$ as input, which produces a cylindrical decomposition \mathcal{E} of \mathbb{R}^n compatible with each set $S_i, 1 \leq i \leq k$. Complexity, number of cells, number of polynomials and degrees are the same as in Proposition ??prp:collins) but, in practice.*

Proof. Let $f_1, \dots, f_s \in \mathbb{R}[x_1, \dots, x_n]$ be the polynomials from formulas F_1, \dots, F_k and let \mathcal{E}' be the CAD of \mathbb{R}^n produced by Proposition @ref{prp:novel-frontier-collins} with f_1, \dots, f_s as input. Since F_i for each $1 \leq i \leq k$ provides a set of sign conditions on polynomials f_1, \dots, f_s , \mathcal{E}' is a refinement of the CAD produced by this algorithm.

Truth values of F_1, \dots, F_k on each cell can easily be computed from the CAD produced by Proposition 8.1 by considering the signs of f_1, \dots, f_s on each cell, which we computed in the construction of \mathcal{E}' . Observe that this does not change the asymptotic complexity or bounds in Proposition ??. \square

It is also common to construct a CAD compatible with a single semialgebraic set.

Corollary 3.2. *Let $S \subset \mathbb{R}^n$ be a semialgebraic set defined by a quantifier-free Boolean formula F containing s different polynomials in $\mathbb{R}[x_1, \dots, x_n]$, having maximum degree d . There is an algorithm, taking F as input, which produces a cylindrical decomposition \mathcal{E} of \mathbb{R}^n compatible with S . Complexity, number of cells, number of polynomials and degrees are the same as in Proposition 3.8.*

Proof. Immediate, by applying Proposition 3.2 to $\{F\}$. \square

The construction described in Proposition 3.9 is somewhat naive. Constructions which are far more efficient in practice are presented in e.g., ?. An alternative approach is presented by ?. While very useful in practice, these constructions are unable to lower the upper bound from Proposition 3.8 due to some extreme cases, e.g., when the input formulas define a sign-invariant CAD (see ?, Section 6.2.1].

3.1.3 The projection operator

Since Collins published his CAD algorithm in 1975, many variations of the projection operator have been proposed. Most of these aim to minimise the number of polynomials appearing in $\text{proj}_{k-1}(\mathbf{F})$, so as to make the algorithm more efficient in practice. For example, ? observed that, in dimension 2, if the set $\mathbf{F} \subset \mathbb{Z}[x_1, x_2]$ of input polynomials is squarefree and pairwise relatively prime, then it is sufficient to include only the leading coefficients, resultants and discriminants of pairs of polynomials in $\text{proj}(\mathbf{F})$. ? then proved that a similar construction is possible in dimension 3. ? later improved upon his previous work by extending this result to dimension $n > 3$, as long as the set of input polynomials is well oriented.

:: {definition} (?, 6.1) Let $\mathbf{F} \subset \mathbb{Z}[x_1, \dots, x_n]$ be a set of polynomials and denote by $\text{prim}(\mathbf{F})$ the set of all primitive parts of \mathbf{F} . \mathbf{F} is called well-oriented if, when $n > 1$,

- for each element $f \in \text{prim}(\mathbf{F})$, $f(\mathbf{x}, y) = 0$ for all $y \in \mathbb{R}$ on a finite number of points $\mathbf{x} \in \mathbb{R}^{n-1}$. (?, condition WO1)
- $\text{proj}(\mathbf{F})$ is well-oriented. (?, condition WO2) :::

? then observed that even more polynomials can be discarded, creating the Reduced McCallum Projection Operator. This is commonly used in practice, e.g., in QEPCAD-B (?).

Let $\mathbf{F} \subset \mathbb{Z}[x_1, \dots, x_n]$ be a well-oriented set of polynomials and D be an \mathbf{F} -invariant CAD constructed using the McCallum projection operator (?). Then, by ? Theorems 2.2.3 and 2.2.4, every cell C of D is an analytic submanifold of \mathbb{R}^n . Informally, if $\dim(C) = k$, then C is a non-empty subset of \mathbb{R}^n which “looks locally like \mathbb{R}^k ” (see Definition 3.27 for the precise definition).

Definition 3.27. $S \subset \mathbb{R}^n$ such that $\dim(S) = k$ is an analytic submanifold if, for every point $\mathbf{x} \in S$, there is an analytic coordinate system about \mathbf{x} with respect to which S is locally the intersection of $n - k$ coordinate hyperplanes.

Proposition 3.10. Let $S \subset \mathbb{R}^n$ be a semialgebraic set defined by a quantifier-free Boolean formula constructed from the set of polynomials $\mathbf{F} \subset \mathbb{Z}[x_1, \dots, x_n]$, such that \mathbf{F} is well-oriented. Then a CAD D of \mathbb{R}^n compatible with S and

such that every cell C of D is an analytic submanifold of \mathbb{R}^n can be obtained by constructing an \mathbf{F} -invariant CAD of \mathbb{R}^n using the McCallum projection operator.

More recently, ? completed the proof that Lazard's projection operator is valid. He also proved, without the condition on well-orientedness of input polynomials, that every cell of a sign-invariant CAD constructed using Lazard's projection operator is an analytic submanifold. Thus, we can generalise Proposition 3.10 further:

Proposition 3.11. *Let $S \subset \mathbb{R}^n$ be a semialgebraic set defined by a quantifier-free Boolean formula constructed from the set of polynomials $\mathbf{F} \subset \mathbb{Z}[x_1, \dots, x_n]$. Then a CAD D of \mathbb{R}^n compatible with S and such that every cell C of D is an analytic submanifold of \mathbb{R}^n can be obtained by constructing an \mathbf{F} -invariant CAD of \mathbb{R}^n using the McCallum projection operator.*

3.1.4 Projection polynomials vs semialgebraic functions

Recall that Proposition 3.6 allows us to pass from a projection polynomial $f \in \mathbb{Z}[x_1, \dots, x_k]$ to the continuous semialgebraic functions $\phi : C' \rightarrow \mathbb{R}$, where $C' \subset \mathbb{R}^{n-1}$ is a cylindrical cell, from Definition 3.17 which define some of the section cells in the cylinder $C' \times \mathbb{R}$. The construction described in ?, Theorem 3.20 (and related lemmas) works with these continuous semialgebraic functions. However, we are given only the projection polynomials f of which ϕ defines one of the roots.

Let C be a section cell in \mathbb{R}^n : a semialgebraic set which is the graph of a continuous semialgebraic function $\phi : C' \rightarrow \mathbb{R}$, where $C' \subset \mathbb{R}^{n-1}$ is a cylindrical cell. A simple example in \mathbb{R}^2 demonstrates that, although C is a semialgebraic set, ϕ may not be a polynomial. Indeed, let

$$C := \{(x, y) \in \mathbb{R}^2 \mid x > 0, y > 0, y = x^2\}$$

be a cylindrical cell in \mathbb{R}^2 such that $C' := \text{proj}_{\mathbb{R}^1}(C) = \{x \in \mathbb{R} \mid x > 0\}$. We can write

$$\phi(x) = +\sqrt{x}$$

such that C is the graph of $\phi|_{C'}$. However, it's clear that ϕ is not a polynomial.

In computations, we work exclusively with section cells. When dealing with sector cells, we will consider the graphs of functions defining their top and bottom. Since we will be using SACLIB and sometimes Singular, which are polynomial libraries, we will not be able to use the functions ϕ in computations. We will instead use a different representation for a section cell C in \mathbb{R}^n . Let C be the root of a polynomial $f \in \mathbb{Z}[x_1, \dots, x_n]$ and $C' := \text{proj}_{\mathbb{R}^{n-1}}(C)$ satisfy the conditions in Proposition ??prp:coste-polynomial-to-cell). Then

$$C \subset Z := \{(\mathbf{x}, x_n) \in \mathbb{R}^n \mid \mathbf{x} \in C', f(\mathbf{x}, x_n) = 0\} \subset C' \times \mathbb{R}.$$

In particular, Z consists of the roots of f in the cylinder $C' \times \mathbb{R}$, including the section cell C . It is clear from the Tarski-Seidenberg theorem that it is possible to represent C as a first-order Boolean formula. In practice, since C is the root of a polynomial, we may define it using Thom's Lemma (?). I.e., a root of a polynomial may be defined by sign conditions on its derivatives. Thus, let

$$C = \{(\mathbf{x}, x_n) \in \mathbb{R}^n \mid \mathbf{x} \in C', f(\mathbf{x}, x_n) = 0, g_1(\mathbf{x}, x_n) > 0, \dots, g_k(\mathbf{x}, x_n) > 0\}$$

where g_1, \dots, g_k are derivatives of the polynomial f .

3.1.5 CAD cells and their defining formulas

Although it is possible to construct a “full” (sign-invariant) CAD, QEPCAD-B provides an implementation of the “truth-invariant” CAD described in Proposition 3.9. Indeed, QEPCAD-B, being designed primarily to perform quantifier elimination, frequently constructs a “partial” CAD of \mathbb{R}^n (?). We will force the construction of the sign-invariant CAD and use McCallum's projection operator so as to obtain smooth cells (see Proposition 3.10).

Once constructed, one may want to examine individual cells of the CAD. In order to do this, we will need a convenient representation for each cell. Depending on the purpose for which the CAD has been constructed, a particular representation may be chosen. For example, if we only need a witness, i.e., some point in the cell, only the sample points will suffice. Indeed, this is a common requirement and Maple's 'RegularChains::CylindricalAlgebraicDecompose' function provides output in this representation (?). On the other hand, one might want to view each cell as a semialgebraic set. In this case, a defining formula will be necessary. Such a representation using only the input and projection polynomials may be impossible and, as such, an “extended language” has been introduced in both Maple and QEPCAD (?, ?). The extended language includes the rules of first-order Boolean formulas (see Definition 3.3) along with a means of representing “the i -th root of polynomial f ”. We find this representation a little inconvenient (TODO why?) and would prefer each cell to be representable by an ordinary first-order Boolean formula.

To see how the signs of projection factors may be insufficient to represent a cylindrical cell, consider the following simple example in \mathbb{R}^2 . Let $f = x - y^2$ and construct an $\{f\}$ -invariant CAD. The projection polynomials

$$P = \{x, x - y^2\}.$$

** TODO: draw a picture, label cells by C_1 and C_2. ** The two distinct cells C_1 and C_2 having sign $x > 0, x - y^2 = 0$. By Thom's Lemma, it is possible to represent these cells by sign conditions on derivatives of f . In this example, it is clear that another polynomial, y , is needed to distinguish between these cells. ? proposed an algorithm for constructing a solution formula where the projection polynomials are not sufficient. The basic idea is that “conflicting pairs”, distinct cells having the same signature (signs of projection factors), are

identified, then additional derivatives of projection factors are added so that the conflicting pairs can be distinguished.

The algorithm proposed by ? is implemented as the “solution formula construction phase” in QEPCAD. A small modification has been made such that this algorithm may be used to construct formulas for individual cells as opposed to the set of points satisfying the input formula. More precisely, the definition of a conflicting pair was slightly modified.

Definition 3.28. Let C_1 and C_2 be distinct cells in a CAD D of \mathbb{R}^n . Each cell C is assigned a truth value $T_C \in \{\mathbf{True}, \mathbf{False}, \mathbf{undet}\}$ and a list $S_C = (S_1, \dots, S_k)$ where $S_i = (s_{i,1}, \dots, s_{i,s_i}, 1 \leq i \leq k)$ is the list of signs of level- i (factorised) projection polynomials $f_1, \dots, f_{i_s} \in \mathbb{Z}[x_1, \dots, x_i]$, with $s_{i,j} \in \{-1, 0, 1\}$. (C_1, C_2) is called a conflicting pair in D if $T_{C_1}, T_{C_2} \in \{\mathbf{True}, \mathbf{False}\}$ and $T_{C_1} = T_{C_2}$ and $S_{C_1} = S_{C_2}$.

The condition on truth values being equal has been dropped so that every cell in the CAD may be distinguished by the signs of its projection polynomials only. In all other aspects, Brown’s solution formula construction algorithm may be applied unchanged. In order to represent a cell by a quantifier-free Boolean formula, it may be necessary to split a cell into several disjoint semialgebraic sets. Thus, each cell will be represented in disjunctive normal form, where each literal is a sign condition on a projection polynomial or one of its derivatives.

Chapter 4

Smooth Stratification

We now turn our attention to an algorithm for computing a smooth stratification of a semialgebraic set. A smooth stratification is a partition of a semialgebraic set into smooth manifolds. (?) proved that every complex (and real) analytic variety admits a smooth stratification. Later, (?) proved that every analytic set admits a smooth stratification and (?) proved that any set definable in an O-minimal structure also admits a Whitney stratification.

In this section, an algorithm due to (?) for computing a basic weak stratification of an elementary semi-pfaffian set, such that each of the strata is “nicely defined” will be presented.

Let $X \subset \mathbb{R}^n$ be a semianalytic set. X is nonsingular and has dimension k at a point $x_0 \in X$ if there exist real analytic functions

$$h_1, \dots, h_{n-k}$$

defined in an open set U containing x_0 such that

$$dh_1(x_0) \neq 0 \wedge \dots \wedge dh_{n-k}(x_0) \neq 0$$

and

$$X \cap U = \{x \in U \mid h_1(x) = 0 \wedge \dots \wedge h_{n-k}(x) = 0\}$$

Let $Y \subset \mathbb{R}^n$ be a semi-pfaffian set. $X \subset Y$ is called effectively nonsingular (of dimension k) if functions h_1, \dots, h_{n-k} belong to the same pfaffian chain as the functions defining Y .

Definition 4.1. A basic weak stratification of a semi-pfaffian set $X \subset \mathbb{R}^n$ is a partition of X into a finite number of non-singular manifolds $X_k, 0 \leq k \leq n$ where each X_k is effectively nonsingular and has codimension k .

4.0.1 Description of the algorithm

We now present the algorithm from ? as it is applied to semialgebraic sets. Since polynomials are a sub-class of Pfaffian functions, the algorithm can be applied directly to semialgebraic sets. However, in the original form, a bound on the number of partial derivatives is computed. For the semialgebraic case, we simply need to take the degree of polynomials defining the semialgebraic set.

Let $f \in \mathbb{Z}[x_1, \dots, x_n]$ be a polynomial and let $(m_1, \dots, m_n) \in \mathbb{Z}_{\geq 0}^n$ be a multi-index of its partial derivative. I.e., we write

$$\partial^{(m_1, \dots, m_n)}$$

to mean

$$\frac{\partial^{m_1} f}{\partial x_1} \dots \frac{\partial^{m_n} f}{\partial x_n}.$$

Definition 4.2. We define the partial differential operator $\partial_{\mathbf{h}, \mathbf{i}, j} f$ (where the argument f is a polynomial) as the determinant

$$\det \begin{pmatrix} \frac{\partial h_1}{\partial x_{i_1}} & \dots & \frac{\partial h_1}{\partial x_{i_k}} & \frac{\partial h_1}{\partial x_j} \\ & \ddots & & \\ \frac{\partial h_k}{\partial x_{i_1}} & \dots & \frac{\partial h_k}{\partial x_{i_k}} & \frac{\partial h_k}{\partial x_j} \\ \frac{\partial f}{\partial x_{i_1}} & \dots & \frac{\partial f}{\partial x_{i_k}} & \frac{\partial f}{\partial x_j} \\ \frac{\partial f}{\partial x_{i_1}} & \dots & \frac{\partial f}{\partial x_{i_k}} & \frac{\partial f}{\partial x_j} \end{pmatrix}.$$

We write $\partial_{\mathbf{h}, \mathbf{i}, j}^m$ to mean the m -th iteration of $\partial_{\mathbf{h}, \mathbf{i}, j}$.

We proceed by induction on i_0 .

- Base case: $i_0 = n$, return immediately.
- Recursive case:

– Initialise

$$X' := X.$$

- Consider each multi-index $\mathbf{m} = (i_n, \dots, i_{k+1}, j) \in \mathbb{Z}_{\geq 0}^{n-k} \times \{1, \dots, s\}$ in ascending lexicographical order, beginning with

$$(0, \dots, 0, 1, 1).$$

Index (i_n, \dots, i_1, i_0) has the following property:

$$M_{i_0} \preceq (i_n, \dots, i_1) \preceq (0, \dots, 1), M_{i_0} = \deg(f_i).$$

- Let $(\mathbf{J}, h_k) \in \mathbf{G}$ such that

$$\mathbf{J} = (0, \dots, 0, i_\ell - 1, i_{\ell-1}, \dots, i_{k+1}).$$

- Compute

$$s_k := \partial_{(h_1, \dots, h_k), (i_1, \dots, i_k), i_\ell} h_k$$

- Define

$$Y_k := \{\mathbf{x} \in X' \mid s_k(\mathbf{x}) \neq 0\}$$

(Note $h_1(\mathbf{x}) = 0$ for all $\mathbf{x} \in Y_k$.)

- Define

$$U_k := \{\mathbf{x} \in \mathbb{R}^n \mid h_k(\mathbf{x}) = 0, s_k(\mathbf{x}) \neq 0, g_1(\mathbf{x}) > 0, \dots, g_l(\mathbf{x}) > 0\},$$

a nonsingular subset of \mathbb{R}^n having codimension $k - 1$

- If Y_k is an open subset of U_k , then Y_k is nonsingular of codimension $k - 1$.

- * Update

$$X_k := X_k \cup Y_k.$$

- * Add polynomial s_k , i.e.,

$$\mathbf{G} := \mathbf{G} \cup \{(i_\ell, \dots, i_1, s_k)\}$$

- Otherwise, Y_k is not smooth. Call recursively

$$\text{Stratify}((k+1, \mathbf{G}, (i_\ell, i_k, \dots, i_1, 0), (s_k, h_k, \dots, h_1, 0), \mathbf{K}))$$

- Update

$$X' := \{\mathbf{x} \in X' \mid s_k(\mathbf{x}) = 0\},$$

a proper (possibly empty) subset of old X' .

- If $X' = \emptyset$, return. Otherwise, continue with the next index with respect to lexicographical order.

