Lazard-style CAD and Equational Constraints

James H. Davenport
University of Bath
United Kingdom
masjhd@bath.ac.uk

Akshar Nair
University of Bath
United Kingdom
akshar.nair@gmail.com

Gregory Sankaran
University of Bath
United Kingdom
g.k.sankaran@bath.ac.uk

Ali K. Uncu
RICAM, Austrian Academy of Sciences
& University of Bath
Austria & United Kingdom
akuncu@ricam.oeaw.ac.at

ABSTRACT

McCallum-style Cylindrical Algebra Decomposition (CAD) is a major improvement on the original Collins version, and has had many subsequent advances, notably for total or partial equational constraints. But it suffers from a problem with nullification. The recently-justified Lazard-style CAD does not have this problem. However, transporting the equational constraints work to Lazard-style does reintroduce nullification issues. This paper explains the problem, and the solutions to it, based on the second author's Ph.D. thesis and the Brown–McCallum improvement to Lazard.

With a single equational constraint, we can gain the same improvements in Lazard-style as in McCallum-style CAD . Moreover, our approach does not fail where McCallum would due to nullification. Unsurprisingly, it does not achieve the same level of improvement as it does in the non-nullified cases. We also consider the case of multiple equational constraints.

CCS CONCEPTS

• **Computing methodologies** → **Symbolic and algebraic manipulation**.

KEYWORDS

Cylindrical algebraic decomposition, Lazard projection and lifting, Equational constraints

ACM Reference Format:
James H. Davenport, Akshar Nair, Gregory Sankaran, and Ali K. Uncu. 2023. Lazard-style CAD and Equational Constraints. In *ISSAC'23: International Symposium on Symbolic and Algebraic Computation, Dates, City, Country*. ACM, New York, NY, USA, 9 pages. https://doi.org/DOIGoesHere

We acknowledge UKRI EPSRC for their constant support. The second author's thesis was supported by EPSRC grant EP/N509589/1. The first and the last authors are partially supported by EPSRC grant EP/T015713/1. The last author also thanks the partial support of Austrian Science Fund FWF grant P34501-N. We are grateful to Chris Brown for explanations of [4].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ISSAC'23, Dates, City, Country
© 2023 Association for Computing Machinery.
ACM ISBN ISBN Goes Here…$15.00
https://doi.org/DOIGoesHere

1 INTRODUCTION

Cylindrical Algebraic Decomposition (CAD) was introduced by Collins in 1975 [6]. There have been many improvements since [6] introduced the projection-lifting paradigm. McCallum [14] made a major one, but this had the drawback that it could not be applied if a polynomial nullified. There have been many developments to [14], often concerned with *equational constraints*, i.e. when the semi-algebraic set lies in a proper sub-variety. [18] justified Lazard's idea in [13], using different forms of projection and lifting to avoid the nullification problem.

This paper is a first step in transplanting the improvements to [14] to the Lazard setting. More specifically, we base ourselves on the Brown–McCallum [4] version of the Lazard approach.

Previous papers [20, 21] were based on [18]. The second author's thesis [19] was largely based on [18], with some updates for [4]. Here we integrate [4] throughout.

2 BACKGROUND AND NOTATION

2.1 CAD by Projection-Lifting

There are various ways of computing cylindrical algebraic decompositions, e.g. via Triangular Decomposition [5], but this paper is cast in the context of the Projection-Lifting paradigm introduced by Collins [6]: see Figure 1. Let $S \subset \mathbb{Z}[x_1, \ldots, x_n]$ be the set of polynomials of interest, and $\phi(S)$ the property we wish our decomposition to have. Generally $\phi(S)$ will be a variant of "each cell is sign-invariant for the polynomials in S", as it is that property that enables quantifier elimination, Collins' original motivation.

$$
\begin{array}{llll}
\mathbb{R}^n & S & & D_n \quad \phi_X(S) \\
 & \downarrow P_X & & \uparrow L_X \\
\mathbb{R}^{n-1} & P_X(S) & & D_{n-1} \quad \phi_X(P_X(S)) \\
 & \downarrow P_X & & \uparrow L_X \\
\vdots & \vdots & & \vdots \quad \vdots \\
\mathbb{R}^2 & P_X^{n-2}(S) & & D_2 \quad \phi_X(P_X^{n-2}(S)) \\
 & \downarrow P_X & & \uparrow L_X \\
\mathbb{R}^1 & P_X^{n-1}(S) & \rightarrow \text{isolate roots} \rightarrow & D_1 \quad \phi_X(P_X^{n-1}(S))
\end{array}
$$

where X indicates the precise variant of Projection/Lifting, and D_i is the intermediate decomposition of $\mathbb{R}^i$.