Let $X \subset \mathbb{R}^n := \{f_1 = 0 \wedge \dots \wedge f_k = 0 \wedge g_1 > 0 \wedge \dots \wedge g_l > 0\}$ be a semialgebraic set. We first apply some pre-processing to the polynomials defining X . Let

$$\mathbf{G} := ((0, \dots, 0, 1), f_1), \dots, ((0, \dots, 0, s), f_s).$$

Then the algorithm is called as follows

$$X = \text{Stratify}(0, \mathbf{G}, (0), (0), (g_1, \dots, g_t)).$$

A couple of shortcuts can be taken in the implementation to make the code simpler and more efficient. First, suppose that we are considering index

$$(0, \dots, 0, i_\ell, \dots, i_{k+1}, i_k, \dots, i_1, i_0).$$

Partial differentials with index lexicographically less than $(i_{k+1}, 0, \dots, 0)$ have been computed on a previous round of induction and were passed in the argument **G**. Furthermore, the function h_k , having index

$$(0, \dots, 0, i_{\ell-1}, \dots, i_0)$$

has already been computed, so it can be taken directly from the list **G**. In fact, there is a convenient way to find the function with index $(0, \dots, 0, i_{\ell-1}, \dots, i_1, i_0)$. Let us first illustrate this with an example. Let $M = (1, 2, 2)$ and consider the lexicographically ordered list of indices

$$\begin{aligned} L_1 &:= (0, 0, 0), (0, 0, 1), (0, 0, 2), (0, 1, 0), (0, 1, 1), (0, 1, 2), (1, 0, 0), (1, 0, 1) \dots \\ L_2 &:= (0, 0, 0), (0, 0, 1), (0, 0, 0), (0, 0, 1), (0, 0, 2), (0, 0, 0), (0, 0, 1) \dots \end{aligned}$$

Observe that for indices $(0, 0, 1)$, $(0, 1, 0)$ and $(1, 0, 0)$ in L_1 , the corresponding element in L_2 is $(0, 0, 0)$ and the corresponding elements for subsequent indices are those in the lexicographical order. I.e., we see that the index of h_k “chases after” the index of s_k , resetting each time ℓ is incremented. More precisely, when $\ell = 1$ and s_1 has index $(0, \dots, 0, i)$, $i > 0$, h_1 has index $(0, \dots, 0, i - 1)$. I.e., our initial “chase index” is the previous element in the list. When ℓ is incremented, we need to return to the first element in the list, $(0, \dots, 0)$ and proceed lexicographically through the indices until ℓ is incremented again, at which point we return to the beginning of the list. This gives us the property that when h_k has index $(0, \dots, 0, i_{\ell}, i_{\ell-1}, \dots, i_1)$, h_1 will have index $(0, \dots, 0, i_{\ell} - 1, i_{\ell-1}, \dots, i_1)$ while only requiring two very basic list operations: get the next element and return to the beginning. In the example, we considered only one polynomial. To extend to multiple polynomials, since each will have a different M , we need to keep a separate “chase list” for each one.

Since the program has been implemented in QEPCAD, which utilises linked lists, it was important to ensure that appending to the end of the list is as efficient as possible. To avoid $O(n)$ insertion complexity, we keep a pointer to the last-but-one element in the list, i.e., suppose $L = (a_1, (\dots, (a_{r-1}, (a_r, (NIL))) \dots)$ then $L_{\text{append}} = (a_{r-1}, (a_r, (NIL)))$. Then, if we wish to append b to the end of L , we simply need to set the reductum of L_{append} to $(a_r, (b, (NIL)))$.

Secondly, observe that only the last row and last column in the $(k \times k)$ -matrix for the partial differential operator

$$\partial_{\mathbf{h}, \mathbf{i}, j} f = \det \begin{pmatrix} \frac{\partial h_1}{\partial x_{i_1}} & \dots & \frac{\partial h_1}{\partial x_{i_k}} & \frac{\partial h_1}{\partial x_j} \\ \vdots & & & \\ \frac{\partial h_k}{\partial x_{i_1}} & \dots & \frac{\partial h_k}{\partial x_{i_k}} & \frac{\partial h_k}{\partial x_j} \\ \frac{\partial f}{\partial x_{i_1}} & \dots & \frac{\partial f}{\partial x_{i_k}} & \frac{\partial f}{\partial x_j} \end{pmatrix}.$$

depend on f and i . As a result, we can re-use the $(k-1 \times k-1)$ -matrix consisting of the first $k-1$ rows and columns of $\partial_{\mathbf{h}, \mathbf{i}, j} f$. In the implementation, we can

receive as an argument, from the previous round of induction, this $(k-1 \times k-1)$ -matrix which will be used throughout that round. We still, however, need to compute the determinant for every new s_k .

We will now present, from ([?], Theorem 1), the proof that this algorithm does indeed produce smooth strata and also that it terminates.

This algorithm is doubly exponential in the number of variables. For the polynomial case, the complexity is

$$s^n(d+1)^{2^{O(n)}}$$

where s is the number of polynomials f_1, \dots, f_s , d is their degree and n is the number of variables. ([?, Section 4])

4.0.2 Worked example

We illustrate the smooth stratification algorithm with a worked example. Consider

$$\{z = 0, x^2 - y^2 = 0\} \subset \mathbb{R}^3.$$

Note that the algorithm imposes an order on polynomials. In this case, $h_1 = z$ will be considered first. The algorithm proceeds as follows

$$k = 1, G_1 = (z, x^2 - y^2), J = (\partial h_1 / \partial y_1)$$

$$\begin{array}{llll} i = 1 & M = (0, 0, 1, 1), & h_1 = z, & s_1 = 0, \\ i = 1 & M = (0, 0, 1, 2), & h_1 = x^2 - y^2 & s_1 = 2x, \end{array} \quad \begin{array}{l} \text{skip} \\ Y_1 := \{z = 0, x^2 - y^2 = 0, 2x \neq 0\} = X \quad \{x = 0\} \\ U_1 := \{x^2 - y^2 = 0, 2x \neq 0\} \end{array} \quad (4.1)$$

Y_1 is not an open subset of U_1 , recurse.

$$k = 2, G_2 = (z, x^2 - y^2, 2x), J = \begin{pmatrix} 2x & \partial s_1 / \partial y_2 \\ \partial h_2 / \partial x & \partial h_2 / \partial y_2 \end{pmatrix}$$

$$\begin{array}{llll} i = 2 & M = (0, 1, 1), & h_2 = z, & s_2 = \det \begin{pmatrix} 2x & 2y \\ 0 & 0 \end{pmatrix} = 0, & \text{skip} \\ i = 2 & M = (0, 1, 2), & h_2 = x^2 - y^2 & s_2 = \det \begin{pmatrix} 2x & 2y \\ 2x & 2y \end{pmatrix} = 0 & \text{skip} \\ i = 2 & M = (0, 1, 3) & h_2 = 2x & \det \begin{pmatrix} 2x & 2y \\ 0 & 0 \end{pmatrix} = 0 & \text{skip} \\ i = 3 & M = (1, 0, 1) & h_1 = z & \det \begin{pmatrix} 2x & 0 \\ 0 & 1 \end{pmatrix} = 2x & Y_2 := Y_1 \cap \{2x \neq 0\} = Y_2 \\ & & & & U_2 := \{z = 0, 2x \neq 0\} \end{array} \quad (4.2)$$

Y_2 is an open subset of U_2 , hence $Y_2 = Y_1$ is a smooth subset of codimension 2. Set $X_{-2} = Y_{-2} = Y_{-1}$. Return to round $k = 1$. Recall that the last

index considered in step $k = 1$ was $(0, 0, 1, 2)$.

$$\begin{array}{llll}
 i = 1 & M = (0, 0, 1, 2), & h_1 = x^2 - y^2 & s_1 = 2x, & Y_1 := \{z = 0, x^2 = y^2\} \\
 \vdots & & & \text{skip until next nonzero partial derivative} & \\
 i = 2 & M = (0, 1, 0, 2) & h_1 = x^2 - y^2 & h_1 = 2y & Y_1 := \{z = 0, x^2 = y^2\} \\
 \vdots & & & \text{the rest of the partial derivatives do not exist} & \\
 & & & (4.3) &
 \end{array}$$

At the end of round $k = 1$, $X_1 = \emptyset$ and we are left with

$$X' := \{z = 0, x^2 + y^2 = 0, 2x = 0, 2y = 0\} = (0, 0, 0),$$

a single point. X_0 is set to X' and the algorithm terminates.

Note that this weak stratification does not impose any conditions on the boundaries of strata. In particular, the boundary of a stratum need not be a union of strata.

Chapter 5

Quasi-affine cells

Let

$$V := V_1 \cup \dots \cup V_k$$

be a family of semialgebraic sets such that $\dim(V_i) \leq 2$ for $1 \leq i \leq k$. According to ([?], Theorem 3.20), the first step in constructing a CAD with monotone cells is to construct a CAD D compatible with V such that each cell of D is the graph of a quasi-affine map. We do this following ([?, Lemma 3.19]).

5.1 Computing the smooth two-dimensional locus of V

We begin by computing the smooth 2-dimensional locus, W , of V . This can be done by computing a smooth stratification of V (as described in Section [?sec:smooth-strat]). Let

$$f_1, \dots, f_s$$

be the polynomials defining V . We use a slightly modified version of the smooth stratification algorithm which will avoid the expensive emptiness check. Instead of checking whether each set Y_k is empty, we just compute all possible partial differentials f_M for each $M = (i_n, \dots, i_1, j)$. A CAD which is sign-invariant with respect to f_1, \dots, f_s and all f_M will be compatible with V and the smooth two-dimensional locus W .

Alternatively, recall, from Section 3.1.1), that smooth cells can be guaranteed by choosing a certain projection operator. More specifically, according to ([?, Theorems 2.2.3 and 2.2.4]), if there is a finite number of blow-up points, which will be the case since input sets have dimension at most 2, then smooth cells will be obtained if McCallus's projection operator is used. Looking towards extending the construction to sets of higher dimension, [?] proved that we could

use Lazard's projection operator, which does not have the restriction on the dimension of blow-up points.

5.2 Ensuring that every cell is the graph of a quasi-affine map

In order to obtain quasi-affine cells, (? , Lemma 3.19) states that we need to consider the critical points, W_i and $W_{i,j}$, of projections of W onto one and two-dimensional coordinate subspaces (indexed by i and j) respectively. The classical CAD compatible with V, W and all W_i and $W_{i,j}$, $1 \leq i \leq j \leq n$ will contain quasi-affine cells.

Let f be one of the polynomials in f_1, \dots, f_s . Then the critical points of the projection map onto $\text{span}\{x_i\}$ can be found by computing the partial derivative and solving

$$\frac{\partial f}{\partial x_i} = 0.$$

The solutions of $\{f = 0, \partial f / \partial x_i = 0\}$ are the local maxima and minima of f along the x_i -axis. Similarly, the critical points of the projection map onto the two-dimensional subspace $\text{span}\{x_i, x_j\}$ are given by the solutions to the system

$$\{f = 0, \partial f / \partial x_i = 0, \partial f / \partial x_j\}.$$

Thus, W_i , the set of critical points of the projection map $\text{proj}_{\text{span}\{x_i\}}$, are given by $\bigvee_{1 \leq k \leq s} \{f_k = 0, \partial f_k / \partial x_i = 0\}$ and $W_{i,j}$, the set of critical points of $\text{proj}_{\text{span}\{x_i, x_j\}}$, is given by $\bigvee_{1 \leq k \leq s} \{f_k = 0, \partial f_k / \partial x_i = 0, \partial f_k / \partial x_j = 0\}$. Alternatively, we can compute a CAD which is sign-invariant with respect to the input polynomials f_1, \dots, f_s and all first partial derivatives (with respect to all variables x_1, \dots, x_n) of the input polynomials. Note that these first partial derivatives will have already been added if we applied the smooth stratification algorithm, which computes them for $k = 1$, to find W .

Since the smooth stratification algorithm has compleity doubly exponential in the number of variables, and this is also a bound on the number of strata (and therefore number of new polynomials to be added), it makes sense to avoid this computation if possible. Therefore, the choice was made to use McCallum's projection operator and compute a CAD sign-invariant with respect to

$$f_1, \dots, f_s$$

(the polynomials appearing in the formula defining V) and the first partial derivatives

$$\partial f_j / \partial x_i$$

for $1 \leq j \leq s$ and $1 \leq i \leq n$. Denote this CAD by D .

Chapter 6

Monotone Cells

Now we have a CAD D such that each 2-dimensional cell is the graph of a quasi-affine map, let us return to (? , Theorem 3.20) and discuss how to obtain monotone cells.

Definition 6.1. Let D be a CAD of \mathbb{R}^n and suppose that $c = \{x_1 = c_1, \dots, x_k = c_k\}$ is a 0-dimensional section cell in the decomposition induced by D on \mathbb{R}^k . By (? , Remark 3.8), the set of all cells of D contained in the effing coordinate subspace

$$\{x_1 = c_1, \dots, x_k = c_k\}$$

forms a cylindrical decomposition of \mathbb{R}^{n-k} . This decomposition will be called the *sub-decomposition* of D above c , and c the *base cell* of the sub-decomposition.

The construction proceeds by induction on n .

- The base case, $n = 1$ is straightforward, since (0) and (1)-cells are already monotone.
- When $n > 1$, consider each cell C of D and let $C'' := \text{proj op1}(C)$
 - If $\dim(C'') = 0$, apply the inductive hypothesis to the sub-decomposition of D above C'' .
 - Otherwise, C has index $\{1, i_2, \dots, i_n\} \in \{0, 1\}^n$. Let α be the smallest among $\{2, \dots, n\}$ such that $i_\alpha = 1$. Then $C' := \text{proj opa}(C)$ is a 2-dimensional sector cell and C is the graph of a quasi-affine map

$$\mathbf{f} = (f_1, \dots, f_{n-2}) : \text{span}\{x_1, x_\alpha\} \rightarrow \text{span}\{x_2, \dots, x_{\alpha-1}, x_{\alpha+1}, \dots, x_n\}.$$

Since \mathbf{f} is quasi-affine, each component $f_j : \text{span}\{x_1, x_\alpha\} \rightarrow \text{span}\{x_j\}$ is quasi-affine, too.

Let $X \subset \text{span}\{x_1, x_\alpha\}$. By (? , Lemma 3.18), there exists a refinement of the CAD of $\text{span}\{x_1, x_\alpha\}$, obtained by intersecting X with straight lines and half-planes of the kind

$$\{x_1 < c\}, \{x_1 = c\}, \{x_1 > c\}$$

where $c \in \mathbb{R}$, such that X is a union of semi-monotone sets B and $f_j|_B$ is a monotone function. By (? , Lemma 3.11), refinements of this kind preserve the cylindrical structure of D .

6.1 Two-dimensional semi-monotone sectors

Theorem 6.1. (? , Lemma 3.18)

Let $X \subset \mathbb{R}^2$ be an open, bounded set and $f : X \rightarrow \mathbb{R}$ be a quasi-affine function. Then there is a cylindrical decomposition of \mathbb{R}^2 compatible with X , obtained by intersecting X with straight lines

$$\{x_1 = c\}$$

and half-planes

$$\{x_1 < c\}, \{x_1 > c\}$$

such that the restriction $f|_B$ for every cell $B \subset X$ is a monotone function.

6.1.1 Two-dimensional semi-monotone sets

The proof of Theorem 6.1 proceeds in two stages: first, $X \subset \text{span}\{x_1, x_\alpha\}$ is refined such that each $Y \subset X$ is a semi-monotone set. Consider the intersection $X \cap \{x_1 = c\}$ for all $c \in \mathbb{R}$. This intersection is a finite union of pairwise disjoint intervals. Let $I(c)$ be the family of these intervals associated with the point c . Define

$$\gamma := \{(x_1, x_2) \in \mathbb{R}^2 \mid x_2 \text{ is an endpoint of an interval in } I(x_1)\}.$$

Lemma 6.1. Let C be a $(1, 0, \dots, 0, 1)$ -cell and $X := \text{proj}_{\text{span}\{x_1, x_\alpha\}}(C)$. Then

1. X is a cylindrical sector cell in \mathbb{R}^2 .
2. The family $I(C)$ is either empty or contains a single open interval for all $c \in \mathbb{R}$.
3. γ consists of the union of the top and bottom of X , each of which (if non-empty) is the graph of a continuous definable function.

Proof. Let $X' := \text{proj}_{\mathbb{R}^1}(C)$ and $C' := \text{proj}_{\mathbb{R}^{\alpha-1}}(C)$. Observe that X' is a (1)-cell and C' is a $(1, 0, \dots, 0)$ -cell. By definition

$$C = \{(\mathbf{x}, t) \mid \mathbf{x} \in C', f(\mathbf{x}) < t < g(\mathbf{x})\}$$

where $f, g : \mathbb{R}^{\alpha-1} \rightarrow \mathbb{R}$ are continuous definable functions such that $f(\mathbf{x}) < g(\mathbf{x})$ for all $\mathbf{x} \in C'$.

Since C' is a $(1, 0, \dots, 0)$ -cell, it can be viewed as the graph of a continuous definable map

$$\mathbf{h} = (h_1, \dots, h_{\alpha-2}) : X' \rightarrow \text{span}\{x_2, \dots, x_{\alpha-1}\}.$$

C can therefore be rewritten in terms of only x_1 and t .

$$C := \{(x_1, \mathbf{h}(x_1), t) \mid x_1 \in X', f(x_1, \mathbf{h}(x_1)) < t < g(x_1, \mathbf{h}(x_1))\}.$$

Define

$$\varphi : C'' \rightarrow \mathbb{R} \qquad \psi : C'' \rightarrow \mathbb{R} \qquad (6.1)$$

$$\varphi(x_1) = f(x_1, \mathbf{h}(x_1)) \qquad \psi(x_1) = g(x_1, \mathbf{h}(x_1)) \qquad (6.2)$$

Observe that φ and ψ are continuous, definable functions. By the definition of f and g , we also have $\varphi(x_1) < \psi(x_1)$ for all $x_1 \in X'$.

Thus, we can write

$$X = \{(x_1, t) \mid x_1 \in X', \varphi(x_1) < t < \psi(x_1)\}. \qquad (6.3)$$

It is clear that X is a cylindrical sector cell. By the definition of sector cell, the bottom of X is the graph of φ and the top of X is the graph of ψ .

Now consider the family

$$I(c) := \{t \mid \varphi(c) < t < \psi(c)\}.$$

Using Equation (6.3), it is easy to deduce that if $c \in X'$ then $I(c)$ contains a single open interval. On the other hand, if $c \notin X'$, $I(c)$ is clearly empty.

It follows that γ consists of the points of the top and bottom of X , which are the graphs of functions ψ and φ respectively. \square

The refinement is achieved by finding a set of real numbers

$$(c_1, \dots, c_t)$$

such that, for each $1 \leq i < t$,

$$X \cap \{c_i < x_1 < c_{i+1}\}$$

is a semi-monotone set. By (? , Theorem 1.7), this can be achieved by ensuring that

$$\gamma \cap \{c_i < x_1 < c_{i+1}\}$$

contains only monotone curve intervals.

Applying Lemma 6.1 to C , we obtain continuous definable functions $\varphi, \psi : X' \rightarrow \mathbb{R}$ defining the bottom, X_B , and top, X_T , of X respectively.

Let

$$(c_1, \dots, c_t)$$

be the critical points of functions φ and ψ . Then, for all $1 \leq i < t$, ϕ (resp ψ) is either strictly increasing in, strictly decreasing in, or independent of x_1 on the interval (c_i, c_{i+1}) .

6.1.2 Finding the critical points of φ and ψ

We now discuss how to find these critical points.

Let

$$A = (A_1, \dots, A_n), \tag{6.4}$$

where $A_i \subset \mathbb{Z}[x_1, \dots, x_i]$, be the projection factor set for D . For convenience, we say that $g(\mathbf{x}) = 0$ for some $g \in A_k$ and $\mathbf{x} \in \mathbb{R}^n$, such that $k < n$, if $g(\text{proj}_{\mathbb{R}^k}(\mathbf{x})) = 0$. Let C be a $(1, 0, \dots, 0, 1)$ -cell of the decomposition induced by D on \mathbb{R}^α and consider the $(1, 0, \dots, 0)$ -cell

$$C' := \text{proj}_{\mathbb{R}^{\alpha-1}}(C)$$

of the decomposition induced by D on $\mathbb{R}^{\alpha-1}$. Observe that

$$\text{proj}_{\mathbb{R}^1}(C) = (a, b) \subset \mathbb{R} \cup \{-\infty, \infty\}$$

and, since C' is a section cell, there exist polynomials

$$g_2 \in A_2, \dots, g_{\alpha-1} \in A_{\alpha-1}$$

such that $g_i(\mathbf{x}) = 0$ for all $1 \leq i \leq \alpha - 1$ and $\mathbf{x} \in (C')$. Therefore, there exists a one-dimensional algebraic variety

$$V' := \{g_2(\mathbf{x}) = 0, \dots, g_{\alpha-1}(\mathbf{x}) = 0\}$$

such that $C' \subset V'$.

Now consider the top of C , denoted by C_T . By definition, if C is not bounded from above, then C_T does not exist. Otherwise, C_T is the graph of a continuous definable function $h : C' \rightarrow \mathbb{R}$. In the first case, there is nothing to do, since C_T can be thought of as being independent of x_α . In the second case, C_T can be written as

$$\{(x_1, \dots, x_\alpha) \mid a < x_1 < b, g_2(x_1, \dots, x_{\alpha-1}) = 0, \dots, g_{\alpha-1}(x_1, \dots, x_{\alpha-1}) = 0, x_\alpha = h(x_1, \dots, x_{\alpha-1})\}. \tag{6.5}$$

By **TODO** C_T is C^∞ smooth, therefore h is a differentiable function everywhere in C' .

This representation lends itself naturally to the problem of Lagrange multipliers.

Proposition 6.1. *Let $f : \mathbb{R}^n \rightarrow \mathbb{R}$ and $g_1, \dots, g_k : \mathbb{R}^n \rightarrow \mathbb{R}$ be continuous functions with continuous first partial derivatives. Local maxima and minima of f , subject to the constraints $g_1 = 0, \dots, g_k = 0$ can be found by computing the critical points of the function*

$$\mathcal{L}(\mathbf{x}, \lambda_1, \dots, \lambda_k) = f(\mathbf{x}) - \lambda_1 g_1(\mathbf{x}) - \dots - \lambda_k g_k(\mathbf{x}),$$

where $\mathbf{x} \in \mathbb{R}^n$ and $(\lambda_1, \dots, \lambda_k) \in \mathbb{R}^k$ are new variables.

Critical points can be found by solving the system of equations

$$\frac{\partial f}{\partial x_i} - \lambda_1 \frac{\partial g_1}{\partial x_i} - \dots - \lambda_k \frac{\partial g_k}{\partial x_i} = 0, \quad (6.6)$$

where $1 \leq i \leq n - 1$,

for $\lambda_1, \dots, \lambda_k$.

Remark. Using variables $(x_1, \dots, x_{\alpha-1})$, constraints $g_2, \dots, g_{\alpha-1}$ and function h from Equation (6.5), the system from Equation (6.6) can be written in matrix form as follows.

$$\begin{pmatrix} \frac{\partial g_2}{\partial x_1} & \dots & \frac{\partial g_{\alpha-1}}{\partial x_1} \\ \vdots & & \vdots \\ \frac{\partial g_2}{\partial x_{\alpha-1}} & \dots & \frac{\partial g_{\alpha-1}}{\partial x_{\alpha-1}} \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_{\alpha-2} \end{pmatrix} = \begin{pmatrix} \frac{\partial h}{\partial x_1} \\ \vdots \\ \frac{\partial h}{\partial x_{\alpha-1}} \end{pmatrix}$$

It is clear that, if a solution exists, then the last column $\left(\frac{\partial h}{\partial x_1}, \dots, \frac{\partial h}{\partial x_{\alpha-1}}\right)^T$ will be a linear combination of the first $\alpha - 1$ columns (in $g_2, \dots, g_{\alpha-1}$). Hence, optimal values for $(x_1, \dots, x_{\alpha-1})$ exist and satisfy

$$\det \begin{pmatrix} \frac{\partial g_2}{\partial x_1} & \dots & \frac{\partial g_{\alpha-1}}{\partial x_1} & \frac{\partial h}{\partial x_1} \\ \vdots & & \vdots & \vdots \\ \frac{\partial g_2}{\partial x_{\alpha-1}} & \dots & \frac{\partial g_{\alpha-1}}{\partial x_{\alpha-1}} & \frac{\partial h}{\partial x_{\alpha-1}} \end{pmatrix} = 0. \quad (6.7)$$

Equation (6.7) gives a direct formula for formula to solve Equation (6.6) for $x_1, \dots, x_{\alpha-1}$. Refinement points (c_1, \dots, c_t) are the x_1 -coordinates of these solutions.

Since C_T is a section cell, there exists a polynomial $g_\alpha \in A_\alpha$ such that

$$C_T := \{\mathbf{x} \in \mathbb{R}^\alpha \mid \text{proj}_{\mathbb{R}^{\alpha-1}}(\mathbf{x}) \in C', g_\alpha(\mathbf{x}) = 0\}.$$

Thus, there exists an algebraic variety

$$V := \{g_2(\mathbf{x}) = 0, \dots, g_{\alpha-1}(\mathbf{x}) = 0, g_\alpha(\mathbf{x}) = 0\} \quad (6.8)$$

such that $C_T \subset V$, where $g_2, \dots, g_{\alpha-1}$ are the same polynomials appearing in Equation (6.5).

Note that $g_\alpha \in \mathbb{Z}[x_1, \dots, x_\alpha]$, while h from Equation (6.7) is expected to be a continuous differentiable function from $\mathbb{R}^{\alpha-1} \rightarrow \mathbb{R}$. We need to compute derivatives of the implicit g_α . I.e.,

$$\frac{dx_\alpha}{dx_i} = \frac{\partial g_\alpha / \partial x_i}{\partial g_\alpha / \partial x_\alpha} \quad (6.9)$$

for $1 \leq i \leq \alpha-1$. Let J_T be the determinant of the matrix of partial derivatives from Equation (6.7) with each $\partial h / \partial x_i$, $1 \leq i \leq \alpha-1$ replaced with dx_α / dx_i from Equation (6.9). Observe that the denominator of $\partial h / \partial x_i$, for all $1 \leq i \leq \alpha-1$, is equal to

$$d_T := \partial g_\alpha / \partial x_\alpha,$$

so the last column of the matrix can be written as

$$\frac{1}{d_T} (\partial g_\alpha / x_1, \dots, \partial g_\alpha / x_{\alpha-1})^T.$$

Thus, the determinant

$$J_T = \frac{1}{d_T} J'_T,$$

where J'_T is the matrix from Equation (6.7) with the $\partial h / \partial x_i$ replaced with $\partial g_\alpha / \partial x_i$ for $1 \leq i \leq \alpha-1$. I.e.,

$$\det \begin{pmatrix} \frac{\partial g_2}{\partial x_1} & \dots & \frac{\partial g_{\alpha-1}}{\partial x_1} & \frac{\partial g_\alpha}{\partial x_1} \\ \vdots & & & \vdots \\ \frac{\partial g_2}{\partial x_{\alpha-1}} & \dots & \frac{\partial g_{\alpha-1}}{\partial x_{\alpha-1}} & \frac{\partial g_\alpha}{\partial x_{\alpha-1}} \end{pmatrix} = 0. \quad (6.10)$$

Observe that J_T is a rational function: a fraction in which the numerator and denominator are both polynomials. $\mathbf{x} \in \mathbb{R}^{\alpha-1}$ is a solution of this rational function if and only if $J'_T(\mathbf{x}) = 0$ and $d_T(\mathbf{x}) \neq 0$. Thus, solutions of the system

$$\{J'_T = 0, d_T \neq 0\}$$

should be found. Observe that this is a semialgebraic set, since $\{d_T(\mathbf{x}) \neq 0\} \leftrightarrow \mathbb{R}^{\alpha-1} \setminus \{d_T(\mathbf{x}) = 0\}$. Note that, when $\deg(d_T) = 0$, d_T is a constant and therefore has no solutions. Thus, the condition $d_T = 0$ can be dropped.

Repeating this process to obtain J'_B and d_B , if sector cell C is bounded from below, we have a set of polynomial equations

$$\{g_2, \dots, g_{\alpha-1}, J'_B, J'_T\}, \quad (6.11)$$

and inequalities

$$\{s_1 g_{1,1} > 0, \dots, s_\ell g_{1,\ell} > 0, d_B \neq 0, d_T \neq 0\} \quad (6.12)$$

where $\{g_{1,1}, \dots, g_{1,\ell}\} \subset \mathbb{Z}[x_1]$ are polynomials from A_1 and $s_1, \dots, s_\ell \in \{-1, 1\}$ such that

$$x \in X' \leftrightarrow s_1 g_{1,1}(x) > 0 \wedge \dots \wedge s_\ell g_{1,\ell}(x) > 0.$$

We can easily find pairs $(s_j, g_{1,j}) \in \{-1, 1\} \times A_1$ by considering the sign of the sample point c of X' on each polynomial $g_{1,j} \in A_1$.

- If $g_{1,j} > 0$, then $s_j = 1$.
- If $g_{1,j}(c) < 0$, then $s_j = -1$.
- The final case, $g_{1,j}(c) = 0$, is impossible since an irreducible univariate polynomial cannot be zero over an interval. We want to find x_1 -coordinates of the roots of polynomials from Equation (6.11) subject to the constraints from Equation (6.12).

CAD can, of course, be used to do this, since we are really solving the quantifier elimination problem

$$\exists x_2, \dots, x_{\alpha-1}, x_\alpha P(x_1, \dots, x_\alpha)$$

where

$$P(x_1, \dots, x_\alpha) = s_1 g_{1,1}(x_1) > 0 \wedge \dots \wedge s_\ell g_{1,\ell}(x_1) > 0 \quad (6.13)$$

$$\wedge g_2(x_1, x_2) = 0 \wedge \dots \wedge g_{\alpha-1}(x_1, \dots, x_{\alpha-1}) = 0 \quad (6.14)$$

$$\wedge d_B(x_1, \dots, x_\alpha) \neq 0 \text{ and } d_T(x_1, \dots, x_\alpha) \neq 0 \quad (6.15)$$

$$\wedge (J'_B(x_1, \dots, x_{\alpha-1}) = 0 \vee J'_T(x_1, \dots, x_{\alpha-1}) = 0). \quad (6.16)$$

$$(6.17)$$

Computing the projection onto \mathbb{R}^1 of polynomials $\{g_2, \dots, g_{\alpha-1}, J'_B, J'_T\}$ from Equations (6.14) and (6.16), a system

$$F := \{f_1 = 0, \dots, f_r = 0\} \subset \mathbb{Z}[x_1]$$

is obtained, whose real roots include the refinement points (c_1, \dots, c_t) . Let $E \subset \mathbb{Z}[x_1]$ be the set of polynomials obtained by computing the projection onto \mathbb{R}^1 of $\{d_B, d_T\}$ (from Equation (6.15)) and

$$G := \{s_1 g_{1,1}, \dots, s_\ell g_{1,\ell}\}$$

from Equation (6.13). (c_1, \dots, c_t) can be found by computing the real roots of F subject to the inequalities G , then discarding those such that a polynomial in E is zero.

In the implementation **IPFSFB** is used, to compute a “finest square-free basis” of polynomials in F .

- this was just copying from Brown’s **CONSTRUCT** function. it returns a list of real roots in the form $[(M, J), \dots]$ where M is the minimal polynomial and J the isolating interval. It also computes multiplicities, which we don’t need.

- in fact, why do we even need a squarefree basis at all. should find out which is the most efficient algorithm from *sacLib*, of squarefree basis or the real root finding algorithms. maybe do some discussion on this.

It is clear that (c_1, \dots, c_t) lie in an open interval $X = (a, b) \subset \mathbb{R} \cup \{-\infty, \infty\}$. Thus, rather than using strict inequalities from Equation (6.13), we could find the roots of F in the interval (a, b) .

Each real root will be represented by a pair

$$c_i := (m_i, J_i)$$

where c_i is the unique root of the polynomial $m_i \in \mathbb{Z}[x_1]$ in the (left-open right-closed) isolating interval J_i . Note that B contains only algebraic numbers and roots will appear in ascending order.

6.1.3 Working with sub-decompositions above a 0-cell

Note that we are working in a sub-decomposition D of \mathbb{R}^{k+n} above a 0-cell \mathbf{c} of \mathbb{R}^k ($k \geq 0$). Therefore the polynomials appearing in Equation (6.8) are actually in $\mathbb{Z}[y_1, \dots, y_k][x_1, \dots, x_n]$. From a polynomial $f \in \mathbb{Z}[y_1, \dots, y_k][x_1, \dots, x_n]$, a polynomial $g := f(\mathbf{c}) \in \mathbb{A}[x_1, \dots, x_n]$ can be obtained by evaluating f at \mathbf{c} . Evaluated polynomials g define the sub-cad of \mathbb{R}^n above \mathbf{c} because \mathbf{c} is a 0-cell.

Note that, since \mathbf{c} is an algebraic number, g has algebraic coefficients. Since roots of g are algebraic numbers, there exists a polynomial $h \in \mathbb{Z}[x_1, \dots, x_n]$ such that, $g(\mathbf{x}) = 0$ if and only if $h(\mathbf{x}) = 0$ for all $\mathbf{x} \in \mathbb{R}^n$. A method for obtaining the polynomial h with integer coefficients is now described.

1. If f is independent of all y_1, \dots, y_k , there is nothing to do, just set $h := f$. Otherwise, f depends on at least one of y_1, \dots, y_k .
2. If \mathbf{c} is rational, then $g = f(\mathbf{c})$ has coefficients in \mathbb{Q} . There exists a non-zero $a \in \mathbb{Z}$ such that $h := ag$ has integer coefficients and the same roots as g .
3. Otherwise, \mathbf{c} is not rational. Then $f(\mathbf{c})$ has coefficients in \mathbb{A} . There exists an algorithm, which takes g as input and returns a polynomial $h \in \mathbb{Z}[x_1, \dots, x_n]$ such that $g(\mathbf{x}) = 0$ if and only if $h(\mathbf{x}) = 0$ for all $\mathbf{x} \in \mathbb{R}^n$.

Remark. Let $f \in \mathbb{Q}[x_1, \dots, x_n]$ be a polynomial such that

$$f(x_1, \dots, x_n) = \frac{a_1}{b_1}m_1 + \dots + \frac{a_k}{b_k}m_k$$

where $m_i = x_1^{d_{i,1}} \dots x_n^{d_{i,n}}$ for $1 \leq i \leq k$ is a monomial in x_1, \dots, x_n .

Let $M := \text{lcm}(b_1, \dots, b_k)$ be the least common multiple of denominators and construct

$$g(x_1, \dots, x_n) = Mf(x_1, \dots, x_n) = \frac{Ma_1}{b_1}m_1 + \dots + \frac{Ma_k}{b_k}m_k.$$

Each coefficient in g will be an integer number since M is divisible by all b_1, \dots, b_k and polynomials f and g have the same roots.

Definition 6.2. Let K be a field and L a finite (algebraic) extension of K . The field L can be thought of as a finite dimensional vector space over K .

Consider f and m as polynomials in $\mathbb{Q}[x_1, \dots, x_n, x_\alpha]$. compute the resultant of f and m to obtain a polynomial in $\mathbb{Q}[x_1, \dots, x_n]$. somehow this gives us the norm of an algebraic polynomial.