Figure 1: Projection/Lifting Paradigm

The original Collins algorithm for Cylindrical Algebraic Decomposition [6] have an expensive (compared to its successors) projection operator P_C, and ϕ_C is indeed sign-invariance. The lifting operation L_C is simple: above each cell $C \in D_i$ there is a cylinder $C \times \mathbb{R} \subset \mathbb{R}^{i+1}$ which is sliced by the real branches of $p_k = 0$ for the polynomials $p_k \in P^{n-i-1}(S)$ and P_C ensures that these branches are delineable over C, i.e. do not intersect and are simple surfaces above the whole of C. The relevant cells of D_{i+1} are then each branch, known as "sections", and the "sectors" between adjacent branches, including the sector "$-\infty < x_{i+1} <$ all sections" and the sector "all sections $< x_{i+1} < \infty$".

2.2 Curtains

Most improvements to Collins' algorithm can encounter difficulties when some polynomials nullify, i.e. vanish over some subset of $\mathbb{R}^k = \langle x_1, \ldots, x_k \rangle$. The term curtain for the varieties where polynomials nullify was introduced in [20]:

Definition 1 ([19, Definition 43]). *A variety $C \subseteq \mathbb{R}^n$ is called a curtain if, whenever $(x, x_n) \in C$, then $(x, y) \in C$ for all $y \in \mathbb{R}$.*

C is a curtain if it is a union of fibres of $\mathbb{R}^n \to \mathbb{R}^{n-1}$.

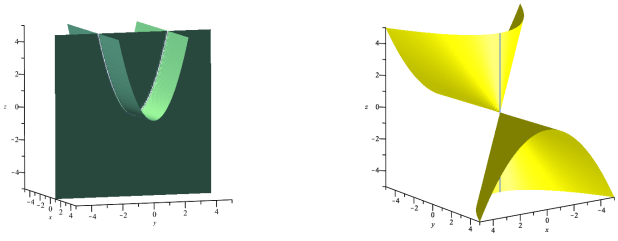
Definition 2 ([19, Definition 44]). *Suppose $f \in \mathbb{R}[x_1, \ldots, x_n]$ and $S \subseteq \mathbb{R}^{n-1}$. We say that V_f (or f) has a curtain at S if for all $(\alpha_1, \ldots, \alpha_{n-1}) \in S$ and for all $y \in \mathbb{R}$ we have $f(\alpha_1, \ldots, \alpha_{n-1}, y) = 0$. We call S the* foot *of the curtain.*

If the foot of a curtain, S, is a singleton, we call it a point-curtain.

Definition 3 (cf [19, Definition 45]). *Suppose the polynomial $f \in \mathbb{R}[x_1, \ldots, x_n]$ factorises as $f = gh$, where $g \in \mathbb{R}[x_1, \ldots, x_{n-1}]$ and $g(\alpha_1, \ldots, \alpha_{n-1}) = 0$. Then the variety of f is said to contain an explicit curtain whose foot is the zero set of g. A curtain which does not contain (set-theoretically) an explicit curtain is said to be implicit, and a curtain which contains an explicit curtain but is not an explicit curtain (because h itself has curtains) is said to be mixed.*

Example 1 (Types of Curtain).

- *Explicit Curtain: $f(x, y, z) = xy^2 - y^2 - xz + z = (x-1)(y^2 - z)$, curtain at $(1, 0)$. The curtain can be seen in Figure 2a as the sheet given by $x = 1$.*
- *Implicit Curtain: $f(x, y, z) = x^2 + yz$, curtain at $(0, y, 0)$. The blue line represents this curtain in Figure 2b.*