Remark. Let $f \in \mathbb{A}[x_1, \dots, x_n]$ be a polynomial such that

$$f(x_1, \dots, x_n) = a_1 m_1 + \dots + a_k m_k$$

where $m_i = x_1^{d_{i,1}} \dots x_n^{d_{i,n}}$ is a monomial in x_1, \dots, x_n and a_i is an element of $\mathbb{Q}(\alpha)$ for $1 \leq i \leq k$.

Let $m \in \mathbb{Z}[\alpha]$ be the minimal polynomial for $\mathbb{Q}(\alpha)$. f can be written as a polynomial in $\mathbb{Q}[x_1, \dots, x_n, \alpha]$ and m can be viewed as a polynomial in $\mathbb{Z}[x_1, \dots, x_n, \alpha]$ which is independent of all x_1, \dots, x_n . Compute

$$g := \text{Res}_\alpha(f, m)$$

to obtain a polynomial in $\mathbb{Q}[x_1, \dots, x_n]$. The property that $f(\mathbf{x}) = 0$ if and only if $g(\mathbf{x}) = 0$ follows from the following property of resultants:

Let $I = \langle f, m \rangle$ be the ideal generated by polynomials $f, m \in K[x_1, \dots, x_n][\alpha]$, where K is an algebraically closed field. If at least one of f and m is monic, then

$$\text{Res}_\alpha(f, m) \in I \cap R.$$

It follows that $\mathbf{x} \in K^n$ is a common zero of the elements of $I \cap R$ if and only if it is a zero of $\text{Res}_\alpha(f, m)$.

TODO this property came from the wiki page on resultants. does it need proving?

6.2 Two-dimensional monotone sections

Recall $X := \text{proj}_{\text{span}\{x_1, x_\alpha\}}(C)$ and let (c_1, \dots, c_r) be the refinement points computed in the previous section such that

$$Y := \{c_i < x_1 < c_{x+1}\} \cap X,$$

for some $1 \leq i \leq r$, is a two-dimensional semi-monotone set. Let us return to (?, Theorem 3.18) to further refine Y such that every two-dimensional section cell is a monotone cell.

Let $f_j : Y \rightarrow \text{span}\{x_j\}$ be a component of the quasi-affine map $\mathbf{f} : X \rightarrow \mathbb{R}$. Since \mathbf{f} is quasi-affine, each component of \mathbf{f} is quasi-affine and the restriction of component f_j to Y is quasi-affine since $Y \subset X$. Hence, f_j is either strictly increasing in, strictly decreasing in, or independent of each variable x_1, x_α . In addition, the restriction $f|_{Y \cap \{x_\alpha=c\}}$ for any $c \in \mathbb{R}$ is either strictly increasing in, strictly decreasing in, or independent of x_α . We will compute a refinement of Y such that $f_j|_Y$ is monotone, each $B \subset Y$ in the refinement. As before, only refinements of the kind

$$\{x_1 < c\}, \{x_1 = c\}, \{x_1 > c\},$$

$c \in \mathbb{R}$, which preserves both the cylindrical structure and existing monotone cells will be performed.

6.2.1 Case 1: $2 \leq j \leq \alpha - 1$

Lemma 6.2. *Let $Y \subset \mathbb{R}^n$ be the graph of a quasi-affine map*

$$\mathbf{f} : (a, b) \rightarrow \mathbb{R}^{n-1}$$

such that $(a, b) \subset \mathbb{R}$. Then Y is a monotone cell.

Proof. It is clear that

$$\dim(Y) = 1.$$

Hence, we need to show that Y is either strictly increasing in, strictly decreasing in, or independent of x_i for all $1 \leq i \leq n$. Consider the projection map $\text{proj}_{\text{span}\{x_i\}} : \mathbb{R}^n \rightarrow \text{span}\{x_i\}$.

Suppose that the restriction

$$\text{proj}_{\text{span}\{x_i\}}|_Y$$

is injective, Then it is clear that the image $\text{proj}_{\text{span}\{x_i\}}(Y)$ is 1-dimensional, since Y itself is 1-dimensional. On the other hand, suppose that the image

$$\text{proj}_{\text{span}\{x_i\}}(Y)$$

is one-dimensional, then it is clear that $\text{proj}_{\text{span}\{x_i\}}|_Y$ is injective if \mathbf{f} is either strictly increasing in, strictly decreasing in, or independent of x_i . It follows that \mathbf{f} is a monotone map, therefore its graph, Y , is a monotone cell. \square

Suppose that $2 \leq j \leq \alpha - 1$. Since

$$\text{proj}_{\text{span}\{x_j, x_\alpha\}}(C)$$

is one-dimensional, connected and the graph of a quasi-affine map, it follows from Lemma 6.2 that $\text{proj}_{\text{span}\{x_j, x_\alpha\}}(C)$ is a monotone cell and $f_j|_Y$ is already a monotone function. No refinement is needed.

6.2.2 Case 2: $\alpha + 1 \leq j \leq n$

Now suppose that $\alpha + 1 \leq j \leq n$. By (? , Theorem 3), if functions

$$\inf_{x_\alpha} f_j : \text{span}\{x_1\} \rightarrow \text{span}\{x_j\} \text{ and } \sup_{x_\alpha} f_j : \text{span}\{x_1\} \rightarrow \text{span}\{x_j\}$$

are monotone, then f_j itself is monotone. Hence, we need to find refinement points

$$\{b_1, \dots, b_r\} \subset (c_i, c_{i+1})$$

such that, for each $1 \leq \ell < r$, the restrictions of both $\inf_{x_\alpha} f_j$ and $\sup_{x_\alpha} f_j$ to $B := \{b_\ell < x_1 < b_{\ell+1}\}$ are monotone.

Consider

$$Z := \text{proj}_{\text{span}\{x_1, x_j\}}(C).$$

If $\dim(Z) < 2$, then it is clear that $\dim(Z) = 1$, because

$$Y' := \text{proj}_{\mathbb{R}^1}(C)$$

is one-dimensional. It follows that $\inf_{x_\alpha} f_j$ and $\sup_{x_\alpha} f_j$ coincide. Since Z is one-dimensional, connected and the graph of a quasi-affine map, Lemma 6.2 implies that f_j is already monotone and no refinement is needed.

On the other hand, if $\dim(Z) = 2$, since \mathbf{f} is quasi-affine, $\text{proj}_{\text{span}\{x_1, x_j\}}|_C$ is injective. Let Z_T be the graph of $\sup_{x_\alpha} f_j$. Despite the notation, Z_T is not necessarily the top of Z . In particular, Z_T may have non-empty intersection with Z . Since semialgebraic sets are closed under intersection and projection, there exist points

$$\{b_1, \dots, b_k\} \subset (c_i, c_{i+1})$$

such that for all $1 \leq i < k$

$$Z_T^{b_i, b_{i+1}} := Z_T \cap \{b_i < x_1 < b_{i+1}\}$$

is either a subset of Z or is disjoint from Z . First suppose that $Z_T^{b_i, b_{i+1}} \subset Z$ for some $1 \leq i < k$. In this case, Z is not bounded from above, so $\sup_{x_\alpha} f_j|_{\{b_i < x_1 < b_{i+1}\}}$ is independent of x_1 and is therefore already monotone. On the other hand, suppose that $Z_T^{b_i, b_{i+1}} \cap Z = \emptyset$. In this case, $Z_T^{b_i, b_{i+1}} \subset \text{proj}_{\text{span}\{x_1, x_j\}}(C_T)$. If there exists a point $b \in (b_i, b_{i+1})$ such that $\text{proj}_{\text{span}\{x_1, x_j\}}(C_T) \cap \{x_1 = b\} \cap Z \neq \emptyset$, then it is clear that $C_T \cap \{b_i < x_1 < b_{i+1}\}$ is not the graph of a continuous function. I.e., $\text{proj}_{\text{span}\{x_1, x_\alpha, x_j\}}(C_T)$ contains an open interval above $(b, y) \in Y_T$. Since $\dim(C) = 2$, all blow-up points (b, y) have dimension 0 and therefore there is a finite number of them. Repeating this process for $\alpha + 1 \leq j \leq n$, it follows that there exist points

$$\{b_{i,1}, \dots, b_{i,k_i}\} \subset (b_i, b_{i+1})$$

such that, for all $1 \leq j < k_i$, $C_T \cap \{b_{i,j} < x_1 < b_{i,j+1}\}$ is the graph of a continuous map. In particular, $C_T \cap \{b_{i,j} < x_i < b_{i,j+1}\}$ is the graph of $\mathbf{f}|_{Y_T \cap \{b_{i,j} < x_1 < b_{i,j+1}\}}$.

Apply the same argument to $\inf_{x_\alpha} f_j$ (for C_B) so that, C_T and C_B , if non-empty, are split into one-dimensional continuous curves which are graphs of continuous functions, and the intervals above 0-dimensional blow-up points. Once this property has been satisfied, we are left with a similar construction of optimisation with constraints as in Section 6.1.2.

6.2.3 Critical points of the top and bottom of C

We now discuss how to perform this refinement on the CAD structure. Let us assume that $C_T \neq \emptyset$. We must construct subsets $W \subset Y$ such that $f_j|_W$ is a smooth, continuous functions. Let

$$A = (A_1, \dots, A_n)$$

be the projection factor set as defined in Equation (6.4) and

$$C' := \text{proj}_{\mathbb{R}^{j-1}}(C), C^j := \text{proj}_{\mathbb{R}^j}(C),$$

Begin with $j := \alpha + 1$ so that C' is a cylindrical sector cell and there exists an algebraic variety V containing polynomials g_2, \dots, g_{j-1} with each $g_i \in A_i$, $2 \leq i \leq j-1$, defined in Equation (6.5), such that $C'_T \subset V$. By definition

$$C_T^j = \text{cl}(C^j) \cap (C'_T \times \mathbb{R})$$

and there exists a polynomial $g_j \in A_j$ such that $g_j(\mathbf{x}) = 0$ for all $\mathbf{x} \in C_T^j$. Since C_T^j is either empty or the graph of a continuous function everywhere except, possibly, for a finite number of blow-up points, we need to find x_1 -coordinates of points in C'_T such that g_j vanishes over an open interval. Since C_T^j is one-dimensional, one way to achieve this is by computing a smooth stratification of C_T^j . If the projection of a one-dimensional stratum onto x_1 is a 0-dimensional point $b \in \mathbb{R}$, then b is the x_1 -coordinate of a blow-up point.

Alternatively, since the refinement points form a 0-dimensional semialgebraic subset of \mathbb{R} , this set can be represented as a first-order Boolean formula

$$\exists x_2, \dots, x_{j-1} \exists_\infty x_j ((x_1, \dots, x_{j-1}) \in C'_T, g_j(x_1, \dots, x_j) = 0)$$

where $\exists_\infty x$ is a special quantifier defined in QEPcad meaning “there exist infinitely many values of x ”. Using \exists_∞ , some of the costly computations with algebraic numbers can be avoided, thereby improving efficiency in practice. **TODO refer to qepcad user documentation.** Theoretically speaking, (? , Algorithm 14.21) provides a singly exponential upper bound for quantifier elimination. **TODO what is the bound?**

Eliminating the $j-1$ quantifiers, we obtain the desired points $(b_{i,1}, \dots, b_{i,k_i})$, where each $b_{i,j}$ is the x_1 -coordinate of a blow-up point of g_j . Now we can

assume that $C_T^j \cap \{b_{i,j} < x_1 < b_{i,j+1}\}$ is the graph of a continuous function. More precisely, the implicit function g_j can be considered as a function from $C_T^j \cap \{b_{i,j} < x_1 < b_{i,j+1}\}$ to \mathbb{R} . We have a very similar construction as in Section 6.1.2. That is, polynomials g_2, \dots, g_{j-1} will provide the constraints in the first $j-2$ columns of the matrix in Equation (6.7), while $\partial h / \partial x_i, 1 \leq i \leq j-1$ in the last column should be replaced with $\partial g_j / \partial x_i$. Denote this polynomial by J_T , and repeat with C_B^j if appropriate. Inequalities $s_1 g_{1,1} > 0, \dots, s_1 g_{1,\ell} > 0$ (from Equation (6.13)), $d_T \neq 0$ and d_B (from Equation (6.15)) are obtained and solved for x_1 in a very similar way as in Section 6.1.2. We obtain a set of refinement points

$$\{b_{j,1}, \dots, b_{j,r_j}\} \subset (c_i, c_{i+1})$$

such that each of C_T^j and C_B^j is either empty or the graph of a continuous, monotone function.

The process is completed by induction in j . Suppose that the refinement has been performed for $\alpha + 1 \leq j$. If $j = n$ we are done, and C is now monotone. Otherwise, let $j := j + 1$. By the induction hypothesis, we know that $C_T^{j-1} \cap \{b_{j-1,i} < x_1 < b_{j-1,i+1}\}, 1 \leq i < r_{j-1}$ is either empty or the graph of a continuous function. If $C_T^{j-1} = \emptyset$, there is nothing to do and the process will terminate, since C_T^j is empty. Otherwise, take $g_{j-1} \in A_{j-1}$ used in the induction hypothesis and find $g_j \in A_j$ such that $g_j(\mathbf{x}) = 0$ for all $\mathbf{x} \in C_T^j$. Perform the same computation described above, to refine C_T^j such that one-dimensional components are graphs of continuous, monotone functions and repeat with C_B^j . As before, a set of refinement points

$$b_{j,1}, \dots, b_{j,r_j}$$

is obtained

When the process terminates, the union of sets of refinement points

$$\{b_{\alpha+1,1}, \dots, b_{\alpha+1,r_{\alpha+1}}\}, \dots, \{b_{n,1}, \dots, b_{n,r_n}\}$$

forms the refinement of C into two- and one-dimensional monotone cells.

6.3 Implementation Details

We now discuss implementation details of this step of the algorithm. First pseudocode is presented.

6.3.1 Algorithm

Input:

$$(E, A)$$

- E : a cylindrical algebraic decomposition of \mathbb{R}^{k+n} , such that each cell is the graph of a quasi-affine map. Each cell C of E has a truth-value (*true* or *false*) attached.
- $A = A_1, \dots, A_{k+n}$ is the family of projection polynomials where each $A_i \subset \mathbb{Z}[x_1, \dots, x_i]$.

Output:

$R := \{R_{\mathbf{b}} = \{c_1, \dots, c_t\} \subset \mathbb{A} \mid \mathbf{b} \in \mathbb{R}^{k-1} \text{ is a } (0, \dots, 0)\text{-cell in the decomposition induced by } E \text{ on } \mathbb{R}^k\}$

such that, for each $R_{\mathbf{b}}$, intersecting the sub-cad D above \mathbf{b} with straight lines and half-planes of the kind $\{x_1 < c_i\}, \{x_1 = c_i\}, \{x_1 > c_i\}, 1 \leq i \leq t$ results in a refinement of D such that each true cell of D with dimension at most 2 is monotone.

First, define a subroutine

$$\text{sub}_{\mathbf{y}}(g) \mid g \in \mathbb{Z}[y_1, \dots, y_m, x_1, \dots, x_n], \mathbf{y} \in \mathbb{A}^m, n > 0,$$

which computes $h := g(\mathbf{y}) \in \mathbb{A}[x_1, \dots, x_m]$ and returns the normalised polynomial of h , in $\mathbb{Z}[x_{m+1}, \dots, x_n]$ (see Section 6.1.3).

It will be convenient to overload this operator and write $\text{sub}_{\mathbf{y}}(L)$, where L is a set (or list) of polynomials to mean $\{\text{sub}_{\mathbf{y}}(g) \mid g \in L\}$.

Now define a structure to store refinement polynomials

$$F := \{F_{\mathbf{b}} := \emptyset \mid \mathbf{b} \in \mathbb{R}^{k-1}, k > 0 \text{ is a } (0, \dots, 0)\text{-cell in the decomposition induced by } E \text{ on } \mathbb{R}^{k-1}\}$$

and a similar structure to store refinement points

$$R := \{R_{\mathbf{b}} := \emptyset \mid \mathbf{b} \in \mathbb{R}^{k-1}, k > 0 \text{ is a } (0, \dots, 0)\text{-cell in the decomposition induced by } E \text{ on } \mathbb{R}^{k-1}\}$$

(Note: for each $(0, \dots, 0)$ -cell \mathbf{b} , $F_{\mathbf{b}}$ is a set of univariate polynomials, $R_{\mathbf{b}}$ is the set of roots of polynomials $F_{\mathbf{b}}$. When $k = 1$, then the CAD of \mathbb{R}^0 consisting of the unique point of \mathbb{R}^0 is used – practically speaking, we work directly with the CAD E .)

Let \mathbf{b} be a $(0, \dots, 0)$ -cell of the decomposition induced by E on $\mathbb{R}^{k-1}, k > 0$. Let D be the sub-decomposition of \mathbb{R}^n of E above \mathbf{b} .

Consider each cell C of D with truth value *true*. Let C have index

$$(0, \dots, 0, 1, i_{k+1}, \dots, i_{k+n})$$

in E . If $i_{k+1} + \dots + i_n > 2$, then $\dim(C) > 2$ and the algorithm will fail. Otherwise, let α be the smallest among $x + 1, \dots, k + n$ such that $i_{k+1} = \dots = i_{\alpha-1} = 0$ and $i_{\alpha} = 1$. Then:

- Let

- $\mathbf{b} := \text{proj}_{\mathbb{R}^{k-1}}(C)$,
(Note: observe that $\mathbf{b} \in A^{k-1}$. If $k = 1$, then \mathbf{b} is the unique cell, $\mathbf{0}$, in the CAD of \mathbb{R}^0 . In this case, it is clear that $\text{sub}_{\mathbf{0}}(g) \equiv g$.)
- $C''(C_B'', C_T'') \subset \mathbb{A} := \text{proj}_{\mathbb{R}^k}(C)$,
- $G := \text{sub}_{\mathbf{b}}\{g_i \mid k+1 \leq i \leq \alpha-1, g_i \in A_i, g_i(\mathbf{x}) = 0 \ \forall \mathbf{x} \in \text{proj}_{\mathbb{R}^{\alpha-1}}(C)\}$,
- $C' := \text{proj}_{\mathbb{R}^\alpha}(C)$.

- C'_T and C'_B be the top and bottom, respectively, C' .

(Note: If C' is not bounded from above (resp. below) by a continuous definable function, i.e., there is no section above (resp. below) C' , then let $C'_T := \emptyset$ (resp C'_B) and skip any computations involving C'_T (resp C'_B).)

- Let $j = \alpha$ and do:
 - $\text{**}(\text{*})\text{**}$ For $\delta \in \{B, T\}$, if C_δ is not empty, do:
 - * Let $\text{sub}_{\mathbf{b}}(g_j) \in \mathbb{Z}[x_{k+1}, \dots, x_j]$ be such that $g_\alpha(\mathbf{x}) = 0$ for all $\mathbf{x} \in C'_\delta$.
 - * Compute

$$f_\delta := \det \begin{pmatrix} \frac{\partial g_{k+1}}{\partial x_k} & \dots & \frac{\partial g_{j-1}}{\partial x_k} & \frac{\partial g_j}{\partial x_k} \\ \vdots & & & \vdots \\ \frac{\partial g_{k+1}}{\partial x_{j-1}} & \dots & \frac{\partial g_{j-1}}{\partial x_{j-1}} & \frac{\partial g_j}{\partial x_{j-1}} \end{pmatrix},$$

from Equation (6.7), using constraints $g_{k+1}, \dots, g_{\alpha+1}$, and discarding denominators $\partial g_\alpha / \partial x_\alpha$ of the total derivatives in the last column.

- Let

$$H := \{g_{k+1}, \dots, g_{j-1}, f_B, f_T\}$$

be a set of polynomials and compute

$$F_{\mathbf{b}} := F_{\mathbf{b}} \cup \text{proj}_{\mathbb{R}^1}(H) \subset \mathbb{Z}[x_k].$$

- If $j < n$, then
 - * Add g_j to G , i.e., let

$$G := G \cup \{g_j\}.$$

- * Let $j := j + 1$.

* Let

$$C' := \text{proj}_{\mathbb{R}^j}(C)$$

C' is a section cell, and its top and bottom may not be graphs of continuous functions. Use QE to compute

...

- For each $(0, \dots, 0)$ -cell $\mathbf{b} \in \mathbb{A}^k$, consider

$$F_{\mathbf{b}} \in F \in \mathbb{Z}[k],$$

compute

$$R_{\mathbf{b}} := (c_1, \dots, c_t)$$

by isolating the real roots of polynomials in $F_{\mathbf{b}}$.

- Return the family R of refinement points.

TODO - what if f_j does not exist - what if $\alpha = k + 1$

6.4 Computing the refinements

We now describe how to refine the CAD D to be compatible with the refinement points in R .

For $0 \leq k \leq n$, consider each refinement point $\mathbf{c} = (c_1, \dots, c_{k-1}, c_k) \in R_k$. It is important to note that refinement points appear in ascending order. Let D' be the sub-cad of \mathbb{R}^{n-k} above (c_1, \dots, c_k) , which is a $(0, \dots, 0)$ -cell of the CAD induced by D on \mathbb{R}^k and let D'' be the CAD induced by D' on \mathbb{R}^{k+1} . We want to refine D'' such that \mathbf{c} is a new (0) -cell. We need only to consider coordinate $k + 1$ of \mathbf{c} . Denote it by c_{k+1} . For each (0) -cell b_{k+1} of D'' , compute $s := \text{sign}(b_{k+1} - c_{k+1})$. If $s \leq 0$, consider the next (0) -cell in D'' . Otherwise, $s > 0$ and there is a (1) -cell C in D'' such that $c_{k+1} \in C$. This is the cell we need to refine. Make two new copies of C , so that we have three identical cells C_1, C_2, C_3 . Recall that, to each cell belongs a sample point and possibly some other information. E.g., in QEPCAD, each cell has a set of signs of projection factors, which should remain unchanged, and a positional index which describes where in the CAD the cell can be found. Let C_1 have positional index $(j_1, \dots, j_k, j_{k+1})$. New cells C_2 and C_3 should have positional indices $(j_1, \dots, j_k, j_{k+1} + 1)$ and $(j_1, \dots, j_k, j_{k+1} + 2)$ respectively. For each existing cell with positional index $(l_1, \dots, l_k, l_{k+1})$ with $l_k + 1 > j_{k+1}$, the index should be updated to $(l_1, \dots, l_k, l_{k+1} + 2)$. We now need to update the sample points. The sample point of the sector cell C_2 should be \mathbf{c} . This might turn out to be the case by chance. In this case, there is nothing to be done with C_2 and the sample points of sector cells C_1 and C_3 are incorrect. Otherwise, c_{k+1} is either an element of C_1 (sample point of C_1 is already correct) or c_{k+1} is an element of

C_3 (sample point of C_3 is already correct). Let C be the open interval (a_1, a_2) . If the sample point of C_1 needs to be updated, then set it to $(a_1 + c_{k+1})/2$ and if the sample point of C_3 needs updating, then set it to $(a_2 + c_{k+1})/2$. Observe that, if we have updated the sample point of a cell C , then the sample points of all cells in the sub-CAD above C are now incorrect. Since polynomials are delineable over the original cell, then they will be delineable over the smaller cells in the refinement, so we simply need to run the sample point computing part of the lifting phase recursively on the stack above C . This involves evaluating each level $k + 1$ projection factor f of the new sample point \mathbf{b} and finding the roots of the (possibly algebraic) univariate polynomial $f(\mathbf{b}, x_n)$.

Repeating this process with each refinement point in \mathbf{R}_k , we obtain the refinement described in (?, Lemma 3.11).

Chapter 7

Frontier Condition

Now suppose we have a CAD D' monotone with respect to V_1, \dots, V_k . Our final step, according to (?, Theorem 3.20) is to refine the frontier ∂C of each 2-dimensional cell $C \subset V$ so that ∂C is a union of cells of D' .

For each cell $C \subset V$, compute ∂C . Since C is monotone, we can assume that ∂C is homeomorphic to a circle. Construct a partition U of ∂C into monotone one-dimensional curves and points, such that U is compatible with all one-dimensional cells of D' . Let $\mathbf{c} = (c_1, \dots, c_n)$ be one of the one-dimensional components of U . If c_1 is not a 0-dimensional cell of the decomposition induced by D' on \mathbb{R}^1 , then perform a refinement of the kind $x_1 < c_1, x_1 = c_1, x_1 > c_1$. Recall that these refinements, according to (?, Lemma 3.11), preserve the cylindrical structure and monotone cells. Otherwise, c_1 is a 0-cell and we apply the same argument to the sub-CAD of D' above c_1 .

Now let $T \in U$ be one of the one-dimensional components. If T is not an existing 1-dimensional cell of D' , then it is a subset of a 2-dimensional cell, Z , of D' . In fact, it divides Z into two cells $Z \setminus T$, both of which, according to (?, Theorem 11), are 2-dimensional monotone cylindrical cells. Hence, replacing Z by T and the two components of $Z \setminus T$ makes Z compatible with ∂C and ensures that Z is refined into monotone cells.

Repeating this process with each 0- and 1-dimensional component of U , we obtain a refinement of D' compatible with ∂C . This completes the construction of a monotone CAD due to ?.

However, a difficulty arises when we try to translate this construction into an algorithm. We need to compute ∂C for each 2-dimensional cell $C \subset V$.

Lemma 7.1. *Let $S \subset \mathbb{R}^n$ be a semialgebraic set defined by a quantifier-free Boolean formula F containing s different polynomials in $\mathbb{R}[x_1, \dots, x_n]$ having maximum degree d . There is an algorithm, taking F as input, which repre-*

sents the semialgebraic set $\text{fr}(S)$ by a quantifier-free Boolean formula F' with complexity

$$(sd)^{O(n^2)}.$$

This is also an upper bound on the number of polynomials in F' and their degrees.

Proof. Observe that $\text{fr}(S)$ can be represented by a first-order Boolean formula

$$\text{fr}(S) = \{\mathbf{x} \in (\mathbb{R}^n \setminus S) \mid \forall \varepsilon > 0 \exists \mathbf{y} \in S (\|\mathbf{x} - \mathbf{y}\| < \varepsilon)\}.$$

Using singly-exponential quantifier elimination algorithm (? , Algorithm 14.21), we represent $\text{fr}(S)$ as a quantifier-free Boolean formula F' with the bounds required in the lemma. \square

We see that the “obvious” way of computing the frontier involves solving a quantifier elimination problem with two quantifier alternations. Even using an efficient QE algorithm, this still has complexity singly exponential in the number of variables. If possible, we would like to avoid calling this singly exponential subroutine on each cell.

7.1 Lazard’s method for dimension not greater than 3

? presents an algorithm for constructing an \mathbf{F} -invariant cylindrical algebraic decomposition of $\mathbb{R}^n, n \leq 3$ such that every cell is a topologically regular set and the frontier condition is satisfied.

According to (? , TODO), in dimension ≤ 2 , there is nothing to do. Furthermore, a cell C satisfies the frontier condition if no cells bordering it are bad.

Lemma 7.2. (? , Theorem 4.4) *Let \mathcal{D} be a sign-invariant decomposition of \mathbb{R}^n and C be a cell of \mathcal{D} with $C' := \text{proj}_{\mathbb{R}^{n-1}}(C)$. If the decomposition induced by D on \mathbb{R}^{n-1} is strong and none of C' and any cell adjacent to it is bad, C is well-bordered, and $\text{fr}(C)$ is a union of some cells of \mathcal{D} .*

Proof. This result follows from the following fact about roots of multivariate polynomials: given a polynomial $f \in \mathbb{Q}[x_1, \dots, x_{n-1}][x_n]$, i.e., considered as a univariate polynomial with polynomial coefficients, its complex roots depend continuously on the parameters in the region of the parameter space where the polynomial does not vanish identically. In fact, this is only true in the projective closure of \mathbb{C} as some roots may pass through the point at infinity if the leading coefficient is zero.

First suppose that C is a section cell. I.e., it is a root of some polynomial f and ∂C is defined as the limit of this root over the boundary of C' . Thus $\text{proj}_{\mathbb{R}^{n-1}}$ is a continuous, injective map from ∂C to $\partial C'$. This follows from the assumption

that the decomposition induced by D on \mathbb{R}^{n-1} is strong and that if the root defining the section C is infinite over some cell $C'' \subset C'$, then it is also infinite over $\partial C''$.

Now suppose that C is a sector cell having section cells (if they exist) C_B and C_T as the bottom and top of C respectively. Let us describe the boundary of C . We will only consider the case where $C \cap \{\mathbf{x} \times \mathbb{R}\}$ is an open interval bounded from below by a point in C_B and above by a point in C_T .

Let $\dim(C) = d + 1$. ∂C consists of

- C_B and C_T , both having dimension d ,
- the cells in ∂C_B and ∂C_T , which have dimension $< d$ and are contained in $\partial C' \times \mathbb{R}$,
- the intervals (which may be empty) bounded by the points in ∂C_B and ∂C_T , contained in $C' \times \mathbb{R}$. Since C_B and C_T are section cells, they are roots of polynomials f and g and, if finite, the points in ∂C_B and ∂C_T are also roots of f and g (respectively). Therefore, these intervals are contained in some cells of D . More precisely, the cells with dimension d contained in $\partial C' \times \mathbb{R}$ are exactly those lying over a cell $C'' \subset \partial C'$ with dimension $d - 1$.

A cell of dimension $< d$ either belongs to ∂C_B or ∂C_T , or is a sector over a cell $D \subset \partial C'$ having dimension less than $d - 1$. As C' is well-bordered, D belongs to the boundary of some cell $E \subset \partial C'$ having dimension $d - 1$. Thus, the sector above D is contained in the boundary of the sector above E , which has dimension d .

This proves that, if the decomposition induced by D on \mathbb{R}^{n-1} is strong, then C is well-bordered and also satisfies the frontier condition. In addition, this proves that C is a topologically regular cell as C and ∂C are either homeomorphic to C' and $\partial C'$, if C is a sector cell, or to an open cylinder, either bounded or unbounded, and its boundary. It is clear that an open cylinder contained in $C' \times \mathbb{R}$, where C' is topologically regular, is also topologically regular.

The other cases are deduced by omitting the cells defined from C_B and C_T as appropriate. \square

Let D be the CAD of \mathbb{R}^3 constructed by ? at the end of Section 5.2. The results in (?, Section 5.3) assume two properties of D . Firstly $\text{fr}(C)$, where $\dim(C) = k$, is the closure of the union of some cells of D of dimension $k - 1$. Secondly, according to (?, Proposition 5.13), any bad cells in the decomposition induced by D on \mathbb{R}^2 (those above which some polynomial vanishes over an open interval) are 0-dimensional isolated points.

Lemma 7.3. (*?, Proposition 5.13*) *Let \mathcal{D} be a sign-invariant decomposition of \mathbb{R}^3 , and suppose that C is a topologically regular cell in \mathcal{D} . Suppose that D is a cell of \mathcal{D} such that $B := D \cap \text{fr}(C) \neq \emptyset$ and $B \neq D$. Then either*

1. C is a 1-dimensional section cell whose endpoint, B , projects on a bad cell, or
2. there exists a cell E of \mathcal{D} such that $\dim(E) < \dim(C)$ and $\emptyset \neq D \cap \text{fr}(E) \neq D$.

To illustrate this, consider the example of the whitney umbrella $W := \{x^2 - y^2z = 0\}$. Let $C := \{x > 0, y > 0, x^2 - y^2z = 0\} \subset W$ be one of the 2-dimensional topologically regular cells in a CAD of \mathbb{R}^3 compatible with W . Then case 2 applies, with $D = \{x = 0, y = 0\}$ and $E = \{x > 0, y = 0, z = 0\}$. Choosing $C = E$, (and the same D) we get case 1.

By the proof of (TODO laz Th 4.4), D will always be a $(0, 0, 1)$ -cell in the cylinder above a bad cell D' .

Let us consider each case for C by cell index.

- $(1, 1, 1)$ -cell: since C is well-bordered, its boundary is the closure of some 2-dimensional cells of D and B is contained in the union of the boundaries of the 2-cells. Thus, case 2 applies and we can choose an appropriate 2-dimensional cell in $\text{fr}(C)$ for E .
- $(1, 0, 1)$ - or $(0, 1, 1)$ -cell, we write $(i, j, 1)$ -cell where $i + j = 1$: observe that C (if bounded) is delineated by some $(i, j, 0)$ -cells (one-dimensional curve intervals). Take B to be the one-dimensional interval delineated by the endpoints of these cells. Since $B \neq C$, at least one of these endpoints is not a $(0, 0, 0)$ -cell of D . Case 2 applies, taking as E the delineating $(i, j, 0)$ -cell.
- $(1, 0, 0)$ - or $(0, 1, 0)$ -cells: consider the previous case, letting $C = E$.
- $(1, 1, 0)$ -cell: these are necessarily the root of a polynomial $f \in \mathbb{Q}[x_1, x_2, x_3]$ which vanishes identically on D' , a $(0, 0)$ cell in the decomposition induced by D on \mathbb{R}^2 . Let V_r be a small neighbourhood (with radius $r > 0$) centered around D' and consider V'_r , the cylinder above the boundary of V_r . Since C is a monotone cell, the maximum and minimum values of x_3 in this intersection are reached on the boundary of C , for sufficiently small r . The limits, as r tends to 0, of this maximum and minimum clearly belong to the boundary of B , if these limits do not tend to infinity. Note that both the maximum and minimum cannot be infinite, since this would imply that either B is empty, or coincides with D . Therefore, the non-infinite limits are the endpoints of some 1-dimensional cell belonging to the boundary of C . Case 2 applies, taking E to be this 1-dimensional cell.
- $(0, 0, 1)$ - or $(0, 0, 0)$ -cells: cannot satisfy the hypothesis, there is nothing to do.

(?, Proposition 5.14) Let $f \in \mathbb{Q}[x_1, x_2, x_3]$ be an irreducible polynomial given by

$$f = g_d x^d + \dots + g_1 x + g_0$$

with each $g_i \in \mathbb{Q}[x_1, x_2]$, and

$$\mathbf{p} = (p_1, p_2)$$

be a common root of g_d, \dots, g_1, g_0 . Let $g \in \mathbb{Q}[x_1, x_2]$ be an irreducible polynomial having \mathbf{p} as a root. The limits of the common roots of f and g as $(x_1, x_2) \rightarrow \mathbf{p}, (x_1, x_2) \neq \mathbf{p}$ may be computed as solutions of a zero-dimensional polynomial system.

Proof. Since f is irreducible, it has at least two coefficients, $g_i, g_j \in \mathbb{Q}[x_1, x_2]$, whose GCD is a constant. Therefore, any ideal generated by these coefficients has dimension zero. Any ideal containing these coefficients also contains another element, h , which is not a multiple of g .

The saturation by h of the ideal $\langle f, g \rangle$ is defined as the ideal

$$S := \langle f, g, 1 - zh \rangle \cap \mathbb{Q}[x_1, x_2, x_3].$$

The irreducible components of the zero set of S are those of the zero set of $\langle f, g \rangle$ at which h is not identically zero. It follows that the vertical line $\{x_1 = p_1, x_2 = p_2\}$ which is part of the zero set of f, g and h , is not contained in the zero set of S . Meanwhile, the zero set of S contains the limit points we are interested in finding.

It follows that the ideal generated by S and the coefficients g_0, \dots, g_d of f is zero-dimensional and the limits of $\{f = 0, g = 0\}$ as $(x_1, x_2) \rightarrow \mathbf{p}$ can be computed as the zeros of S . \square

7.1.1 An aside on polynomial ideals and Zariski closures

? uses the trick of saturating the ideal $\langle f, g \rangle$ by some h , which is not a multiple of g . The key to this trick lies in Zariski closures. Let us take an aside to discuss this. We closely follow ? and all results are proved in a generic polynomial ring $K[x_1, \dots, x_n]$ unless otherwise stated.