(a) Surface with an Explicit Curtain **(b) Surface with an Implicit Curtain**

Figure 2: Different types of curtains

2.3 McCallum's Improvement to Collins

Definition 4 ([14, p. 46]). *Let A be a set of polynomials in $\mathbb{Z}[x_1, \ldots, x_n]$, $\mathrm{cont}(A)$ the contents of the elements of A with respect to x_n, and B be a square-free basis[1] for A. Let $\mathrm{coeff}(B)$ be the set of all coefficients of the members of B (viewed as polynomials in x_n), $\mathrm{disc}(B)$ be the set of discriminants of elements of B (with respect to x_n) and $\mathrm{res}(B)$ the set of all resultants (with respect to x_n) of pairs of elements of B. McCallum's projection operator P_M is defined as*

$$P_M(A) := \mathrm{cont}(A) \cup \mathrm{coeff}(B) \cup \mathrm{disc}(B) \cup \mathrm{res}(B). \tag{1}$$

Using this definition, McCallum [14] improves on the Collins' original [6]. The projection P_M is distinctly smaller than P_C, and the corresponding ϕ_M is order-invariance, i.e. if a polynomial vanishes on a cell $C \in D_i$ it does so to constant order on that cell. The corresponding lifting operator L_M is identical to L_C, except that, if a polynomial q vanishes on a zero-dimensional cell, it is replaced by the corresponding *delineating polynomial* [15, Definition 1]. However, L_M may fail (giving an error message, not the wrong result) if relevant polynomials nullify (are identically zero) over a cell of $\mathbb{R}^k$ of positive dimension.

2.4 Brown's Improvement to McCallum

Brown [3] introduced a useful improvement on [14].

Definition 5. *We use the same notation as Definition 4, and* lc *for the leading coefficient. Brown's projection is defined as*

$$P_B(A) := \mathrm{cont}(A) \cup \mathrm{lc}(B) \cup \mathrm{disc}(B) \cup \mathrm{res}(B). \tag{2}$$

The corresponding lifting L_B has several (necessary) enhancements over L_M but the details are largely irrelevant here: the relevant one is that, to the CAD we are building over, we add those points above which some projection factor nullifies (if it nullifies on a set of positive dimension, then the system is not well-oriented). For any non-trivial set A of polynomials, $P_B(A) \subsetneq P_M(A)$, the difference being the non-leading coefficients of B. This does not affect the O-asymptotics, which are driven by the $\mathrm{res}(B)$ term, but is very effective in practice. [3, Tables 1,2] consider an example A in six variables, where $|P_B^4(A)| = 18$, while $|P_M^4(A)| = 129$. $|P_B^5(A)| = 141$, while $P_M^5(A)$ could not be computed, but probably $|P_M^5(A)| > 8000$.

2.5 Lex-least valuation and Lazard lifting

Lazard [13] introduced a novel approach to CAD to bypass nullification, which was later made rigorous by [18]. This section contains the relevant definitions, as used in [19, 22].

Definition 6. *Let $v, w \in \mathbb{Z}^n$. We say that $v = (v_1, \ldots, v_n) \geq_{lex} (w_1, \ldots, w_n) = w$ if and only if either $v = w$ or there exists an $i \leq n$ such that $v_i > w_i$ and $v_k = w_k$ for all k in the range $1 \leq k < i$.*

Definition 7 ([18, Definition 2.4]). *Let $n \geq 1$ and suppose that $f \in \mathbb{R}[x_1, \ldots, x_n]$ is non-zero and $\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^n$. The lex-least valuation $v_\alpha(f)$ at α is the least (with respect to $\geq_{lex}$) element $v = (v_1, \ldots, v_n) \in \mathbb{N}^n$ such that f expanded about α has the term*

$$c(x_1 - \alpha_1)^{v_1} \cdots (x_n - \alpha_n)^{v_n},$$

where $c \neq 0$.

[1] [14] uses the finest square-free basis, but this is for reasons of efficiency.

Note that $v_\alpha(f) = (0, \ldots, 0)$ if and only if $f(\alpha) \neq 0$. We should note that the terminology we are using is slightly different than the previous ones. For example, Lex-least valuation is referred to as the Lazard valuation in [18]. We do it in an effort to be precise.

Lex-least valuation of f at $\alpha \in \mathbb{R}^n$ is calculated by first picking an order of variables $x_1 \prec x_2 \prec \cdots \prec x_n$, then successively substituting in α_i in the given variable order. (The order of projections start with x_n first, the order of valuation calculations starst with x_1 first.) At each step of the successive substitutions $f(\alpha_1, \ldots, \alpha_{i-1}, x_i, \ldots, x_n)$ we identify the largest positive integer v_i such that

$$(x_i - \alpha_i)^{v_i} \mid f(\alpha_1, \ldots, \alpha_{i-1}, x_i, \ldots, x_n).$$

The v_i is the component of the Lex-least valuation of f at α that corresponds to α_i. We continue the valuation calculations by first replacing $f(\alpha_1, \ldots, \alpha_{i-1}, x_i, \ldots, x_n)$ with $f(\alpha_1, \ldots, \alpha_{i-1}, x_i, \ldots, x_n) \times (x - \alpha_i)^{-v_i}$ and then substituting α_i in this new function.