7.1.1.1 Ideals, Groebner bases and Buchberger's algorithm

Definition 7.1. Let $K[x_1, \dots, x_n]$ be a polynomial ring. A subset $I \subset K[x_1, \dots, x_n]$ is called a polynomial ideal if

- $0 \in I$,
- if $a, b \in I$ then $a + b \in I$, and
- if $a \in I, b \in K[x_1, \dots, x_n]$, then $ab \in I$.

Definition 7.2. Let K be a field and $f_1, \dots, f_k \in K[x_1, \dots, x_n]$. We set

$$\langle f_1, \dots, f_k \rangle = \{h_1 f_1 + \dots + h_k f_k \mid h_1, \dots, h_k \in K[x_1, \dots, x_n]\}$$

and call it the ideal generated by f_1, \dots, f_k .

? proved, nonconstructively, that every polynomial ideal is finitely generated.

Theorem 7.1. *Every ideal $I \subset k[x_1, \dots, x_n]$ has a finite generating set. I.e., $I = \langle g_1, \dots, g_t \rangle$ for some finite collection of polynomials $g_1, \dots, g_t \in I$.*

A proof can be found in (?, p76).

Later, ? published an algorithm for computing this finite set of generators. He also introduced the concept of a Groebner basis.

A Groebner basis requires an ordering on monomials to be fixed.

Definition 7.3. A monomial ordering on $K[x_1, \dots, x_n]$ is a relation \succ on $\mathbb{Z}_{\geq 0}^n$, or equivalently, a relation on the set of monomials $x^\alpha, \alpha \in \mathbb{Z}_{\geq 0}^n$, satisfying the following properties.

- \succ is a total order¹ (or linear order) on $\mathbb{Z}_{\geq 0}^n$.
- If $\alpha \succ \beta$ and $\gamma \in \mathbb{Z}_{\geq 0}^n$, then $\alpha + \gamma \succ \beta + \gamma$.
- \succ is a well-ordering on $\mathbb{Z}_{\geq 0}^n$. I.e., every nonempty subset of $\mathbb{Z}_{\geq 0}^n$ has a minimal element under \succ .

Some examples of monomial orderings used in computing Groebner bases include

- **lex:** Let $\alpha = (\alpha_1, \dots, \alpha_n), \beta = (\beta_1, \dots, \beta_n) \in \mathbb{Z}_{\geq 0}^n$. We say $\alpha \succ \beta$ if the leftmost nonzero element of $\alpha - \beta \in \mathbb{Z}^n$ is positive. This imposes a lexicographic order on variables $x_1 \succ x_2 \succ \dots \succ x_n$.
- **grlex:** (graded lex) Let $\alpha, \beta \in \mathbb{Z}_{\geq 0}^n$ and let $|\alpha| = \alpha_1 + \dots + \alpha_n$ and $|\beta| = \beta_1 + \dots + \beta_n$. We say $\alpha \succ \beta$ if $|\alpha| > |\beta|$, or if $|\alpha| = |\beta|$ and $\alpha \succ_{\text{lex}} \beta$.
- **grevlex:** (graded reverse lex) Let $\alpha, \beta \in \mathbb{Z}_{\geq 0}^n$, and define $|\alpha|$ and $|\beta|$ as before. We say $\alpha \succ_{\text{grevlex}} \beta$ if $|\alpha| > |\beta|$ or if $|\alpha| = |\beta|$ and the rightmost nonzero entry of $\alpha - \beta$ is negative.

Definition 7.4. Let f be a polynomial in $K[x_1, \dots, x_n]$ with monomials labelled by indices in \mathbb{Z}^n , of the form $\sum_{\alpha} a_{\alpha} x_1^{\alpha_1} \dots x_n^{\alpha_n}$, and fix a monomial order \succ . Define the following

- $\text{multideg}(f) = \max(\alpha \in \mathbb{Z}_{\geq 0}^n \mid a_{\alpha} \neq 0)$,
- $\text{lc}(f) = a_{\alpha} \mid \alpha = \text{multideg}(f)$,
- $\text{lm}(f) = x_1^{\alpha_1} \dots x_n^{\alpha_n} \mid \alpha = \text{multideg}(f)$.
- $\text{lt}(f) = \text{lc}(f) \cdot \text{lm}(f)$

¹Total order: a reflexive, antisymmetric, and transitive relation R on a set X such that either $(a, b) \in R$ or $(b, a) \in R$ for all $a, b \in X$.

Definition 7.5. Fix a monomial ordering \succ on $K[x_1, \dots, x_n]$. Let $I \subset K[x_1, \dots, x_n]$ be an ideal. A finite set $G = \{g_1, \dots, g_t\} \subset I$ is called a Groebner basis if

$$\langle \text{lt}(g_1), \dots, \text{lt}(g_t) \rangle = \langle \text{lt}(I) \rangle$$

where $\text{lt}(I) = \{\text{lt}(f) \mid f \in I\}$

Definition 7.6. We write \bar{f}^F for the remainder of division of f by the tuple $F = (g_1, \dots, g_t)$. If F is a Groebner basis, F can be considered as an (unordered) set by (? , p83, Proposition 1).

Definition 7.7. Let $f, g \in K[x_1, \dots, x_n]$ such that $\text{multideg}(f) = \alpha$ and $\text{multideg}(g) = \beta$. Define $\gamma = \max(\alpha, \beta)$ (where max is taken component-wise). $x_1^{\gamma_1} \cdots x_n^{\gamma_n} = \text{lcm}(\text{lm}(f), \text{lm}(g))$ is called the least common multiple of the leading monomials of f and g .

Lemma 7.4. Let $f, g \in K[x_1, \dots, x_n]$ be nonzero polynomials. Then

1. $\text{multideg}(fg) = \text{multideg}(f) + \text{multideg}(g)$,
2. If $f + g \neq 0$ then $\text{multideg}(f + g) \leq \max(\text{multideg}(f), \text{multideg}(g))$. As a special case, if $\text{multideg}(f) \neq \text{multideg}(g)$ then equality occurs.

Proof. Write $f = ax^\alpha + f'$ and $g = bx^\beta + g'$ so that $\text{multideg}(f) = \alpha$ and $\text{multideg}(g) = \beta$. Observe that $\text{multideg}(f') < \alpha$ and $\text{multideg}(g') < \beta$ and $\text{lm}(f) \cdot \text{lm}(g) = x^\alpha \cdot x^\beta = x^{\alpha+\beta}$. Since $\text{multideg}(f') < \alpha$ and $\text{multideg}(g') < \beta$, we have $\text{multideg}(f' \cdot g') < \alpha + \beta$. Hence $\text{multideg}(f \cdot g) = \alpha + \beta$. This proves property 1.

Now consider $(ax^\alpha + f') + (bx^\beta + g') = (ax^\alpha + bx^\beta) + (f' + g')$. We have $\text{multideg}(f') < \alpha$ and $\text{multideg}(g') < \beta$ as before, so it suffices to consider $ax^\alpha + bx^\beta$. If $\alpha \neq \beta$ then $\text{multideg}(f + g) = \max(\alpha, \beta)$. If $\alpha = \beta$ then if $a + b \neq 0$ we have $ax^\alpha + bx^\alpha = (a + b)x^\alpha$, hence $\text{multideg}(f + g) = \alpha$. Otherwise we have $ax^\alpha + bx^\alpha = 0x^\alpha$ so $\text{multideg}(f + g) = \text{multideg}(f' + g') < \alpha$. This proves property 2. \square

Lemma 7.5. Let $I \subset K[x_1, \dots, x_n]$ be an ideal, $G = \{g_1, \dots, g_t\}$ be a Groebner basis for I and f be an arbitrary polynomial in $K[x_1, \dots, x_n]$. Then there exists a unique $r \in K[x_1, \dots, x_n]$ such that

1. no term of r is divisible by any of $\text{lt}(g_1), \dots, \text{lt}(g_t)$, and
2. there exists $g \in I$ such that $f = g + r$.

In particular, r is the remainder of division of f by all elements of G (taken in any order).

Proof. Firstly, using division, we can write $f = h_1g_1 + \dots + h_tg_t + r$, hence r is not divisible by any polynomials in G (first condition). This decomposition of f can be used to set $g = h_1g_1 + \dots + h_tg_t \in I$, so that $f = g + r$ (satisfying the second condition).

Now, to prove that r is unique, suppose that $f = g + r = g' + r'$ for g, g' and r, r' satisfying both conditions. Then $r - r' = g - g' \in I$ (by definition of an ideal). Assume $r \neq r'$, then $\text{lt}(r - r') = \langle \text{lt}(I) \rangle = \langle \text{lt}(g_1), \dots, \text{lt}(g_t) \rangle$ since G is a Groebner basis for I . By (? p70, Lemma 2), this implies $\text{lt}(r - r')$ is divisible by some $\text{lt}(g_i)$, which is a contradiction since r and r' cannot be divisible by any of the $\text{lt}(g_i)$'s.

The final part of the proposition follows from uniqueness of r . \square

Corollary 7.1. *Let $G = \{g_1, \dots, g_k\}$ be a Groebner basis for an ideal $I \subset K[x_1, \dots, x_n]$ and let $f \in K[x_1, \dots, x_n]$. Then $f \in I$ if and only if the remainder on division of f by G is zero.*

Proof. If the remainder on division is zero, then we already proved that $f \in I$. Conversely, if $f \in I$ then $f = f + 0$ satisfies the two conditions of Lemma~7.5. It follows that the remainder on division of f by G is zero. \square

Definition 7.8. Let $f, g \in K[x_1, \dots, x_n]$. The S -polynomial of f and g is defined as

$$S(f, g) = \frac{\text{lcm}(\text{lt}(f), \text{lt}(g))f}{\text{lt}(f)} - \frac{\text{lcm}(\text{lt}(f), \text{lt}(g))g}{\text{lt}(g)}.$$

S -polynomials are constructed to reduce the degree of terms, as shown in the following lemma.

Lemma 7.6. *Suppose that $f \in K[x_1, \dots, x_n]$ is a sum of polynomials $\sum_{i=1}^s p_i$ such that $\text{multideg}(p_i) = \delta$ for all $1 \leq i \leq s$. If $\text{multideg}(\sum_{i=1}^s p_i)$ is a linear combination $\sum_{(i,j) \in \{1, \dots, s\}^2} a_{ij} S(p_i, p_j)$ where all $a_{ij} \in K$. Furthermore all $\text{multideg}(S(p_i, p_j)) < \delta$.*

Proof. We can write each $p_i = c_i(x_1^{\delta_1} \dots x_n^{\delta_n} + g_i)$ where $\delta = (\delta_1, \dots, \delta_n)$, $c_i \in K$ and $\text{multideg}(g_i) < \delta$. In order for $\text{multideg}(\sum_{i=1}^s p_i) < \delta$, leading monomials of p_i 's must cancel in the sum. I.e., $\sum_{i=1}^s c_i \text{lt}(p_i) = 0$ implies $\sum_{i=1}^s c_i = 0$.

Since $\text{multideg}(p_i) = \text{multideg}(p_j)$ for all i, j , $\text{lcm}(\text{lt}(p_i), \text{lt}(p_j)) = x_1^{\delta_1} \dots x_n^{\delta_n} = h = \frac{1}{c_i} \text{lt}(p_i) = \frac{1}{c_j} \text{lt}(p_j)$. Consider

$$S(p_i, p_j) = \frac{\text{lcm}(p_i, p_j)p_i}{\text{lt}(p_i)} - \frac{\text{lcm}(p_i, p_j)p_j}{\text{lt}(p_j)} = \frac{hp_i}{c_i h} - \frac{hp_j}{c_j h} = \frac{1}{c_i} p_i - \frac{1}{c_j} p_j.$$

Hence

$$S(p_i, p_j) = \sum_{i=1}^s c_i \frac{1}{c_i} (x_1^{\delta_1} \cdots x_n^{\delta_n} + g_i) - \frac{1}{c_j} c_j (x_1^{\delta_1} \cdots x_n^{\delta_n} + g_j) = \sum_{i=1}^s g_i - g_j.$$

Since $\text{multideg}(g_i) < \delta$ and $\text{multideg}(g_j) < \delta$, their linear combination, $S(p_i, p_j)$ has multidegree less than δ .

Now return to the sum

$$\sum_{i=1}^s p_i = \sum_{i=1}^s c_i (x_1^{\delta_1} \cdots x_n^{\delta_n} + g_i).$$

Since terms of the kind $x_1^{\delta_1} \cdots x_n^{\delta_n}$ cancel,

$$\sum_{i=1}^s p_i = \sum_{i=1}^s c_i g_i = c_1 g_1 + \cdots + c_s g_s.$$

We show that this is a linear combination of S -polynomials.

Consider the sum

$$\sum_{i=1}^{s-1} c_i S(p_i, p_s) = \sum_{i=1}^{s-1} c_i (g_i - p_s) = \sum_{i=1}^{s-1} c_i g_i - c_i p_s,$$

hence

$$\sum_{i=1}^{s-1} c_i S(p_i, p_s) = c_1 g_1 + \cdots + c_{s-1} g_{s-1} - (c_1 + \cdots + c_{s-1}) p_s.$$

Since $c_1 + \cdots + c_{s-1} + c_s = 0$, we have $c_1 + \cdots + c_{s-1} = -c_s$, and we can write

$$\sum_{i=1}^{s-1} c_i S(p_i, p_s) = g_1 + \cdots + g_{s-1} - (-c_s) \cdot p_s = \sum_{i=1}^s c_i g_i,$$

proving that $\sum_{i=1}^s p_i$ is a linear combination of S -polynomials. \square

We are now ready to present Buchberger's Criterion, the condition under which Buchberger's algorithm will terminate.

Lemma 7.7. *Let I be a polynomial ideal. Then a basis $G = \{g_1, \dots, g_t\}$ of I is a Groebner basis of I if and only if, for all pairs $i \neq j$, the remainder from division of $S(g_i, g_j)$ by G (with a fixed order) is zero.*

Proof. \Rightarrow : If G is a Groebner basis for I , then $S(g_i, g_j) \in I$ for all $g_i, g_j \in G$ and, by Corollary 7.1, the remainder of $\overline{S(g_i, g_j)}^F$ is zero.

\Leftarrow : Let $f \in I$ be a nonzero polynomial. We need to show that $\text{lt}(f) \in \langle \text{lt}(g_1), \dots, \text{lt}(g_k) \rangle$. We have to show that

$$f = \sum_{i=1}^k h_i g_i$$

where $h_i \in K[x_1, \dots, x_n]$, $1 \leq i \leq k$. By Lemma 7.4, we have

$$\text{multideg}(f) \leq \max(\text{multideg}(h_i g_i) \mid h_i g_i \neq 0).$$

We want to try to find the “most efficient” representation of f . I.e., one where

$$\delta = \max(\text{multideg}(h_i g_i) \mid h_i g_i \neq 0)$$

is minimal. Such a δ exists by the well-ordering property (a minimal element always exists). We have

$$\text{multideg}(f) \leq \delta = \max(\text{multideg}(h_i g_i) \mid h_i g_i \neq 0)$$

First consider the case when $\text{multideg}(f) = \delta$. We have $\text{multideg}(f) = \text{multideg}(h_i g_i)$ for some $1 \leq i \leq k$. Therefore $\text{lt}(f)$ is divisible by $\text{lt}(g_i)$. Hence $\text{lt}(f) \in \langle g_i, \dots, g_k \rangle$.

Now suppose that $\text{multideg}(f) < \delta$. We can write f as follows

$$\sum_{i=1}^k h_i g_i = \sum_{i=1}^k \text{lt}(h_i) g_i + \sum_{i=1}^k (h_i - \text{lt}(h_i)) g_i$$

and use this representation to isolate terms with maximum multidegree

$$\sum_{\text{multideg}(h_i g_i) = \delta} \text{lt}(h_i) g_i + \sum_{\text{multideg}(h_i g_i) = \delta} (h_i - \text{lt}(h_i)) g_i + \sum_{\text{multideg}(h_i g_i) < \delta} h_i g_i.$$

The last two terms have $\text{multideg} < \delta$, so we attempt to rewrite $\sum_{\text{multideg}(h_i g_i) = \delta} \text{lt}(h_i) g_i$ to decrease δ , thereby obtaining a contradiction. Applying Lemma 7.6 with $p_i = \text{lt}(h_i) g_i$ with multidegree δ and f with multidegree $< \delta$, we can write $\sum_{\text{multideg}(h_i g_i) = \delta} \text{lt}(h_i) g_i$ as a linear combination (with coefficients in K) of S -polynomials having multidegree $< \delta$. Hence, we know that f can be rewritten as a sum of polynomials with multidegree $< \delta$, contradicting the assumption that δ is minimal.

Given an expression $f = \sum_{i=1}^t h_i g_i$ with minimal δ , we begin by isolating the part of the sum having multidegree δ .

$$f = \sum_{\text{multideg}(h_i g_i) = \delta} h_i g_i + \sum_{\text{multideg}(h_i g_i) < \delta} h_i g_i \quad (7.1)$$

$$= \sum_{\text{multideg}(h_i g_i) = \delta} (h_i - \text{lt}(h_i)) g_i + \sum_{\text{multideg}(h_i g_i) < \delta} h_i g_i \quad (7.2)$$

The second and third sums in Equation (7.2) have multidegree $< \delta$. Since $\text{multideg}(f) < \delta$, the multidegree of the first sum in Equation (7.2) must also have $\text{multideg} < \delta$.

The key to decreasing δ is to rewrite the first term in this sum in two stages:

- use (?, Lemma 5) to rewrite the first term in the sum in terms of S -polynomials,
- then use $S(g_i, g_j) = 0$ to rewrite the S -polynomials without cancellation.

By (?, Lemma 5),

$$\sum_{(\text{multideg})(h_i g_i) < \delta} \text{lt}(h_i) g_i$$

is a linear combination of the coefficients in K of the S -polynomials

$$S(\text{lt}(h_i) g_i, \text{lt}(h_j) g_j),$$

since each $\text{lt}(f_i) g_i$ has multidegree δ while the sum has multidegree $< \delta$.

We have

$$S(\text{lt}(h_i) g_i, \text{lt}(h_j) g_j) = x^{-i,j} S(g_i, g_j) \quad (7.3)$$

where $x^{\gamma_{i,j}} = \text{lcm}(\text{lm}(g_i), \text{lm}(g_j))$ it follows that the first sum (Equation (7.3)) is a linear combination of $x^{\delta - \gamma_{i,j}} S(g_i, g_j)$ for appropriate pairs (i, j) .

Consider one of the $S(g_i, g_j)$. Since $\overline{S(g_i, g_j)}^G = 0$, we can rewrite it as

$$S(g_i, g_j) = \sum_{\ell=1}^t A_\ell g_\ell, \quad (7.4)$$

where $A \in K[x_1, \dots, x_n]$ and

$$\text{multideg}(A g) \leq \text{multideg}(S(g_i, g_j)) \quad (7.5)$$

where $A_\ell g_\ell \neq 0$. By multiplying both sides of Equation (7.4) by $x^{\delta - \gamma_{i,j}}$, we get

$$x^{\delta - \gamma_{i,j}} S(g_i, g_j) = \sum_{\ell=1}^t B_\ell g_\ell,$$

where $B_\ell = x^{\delta - \gamma_{i,j}} A_\ell$. Then it follows from Equation (7.5) that

$$\text{multideg}(B g) \leq \text{multideg}(S(g_i, g_j)) < \delta$$

since $\text{lt}(S(g_i, g_j)) < \text{lcm}(\text{lm}(g_i), \text{lm}(g_j)) = x^{i,j}$.

It follows that we can rewrite the first term in the sum as

$$\sum_{\text{multideg}(h_i g_i) < \delta} \text{lt}(h_i) g_i = \sum_{\ell=1}^t \tilde{B}_\ell g_\ell$$

with the property that when $\tilde{B}_\ell g_\ell \neq 0$, we have $\text{multideg}(\tilde{B}_\ell g_\ell) < \delta$.

we have obtained an expression for f as a polynomial combination of the g_i 's where every term has multidegree $< \delta$. This contradicts the assumption that δ was minimal and completes the proof. \square

The basic idea of Buchberger's algorithm is that polynomials are added to the input set until a Groebner basis is obtained – i.e., until Buchberger's criterion is satisfied.

There now follows a worked example to illustrate Buchberger's algorithm.

Consider $I = \langle f_1, f_2 \rangle = \langle x^3 - 2xy, x^2y - 2y^2 + x \rangle \subset Q[x, y]$ with grlex ordering. $\langle f_1, f_2 \rangle$ is not a Groebner basis for I . Indeed,

$$\begin{aligned} S(f_1, f_2) &= \frac{x^3y}{x^3} (x^3 - 2xy) - \frac{x^3y}{x^2y} (x^2y - 2y^2 + x) \\ &= (x^3y - 2xy^2) - (x^3y - 2xy^2 + x^2) \\ &= -x^2 \notin \{\text{lt}(f_1), \text{lt}(f_2)\} \end{aligned}$$

Compute

$$\overline{S(f_1, f_2)}^F.$$

We have

$$S(f_1, f_2) = -x^2 = g_1(x^3 - 2xy) + g_2(x^2y - 2y^2 + x) + h.$$

Setting $g_1 = -y, g_2 = x$ gives

$$\begin{aligned} &-y(x^3 - 2xy) + x(x^2y - 2y^2 + x) + h \\ &= -x^3y + 2xy^2 + x^3y - 2xy^2 + x^2 + h \\ &\Rightarrow h = -x^2 \end{aligned}$$

Add $-x^2$ to F . Since $-x^2 \in I$, we have $\overline{S(f_1, f_2)}^F = 0$.

We now need to consider $S(f_1, f_3)$ and $S(f_2, f_3)$.

$$\begin{aligned} S(f_1, f_3) &= \frac{x^3}{x^3} (x^3 - 2xy) - \frac{x^3}{x^2} x^2 = -2xy \\ S(f_2, f_3) &= \frac{x^2y}{x^2y} (x^2y - 2y^2 + x) - \frac{x^2y}{x^2} x^2 = -2y^2 + x \end{aligned}$$

The remainder of $\overline{S(f_1, f_3)}^F$ is $-2xy \in I$, so we add $-2xy$ to F . Similarly, the remainder $\overline{S(f_2, f_3)}^F$ is $-2y^2 + x \in I$, so we add $-2y^2 + x$ to F .

We have

$$F = (x^3 - 2xy, x^2y - 2y^2 + x, -x^2, -2xy, -2y^2 + x),$$

and $\overline{S(f_i, f_j)}^F = 0$ for all $f_i, f_j \in F$. We conclude that F is a Groebner basis for I .

We now present the algorithm.

Input: $F = \{f_1, \dots, f_k\} \subset K[x_1, \dots, x_n]$ – a finite set of multivariate polynomials

Output: A Groebner basis G for $I = \langle f_1, \dots, f_k \rangle$, s.t. $F \subset G$

- Let $G := F$
- do
 - let $G' := G$
 - for $f, g \in G$ s.t. $f \neq g$:
 - * let $r = \overline{S(f, g)}^{G'}$
 - * if $r \neq 0$ then $G := \{r\} \cup G$
- while $G \neq G'$
- return G

See (?, p91, Theorem 2) for a proof that this algorithm does indeed produce a Groebner basis for F and always terminates in a finite number of steps. Following Buchberger's algorithm, there were several improvements in efficiency, for example ... and Groebner basis remain a vital tool in computer algebra.

7.1.1.2 Zariski closures and saturations

Let $f_1, \dots, f_k \in K[x_1, \dots, x_n]$. We define

$$V(f_1, \dots, f_k) := \{x \in K^n \mid f_1(x), \dots, f_k(x) = 0\}.$$

If $I \subset K[x_1, \dots, x_n]$ is a polynomial ideal, we write

$$V(I) := \{x \in K^n \mid f(x) = 0 \text{ for all } f \in I\}.$$

Let $V \subset K^n$ be an affine algebraic variety. Then

$$I(V) := \{f \in K[x_1, \dots, x_n] \mid f(a) = 0 \text{ for all } a \in V\}.$$

To every variety belongs an ideal:

Lemma 7.8. *If $V \subset K^n$ is an affine variety, then $V(I) \subset K[x_1, \dots, x_n]$ is an ideal (called the ideal of V).*

Proof. We prove $I(V)$ satisfies the properties of an ideal.

- $0(x) = 0$ for all $x \in K^n$, hence $0(x) = 0$ for all $x \in V$, so $0 \in I(V)$.
- Let $x \in V$. If $f, g \in I(V)$ then by definition we have $f(x) = 0$ and $g(x) = 0$. Then $f(x) + g(x) = 0 + 0 = 0$, hence $f + g \in I(V)$.
- Let $x \in V$. If $f \in I(V)$ and $h \in K[x_1, \dots, x_n]$. By definition we have $f(x) = 0$. $h(x)f(x) = h(x) \cdot 0 = 0$, hence $hf \in I(V)$ for all $h \in K[x_1, \dots, x_n]$.

□

...and to every ideal belongs a variety.

Lemma 7.9. $V(I)$ is an affine variety. In particular if $I = \langle f_1, \dots, f_k \rangle$ then $V(I) = V(f_1, \dots, f_k)$.

Proof. By the Hilbert Basis Theorem I is finitely generated, therefore $I = \langle f_1, \dots, f_k \rangle$ for some finite set of generators in $K[x_1, \dots, x_n]$. We want to show that $V(I) = V(f_1, \dots, f_k)$.

Since $f_1, \dots, f_k \in I$ and $f(a) = 0$ for all $f \in I$, $V(I) \subset V(f_1, \dots, f_k)$. On the other hand, suppose $x \in V(f_1, \dots, f_k)$. Since $I = \langle f_1, \dots, f_k \rangle$, every $f \in I$ can be written

$$f = h_1 f_1 + \dots + h_k f_k.$$

Since $f_i(x) = 0$ for all $1 \leq i \leq k$, we have

$$f(x) = h_1(x) \cdot 0 + \dots + h_k(x) \cdot 0 = 0 + \dots + 0 = 0.$$

Therefore $V(f_1, \dots, f_k) \subset V(I)$, hence $V(I) = V(f_1, \dots, f_k)$. □

Let $S \subset K^n$ be an arbitrary subset, not necessarily an affine algebraic variety. $I(S)$ (defined in the same way as above) is an ideal and by the ideal-variety correspondence, $V(I(S))$ is an affine algebraic variety.

Lemma 7.10. Let $S \subset K^n$. $V(I(S))$ is the smallest affine algebraic variety containing S .

(I.e., for all affine algebraic varieties $W \supset S$, $V(I(S)) \subset W$.)

Proof. If $S \subset W$, then $I(W) \subset I(S)$ because the operation I is inclusion reversing. In addition, $V(I(S)) \subset V(I(W))$ because V is also inclusion reversing. Since W is an affine algebraic variety, $V(I(W)) = W$, so we have $V(I(S)) \subset W$. □

Definition 7.9. The Zariski closure of a subset $S \subset K^n$ is the smallest affine algebraic variety containing S . We denote the Zariski closure of S by \overline{S} .

The following property relating Zariski closures and ideals is relevant to (? , Proposition 5.14).

Lemma 7.11. (*?, p203, part of Theorem 10*) *Let $I, J \subset K[x_1, \dots, x_n]$ be ideals, then $\text{cl}(V(I) \setminus V(J)) \subset V(I : J^\infty)$,*

Proof. We claim that $I \subset I : J \subset I : J^2 \subset \dots \subset I : J^\infty \subset I(V(I) \setminus V(J))$. We have $I \subset I : J \subset I : J^2 \subset \dots \subset I : J^\infty$ by the ascending chain condition. Suppose that $f \in I : J^\infty$ and $a \in V(I) \setminus V(J)$. Then $fg^N \in I$ for all $g \in J$ and a suitable (large enough) $N \geq 0$. Since $a \in V(I)$, $f(a)g^N(a) = 0$ (for all $g \in J$ and large enough N). Since $a \notin V(J)$, there exists $f \in J$ and $N \geq 0$ such that $g(a) \neq 0$. Hence $f(a) = 0$ for all $a \in V(I) \setminus V(J)$. We have $f \in I(V(I) \setminus V(J))$. Transitivity of \subset proves the claim. Since V reverses inclusion, we have $\text{cl}(V(I) \setminus V(J)) = V(I(V(I) \setminus V(J))) \subset V(I : J^\infty)$. \square

Now we can return to (? , Proposition 5.14). By (? , Proposition 5.13), we may assume that $\text{fr}(C)$ is the closure of some one-dimensional cells and that blow-up points, if they exist, are 0-dimensional cylindrical cells in the induced decomposition of \mathbb{R}^2 . We begin with an irreducible polynomial $f \in \mathbb{Q}[x_1, x_2, x_3]$, such that there exists a $(1, 1, 0)$ -cell $C \subset \{\mathbf{x} \in \mathbb{R}^3 \mid f(\mathbf{x}) = 0\}$, a point $\mathbf{p} = (p_1, p_2) \in \mathbb{R}^2$ such that \mathbf{p} is a blow-up point of f and $g \in \mathbb{Q}[x_1, x_2]$ such that $g(\mathbf{p}) = 0$. Consider the variety $V(\langle f \rangle + \langle g \rangle) = V(f) \cap V(g)$, which contains some of the one-dimensional cells which form part of $\text{fr}(C)$ and the blow-up subset $B \subset \mathbf{p} \times \mathbb{R}$. The saturation by h of I ,

$$S := \langle f, g, 1 - zh \rangle \cap \mathbb{Q}[x_1, x_2, x_3]$$

is constructed. By (? , Theorem 10) (TODO),

$$\text{cl}(V(\langle f, g \rangle) \setminus V(\langle h \rangle)) \subset S.$$

Since our blow-up point \mathbf{p} is 0-dimensional, computing this Zariski closure effectively “fills in the hole” left by removing the blow-up subset from $V(\langle f, g \rangle)$. Thus, we have the limit points of the cell C as it approaches the blow-up subset B . In order to find the refinement points above \mathbf{p} , we compute a generating set (Groebner basis) $\{f_1, \dots, f_k\}$ for S and solve the system of equations

$$f_1 = 0, \dots, f_k = 0, g = 0, h = 0.$$

Note that this procedure only works if the blow-up points have dimension zero, otherwise the Zariski Closure will not be sufficient to find the limit points.

7.1.2 Algorithm for lifting with bad points in \mathbb{R}^3

Let D be a cylindrical algebraic decomposition of \mathbb{R}^n , $n \leq 3$. If $n < 3$, D already satisfies the frontier condition since there are no bad cells in the decomposition

induced by D on \mathbb{R}^1 . Otherwise $n = 3$ and we apply [?](#), Algorithm 5.15 to D to produce a set of refinement points. This algorithm, written using our notation, is reproduced below.

- Let D' be the decomposition induced by D on \mathbb{R}^2 .
- For each $(0,0)$ -cell (point) \mathbf{p} of D' , determine whether it is a bad cell. \mathbf{p} is a bad cell if there exists a polynomial $f \in A_3$ such that f has constant sign on a $(0,0,1)$ -cell C of D such that $\mathbf{p} = \text{proj}_{\mathbb{R}^2}(C)$.

(Note that this can easily be done by checking sings of projection factors for each $f \in A_3$ assigned to C).

- Let $g_1 \in A_1, g_2 \in A_2$ such that $g_1(\mathbf{p}) = 0$ and $g_2(\mathbf{p}) = 0$.
- Let $A' := A_1 \cup A_2$.
- For each $g \in A'$, do
 - If $g \in \mathbb{Q}[x_1]$, then $h := g_2$. Otherwise, $g \in \mathbb{Q}[x_1, x_2]$ and $h := g_1$.
 - Define the saturation

$$I := \langle f, g, 1 - zh \rangle,$$

where z is a new variable.

- Compute a generator system S , i.e., a Groebner basis, S for the ideal I .
- Compute

$$L := \{\text{sub}_{\mathbf{p}}(f') \in \mathbb{Q}[x_3] \mid f' \in S\}$$

and compute the roots

$$(c_1, \dots, c_t)$$

of L .

- Let

$$R_{\mathbf{p}} := (c_1, \dots, c_t).$$

- return R .

Compute the refinement of sections above bad cells using the same method described in Section 6.4.

7.2 Generalisation of Lazard

We can now generalise this result to cells of dimension ≤ 2 in a CAD of arbitrary dimension. Our goal is to construct a CAD of \mathbb{R}^n , compatible with $V := V_1 \cup \dots \cup V_k$ such that $\dim(V_i) \leq 2$ for all $1 \leq i \leq k$, such that each cell is monotone with respect to V and and each cell $C \subset V$ satisfies the frontier condition. In Section 5, we constructed a decomposition such that each cell has

constant sign on every polynomial defining V , with the result that each cell in the decomposition is smooth. In Section 6, we refined that decomposition so that each cell $C \subset V$ is monotone. In order to extend ?, Algorithm 5.14, we will require this monotonicity property. We will also require an additional property.

Lemma 7.12. (*?, Proposition 5,2*) *Let D be a CAD of \mathbb{R}^n and D' the decomposition induced by D on \mathbb{R}^{n-1} . Let C be a cell of D and $C' := \text{proj}_{\mathbb{R}^{n-1}}(C)$ such that C' is well-bordered. If $p \in \text{fr}(C')$, then $I := (p \times \mathbb{R}) \cap \text{fr}(C)$ is connected. I is thus a point or a closed segment, either bounded or unbounded. Furthermore, if C is a sector cell, then the set of points of I which do not belong to the boundaries of the section cells delimiting C is either empty or an open interval.*

By @7.2, if no cell adjacent to C is bad and the induced decomposition is strong, then C is well-bordered, topologically regular and satisfies the frontier condition. Thus, we need to look at what happens in the neighbourhood of blow-up points only.

Lemma 7.13. *Let D be a sign-invariant CAD of \mathbb{R}^n , C be a cell of D and $C' := \text{proj}_{\mathbb{R}^{n-1}}(C)$ satisfy the frontier condition. If no cell in $\text{fr}(C')$ is bad, then C satisfies the frontier condition.*

Proof is either by Lazard or Davenport, Locatelli and Sankaran.

If there is a blow-up point, then the following situation will occur.

Lemma 7.14. *Let \mathcal{D} be a sign-invariant decomposition of \mathbb{R}^n compatible with a semialgebraic set $V \subset \mathbb{R}^n$ having dimension at most 2, and suppose that $C \subset V$ is a topologically regular cell in \mathcal{D} . Suppose that D is a cell of \mathcal{D} such that $B := D \cap \text{fr}(C) \neq \emptyset$ and $B \neq D$. Then either*

1. *C is a 1-dimensional section cell whose endpoint, B , projects on a bad cell, or*
2. *there exists a cell E of \mathcal{D} such that $\dim(E) < \dim(C)$ and $\emptyset \neq D \cap \text{fr}(E) \neq D$.*

By the proof of (?, Theorem 4.4), D will always be a $(0, \dots, 0, 1)$ -cell in the cylinder above a bad cell D' .

Since $\dim(V) \leq 2$, C has dimension at most 2. We consider each case by cell index.

- $(i_1, \dots, i_{n-1}, 1)$ -cell, where $\sum_{j=1}^{n-1} i_j = 1$: observe that C (if bounded) is delineated by some $(i_1, \dots, i_{n-1}, 0)$ -cells (one-dimensional curve intervals). Take B to be the one-dimensional interval delineated by the endpoints of these curves. Since $B \neq C$, at least one of these endpoints is not a $(0, \dots, 0, 0)$ -cell of D . Case 2 applies, taking as E the delineating $(i_1, \dots, i_{n-1}, 0)$ -cell.

- $(i_1, \dots, i_{n-1}, 0)$ -cells, where $\sum_{j=1}^{n-1} i_j = 1$: consider the previous case, letting $C = E$.
- $(i_1, \dots, i_{n-1}, 0)$ -cell, where $\sum_{j=1}^{n-1} i_j = 2$: are necessarily the root of a polynomial $f \in \mathbb{Q}[x_1, \dots, x_n]$ which vanishes identically on D' , a $(0, \dots, 0)$ cell in the decomposition induced by D on \mathbb{R}^{n-1} . Let V_r be a small neighbourhood (with radius $r > 0$) centered around D' and consider V'_r , the cylinder above the boundary of V_r . Since C is a monotone cell, the maximum and minimum values of x_n in this intersection are obtained on the boundary of C , for sufficiently small r . The limits, as r tends to 0, of this maximum and minimum clearly belong to the boundary of B , if these limits do not tend to infinity. Note that both the maximum and minimum cannot be infinite, since this would imply that either B is empty, or coincides with D . Therefore, the non-infinite limits are the endpoints of some 1-dimensional cell belonging to the boundary of C . Case 2 applies, taking E to be this 1-dimensional cell.
- $(0, \dots, 0, 1)$ - or $(0, \dots, 0, 0)$ -cells: do not satisfy the hypothesis, there is nothing to do.

The previous two results mean that we can extend ? Theorem 5.13 to 2-dimensional cells in sign-invariant decompositions of \mathbb{R}^n .

Let $f \in \mathbb{Q}[x_1, \dots, x_n]$ be an irreducible polynomial given by

$$f = g_d x^d + \dots + g_1 x + g_0$$

with each $g_i \in \mathbb{Q}[x_1, \dots, x_{n-1}]$, and

$$\mathbf{p} = (p_1, \dots, p_{n-1})$$

be a common root of g_d, \dots, g_1, g_0 . Let $g_j \in \mathbb{Q}[x_1, \dots, x_j], j \in \{1, \dots, n-1\}$ ℓ , where $1 \leq \ell \leq n-1$, be irreducible polynomials having \mathbf{p} as a root. The limits of the common roots of f and the g_i 's as $(x_1, x_2) \rightarrow \mathbf{p}, (x_1, x_2) \neq \mathbf{p}$ may be computed as solutions of a zero-dimensional polynomial system.

Proof. This is wrong!

Since f is irreducible, it has at least two coefficients, $g_i, g_j \in \mathbb{Q}[x_1, \dots, x_{n-1}]$, whose GCD is a constant. Therefore, any ideal generated by these coefficients has dimension zero. Any ideal containing these coefficients also contains another element, h , which is not a multiple of g .

The saturation by h of the ideal $\langle f, g \rangle$ is defined as the ideal

$$S := \langle f, g, 1 - zh \rangle \cap \mathbb{Q}[x_1, x_2, x_3].$$

The irreducible components of the zero set of S are those of the zero set of $\langle f, g \rangle$ at which h is not identically zero. It follows that the vertical line $\{x_1 = p_1, x_2 = p_2\}$ which is part of the zero set of f, g and h , is not contained in the zero set

of S . Meanwhile, the zero set of S contains the limit points we are interested in finding.

It follows that the ideal generated by S and the coefficients g_0, \dots, g_d of f is zero-dimensional and the limits of $\{f = 0, g = 0\}$ as $(x_1, x_2) \rightarrow \mathbf{p}$ can be computed as the zeros of S . \square

Chapter 8

A novel algorithm for the obtaining the frontier condition on a cylindrical decomposition of \mathbb{R}^n

So far, we have been concerned only with cylindrical algebraic decomposition compatible with subsets of \mathbb{R}^n having dimension at most two. We will now present a novel algorithm for obtaining the frontier condition on a cylindrical algebraic decomposition of \mathbb{R}^n compatible with a set of arbitrary dimension.

Theorem 8.1. *Let $S \subset \mathbb{R}^n$ be a semialgebraic set defined by a quantifier-free Boolean formula F with s different polynomials of maximum degree d in $\mathbb{R}[x_1, \dots, x_n]$. There is an algorithm, taking F as input, which outputs a cylindrical decomposition \mathcal{D} of \mathbb{R}^n compatible with S and satisfying the frontier condition. The complexity of this algorithm is $(sd)^{O(1)^{n2^n}}$. This is also an upper bound on the number of cells in \mathcal{D} , number of polynomials defining cells and their degrees.*

Remark. Algorithm in Theorem 8.1 is understood as a Blum-Shub-Smale (BSS) real numbers machine ?. A similar statement is also true for the Turing machine model, in which case the complexity bound depends, in addition, polynomially on maximal bit-size of coefficients (cf. ?).

8.1 Subroutines

We first present some subroutines necessary to obtain a CAD with frontier condition.

8.1.1 The “Classical” CAD

We have been working with cylindrical algebraic decomposition throughout this work. Here we present a slightly improved version of the algorithm due to ?.

Proposition 8.1. (*?, Algorithm 11.2*) *Let $\mathbf{F} := f_1, \dots, f_s$ be polynomials in $\mathbb{R}[x_1, \dots, x_n]$ having maximum degree d . There is an algorithm, taking $\{f_1, \dots, f_s\}$ as input, which produces a cylindrical decomposition \mathcal{E} of \mathbb{R}^n such that every cell in \mathcal{E} has constant sign on every polynomial f_1, \dots, f_s . The complexity of the algorithm is $(sd)^{O(1)^n}$. This is also an upper bound on the number of cells in \mathcal{E} , number of polynomials defining cells and their degrees.*

This is the so-called **F**-invariant CAD. We also present two variations of the algorithm for constructing a “coarser” CAD, i.e., one containing fewer cells.

Proposition 8.2. *Let S_1, \dots, S_k be a finite collection of semialgebraic subsets of \mathbb{R}^n defined by quantifier-free Boolean formulas F_1, \dots, F_k respectively. Together, these formulas include s different polynomials in $\mathbb{R}[x_1, \dots, x_n]$, having maximum degree d . There is an algorithm, taking $\{F_1, \dots, F_k\}$ as input, which produces a cylindrical decomposition \mathcal{E} of \mathbb{R}^n compatible with each $S_i, 1 \leq i \leq k$. Complexity, number of cells, number of polynomials and degrees are the same as in Proposition 8.1.*

Proof. Let $f_1, \dots, f_s \in \mathbb{R}[x_1, \dots, x_n]$ be the polynomials from formulas F_1, \dots, F_k and let \mathcal{E}' be the CAD of \mathbb{R}^n produced by Proposition @ref{prp:novel-frontier-collins} with f_1, \dots, f_s as input. Since F_i for each $1 \leq i \leq k$ provides a set of sign conditions on polynomials f_1, \dots, f_s , \mathcal{E}' is a refinement of the CAD produced by this algorithm.

Truth values of F_1, \dots, F_k on each cell can easily be computed from the CAD produced by Proposition 8.1 by considering the signs of f_1, \dots, f_s on each cell, which we computed in the construction of \mathcal{E}' . Observe that this does not change the asymptotic complexity or bounds in Propoosition ??.

The construction described in the above proof is very naive. This algorithm, in a more tractable form, is presented in, e.g., ?, and an alternative approach is presented in ?.

Proposition 8.3. *Let $S \subset \mathbb{R}^n$ be a semialgebraic set defined by a quantifier-free Boolean formula F containing s different polynomials in $\mathbb{R}[x_1, \dots, x_n]$, having maximum degree d . There is an algorithm, taking F as input, which produces a cylindrical decomposition \mathcal{E} of \mathbb{R}^n compatible with S . Complexity, number of cells, number of polynomials and degrees are the same as in Proposition 8.1.*

Proof. Observe that this is the same as the algorithm as in Proposition ?? taking $\{F\}$ as input.

8.1.2 Computing the Frontier

Now we describe an algorithm, having singly-exponential complexity, for finding the frontier, $\text{fr}(S) = \text{cl}((S) - S)$, of a semialgebraic set $S \subset \mathbb{R}^n$.

Lemma 8.1. *Let $S \subset \mathbb{R}^n$ be a semialgebraic set defined by a quantifier-free Boolean formula F containing s different polynomials in $\mathbb{R}[x_1, \dots, x_n]$ having maximum degree d . There is an algorithm, taking F as input, which represents the semialgebraic set $\text{fr}((S)$ by a quantifier-free Boolean formula F' with complexity $(sd)^{O(n^2)}$. This is also an upper bound on the number of polynomials in F' and their degrees.*

∴ Observe that $\text{fr}(S)$ can be represented by a first-order Boolean formula

$$\text{fr}(S) = \{\mathbf{x} \in (\mathbb{R}^n - S) \mid \forall \varepsilon > 0 \exists \mathbf{y} \in S (\|\mathbf{x} - \mathbf{y}\| < \varepsilon)\}.$$

Using singly-exponential quantifier elimination algorithm ?, Algorithm 14.21, we represent $\text{fr}((S)$ as a quantifier-free Boolean formula F' with the bounds required in the lemma. ∴

8.2 Obtaining the frontier condition

Recall that, according to Definition 3.17, to each cylindrical cell in a CAD of \mathbb{R}^n a (multi-)index $(i_1, \dots, i_n) \in \{0, 1\}^n$ is assigned. Introduce a lexicographic order $<_{\text{lex}}$ on the set of all cell indices as follows. For any two indices, $M := (i_1, \dots, i_n)$ and $N := (j_1, \dots, j_n)$, we set $M <_{\text{lex}} N$ iff for the maximal k , $0 \leq k < n$, such that $i_1 = j_1, \dots, i_k = j_k$ we have $i_{k+1} < j_{k+1}$. If $M <_{\text{lex}} N$ or $M = N$, we write $M \leq_{\text{lex}} N$.

Lemma 8.2. *Let \mathcal{D} be a CAD of \mathbb{R}^n and let C be a cell of \mathcal{D} with index M . Then $\text{fr}(C)$ is contained in a union of cells of \mathcal{D} with indices lexicographically less than M .*

Proof. Proceed by induction on n .

When $n = 1$, the cell C either has index $M = (0)$ or $M = (1)$. If $M = (0)$ then C is a single point and its frontier is empty. If $M = (1)$, C is an open interval. By the definition of CAD, endpoints of C are (0)-cells.

Now let $n > 1$ and let C have index $M = (i_1, \dots, i_n)$. The projection $C' = \text{proj}_{\mathbb{R}^{n-1}}(C)$ is a cell in the induced decomposition \mathcal{D}' of \mathbb{R}^{n-1} with index (i_1, \dots, i_{n-1}) . By the induction hypothesis, $\text{fr}(C')$ is contained in a union of cells of \mathcal{D}' with indices $(j_1, \dots, j_{n-1}) <_{\text{lex}} (i_1, \dots, i_{n-1})$.

If $i_n = 0$, then C is a section cell. Its frontier $\text{fr}((C)$ is contained in $\text{fr}((C') \times \mathbb{R})$ and, therefore, in a union of cells of \mathcal{D} with indices $(j_1, \dots, j_{n-1}, j_n)$ for some $j_n \in \{0, 1\}$. We have shown that $(j_1, \dots, j_{n-1}, j_n) <_{\text{lex}} M$.

If $i_n = 1$, then C is a sector cell. In this case, $\text{fr}((C))$ is contained in $(\text{fr}((C')) \times \mathbb{R}) \cup C_T \cup C_B$, hence in the union of cells of \mathcal{D} with indices $(j_1, \dots, j_{n-1}, j_n)$ for some j_n and two cells, C_T and C_B , having the same index $(i_1, \dots, i_{n-1}, 0)$. All of these indices are lexicographically less than M . \square

In further proofs we will need the following technical statement.

Lemma 8.3. *Let $f : X \rightarrow \mathbb{R}$ be a continuous polynomial function on an open set $X \subset \mathbb{R}^n$ having the graph $F \subset \mathbb{R}^{n+1}$. Let G be a semialgebraic subset of F , which is closed in F , and having dimension less than $\dim(F) = n$. Then $\text{fr}_F(F \setminus G) = G$, where fr_F denotes the frontier in F .*

Proof. Let $Y = \text{proj}_{\mathbb{R}^{n-1}}(G)$ and observe that $X = \text{proj}_{\mathbb{R}^{n-1}}(F)$. Let us first prove that $\text{fr}_X(X \setminus Y) = Y$. By the definition of frontier, we have

$$\text{fr}_X(X \setminus Y) = \text{cl}_X(X \setminus Y) \setminus (X \setminus Y),$$

where cl_X denotes the closure in X . $\text{cl}_X(X \setminus Y) = X$. Indeed, $X \setminus Y$ is open and n -dimensional since Y is closed in X and $\dim(Y) < n$. Assume that $\text{cl}_X(X \setminus Y) \neq X$. Then there is a nonempty subset $Z \subset X$ such that $Z \not\subset \text{cl}_X(X \setminus Y)$. Since $X \setminus Y \subset \text{cl}_X(X \setminus Y)$, Z is necessarily a subset of Y and Z has dimension $< n$. This implies that $\text{cl}(X) \setminus X \setminus Y$ is not, which is a contradiction. It follows that $\text{fr}_X(X \setminus Y) = X \setminus (X \setminus Y) = X \cap Y = Y$.

Finally, since f is a continuous function, and therefore Y and X are homeomorphic to G and F respectively, we obtain the required formula $\text{fr}_F(F \setminus G) = G$. \square

We now prove that the decomposition induced on \mathbb{R}^k , $1 \leq k \leq n-1$ by any CAD satisfying the frontier condition also satisfies the frontier condition.

Theorem 8.2. *Let \mathcal{D} be a CAD of \mathbb{R}^n compatible with a semialgebraic set $S \subset \mathbb{R}^n$ and satisfying the frontier condition. Then the decomposition induced by \mathcal{D} on \mathbb{R}^k for all $1 \leq k < n$ is a cylindrical decomposition of \mathbb{R}^k compatible with $\text{proj}_{\mathbb{R}^k}(S)$ and satisfying the frontier condition.*

Proof. Let \mathcal{D} be a CAD compatible with S and satisfying the frontier condition. By the definition of CAD, $\text{proj}_{\mathbb{R}^{k_0}}(S)$ is a union of cells in the decomposition induced by \mathcal{D} on \mathbb{R}^k for all $1 \leq k \leq n-1$.

We now prove that if \mathcal{D} satisfies the frontier condition then all decompositions induced by \mathcal{D} on \mathbb{R}^k for all $1 \leq k \leq n-1$ satisfy the frontier condition. Consider Γ , the union of cells of \mathcal{D} contained in the k -dimensional cylinder $\text{proj}_{\mathbb{R}^{n-1}}(C) \times \mathbb{R}$ where C is a cell of \mathcal{D} . Observe that $\bigcup_{C \in \Gamma} C = \text{proj}_{\mathbb{R}^{n-1}}(C) \times \mathbb{R}$. Consider

$$\text{fr}(\Gamma) := \bigcup_{C \in \Gamma} \text{fr}(C) \setminus \bigcup_{C \in \Gamma} C.$$

Observe that $\text{fr}(\Gamma)$ is a union of cells of \mathcal{D} since every cell C in Γ satisfies the frontier condition. % since $\text{proj}_{\mathbb{R}^{n-1}}(C) \times \mathbb{R}$ is the union of cells of \mathcal{D} contained in Γ and every cell C in Γ satisfies the frontier condition. We claim that $\text{fr}(\Gamma)$ is the union of some similarly defined cylinders of dimension less than k . Proceed by induction on k .

The base case, $k = 1$, is simple since $\text{fr}(\Gamma) = \emptyset$.

Now suppose that $k > 1$. Each sector cell in Γ has dimension k and each section cell in Γ has dimension $k - 1$, so the dimension of the union of frontiers of all cells in Γ is at most $k - 1$. Hence $\dim(\text{fr}(\Gamma))$ is at most $k - 1$. By the cylindrical property of CAD, the projection of any two cells contained in $\text{fr}(\Gamma)$ is either disjoint or coincides. Therefore, $\text{fr}(\Gamma)$ is a union of cylinders $\Gamma' := \text{proj}_{\mathbb{R}^{n-1}}(B) \times \mathbb{R}$ where B is a cell of \mathcal{D} contained in $\text{fr}(\Gamma)$. Observe that each cylinder Γ' has dimension less than k . By the induction hypothesis, $\text{fr}(\Gamma')$ is a union of cylinders with dimension less than $\dim(\Gamma')$.

By the definition of CAD, each of these cylinders Γ' projects onto a single cell in the decomposition \mathcal{D}' induced by \mathcal{D} on \mathbb{R}^{n-1} . Hence $\text{fr}(C')$, where $C' := \text{proj}_{\mathbb{R}^{n-1}}(C)$ for all $C \in \Gamma$ is a union of cells $B' := \text{proj}_{\mathbb{R}^{n-1}}(B)$ of \mathcal{D}' for all $B \in \Gamma'$. It follows that \mathcal{D}' satisfies the frontier condition.

The proof that every decomposition induced by \mathcal{D} on \mathbb{R}^k , $1 \leq k \leq n - 1$ satisfies the frontier condition is completed by iterating the projection operation. \square

8.2.1 Proof of Theorem 8.1

We are now ready to prove the main result. The worked example, presented in Section ?? is referenced to help illustrate the procedure. We begin with a mathematical description of the algorithm.

8.2.1.1 Algorithm

Input: F , a quantifier-free Boolean formula defining a semialgebraic set $S \subset \mathbb{R}^n$

Output: \mathcal{D} , a CAD of \mathbb{R}^n compatible with S and satisfying the frontier condition\

First, apply algorithm from Proposition 8.3 to F . We obtain a CAD \mathcal{E} of \mathbb{R}^n compatible with S .

The algorithm computes a sequence of pairs (\mathcal{E}_M, X_M) where $M \in \{0, 1\}^{n-1}$ is an index, \mathcal{E}_M is a CAD of \mathbb{R}^n , which is a refinement of \mathcal{E} , and $X_M \subset \mathbb{R}^n$ is a semialgebraic set contained in the union of all cells of \mathcal{E}_M with indices $\leq_{\text{lex}} (i_1, \dots, i_{n-1}, 1)$. This family of cells will be denoted by \mathcal{U}_M . Note that index M refers to the pair rather than its elements and \mathcal{E}_M and X_M are written with indices for convenience.

The sequence of pairs is computed recursively, starting with index $M = (1, 1, \dots, 1)$, in descending order of indices $M = (i_1, \dots, i_{n-1})$ with