Example 2. *If $n = 1$ and $f(x_1) = x_1^3 - 2x_1^2 + x_1 = x_1(x_1 - 1)^2$, then $v_0(f) = 1$, $v_1(f) = 2$, and $v_x(f) = 0$, for all $x \neq 0, 1$. Similarly, if $n = 2$ (with the standard order $x_1 \prec x_2$) and $f(x_1, x_2) = x_1(x_2 - 1)^2$, then $v_{(0,0)}(f) = (1, 0)$, $v_{(2,1)}(f) = (0, 2)$, $v_{(0,1)}(f) = (1, 2)$, and $(0, 0)$ otherwise.*

The lex-least valuation is strongly dependent on the order of the variables, as the following example illustrates.

Example 3. *Let $f(x, y, z, w) = x^2 + y^2 z - 2yz^2 + zw$, $\alpha_1 = (0, 1, 0, 1)$ and $\alpha_2 = (0, 0, 1, 0)$. With respect to the ordering $x \prec y \prec z \prec w$ we get $v_{\alpha_1}(f) = (0, 0, 1, 0)$ and $v_{\alpha_2}(f) = (0, 0, 0, 1)$. With respect to the ordering $x \prec z \prec y \prec w$ we get $v_{\alpha_1}(f) = (0, 1, 0, 0)$ and $v_{\alpha_2}(f) = (0, 0, 0, 1)$.*

Note that in the case of α_2, the ordering of variables does not change the valuation unlike the case for α_1. Ordering of variables is essential and must be fixed when comparing valuations of points.

Proposition 1 ([18, proposition 3.1]). *v_α is a valuation: that is, if f and g are non-zero elements of $\mathbb{R}[x_1, \ldots, x_n]$ and $\alpha \in \mathbb{R}^n$, then*

$$v_\alpha(fg) = v_\alpha(f) + v_\alpha(g) \text{ and } v_\alpha(f + g) \geq_{lex} \min\{v_\alpha(f),\ v_\alpha(g)\}.$$

Definition 8 (After [18, Definition 2.6]). *Let $n \geq 2$, and suppose that $f \in \mathbb{R}[x_1, \ldots, x_n]$ is non-zero and that $\beta \in \mathbb{R}^{n-1}$. The lex-least semi-valuation of f on/above β, $v'_\beta(f) = (v_1, \ldots, v_{n-1})$, is defined as the vector of $n-1$ non-negative integers consisting of the valuation outcomes of f at β, regarded as an element of $K[x_1, \ldots, x_{n-1}]$ where $K = \mathbb{R}[x_n]$. So $f = f_\beta(x_1 - \beta_1)_{v_1} \cdots (x_{n-1} - \beta_{n-1})^{v_{n-1}}$, and we call f_β the Lazard residue of f.*

The lex-least semi-valuation of f is called the *Lazard evaluation* of f above β, denoted by $v_\beta(f) \ni \mathbb{N}^{n-1}$ in [18]. The Lazard residue $f_\beta \in \mathbb{R}[x_n]$ is called the *Lazard evaluation* of f at β in [18].

Note that [18] uses v for two functions. When v is used as a valuation at, it is treated as a function from $\mathbb{R}^n \times \mathbb{R}[x_1, \ldots, x_n] \to \mathbb{N}^n$. When v is used as a valuation on/above, it is treated as a function from $\mathbb{R}^{n-1} \times \mathbb{R}[x_1, \ldots, x_n] \to \mathbb{N}^{n-1}$. We felt that it is necessary to keep these two functions separate. The older notation also has the potential to cause confusion when reading $v_\alpha(f)$. The reader needs to check whether $\alpha \in \mathbb{R}^n$ or $\alpha \in \mathbb{R}^{n-1}$, or whether $v_\alpha(f) \in \mathbb{N}^n$ or $v_\alpha(f) \in \mathbb{N}^{n-1}$.

The lex-least semi-valuation of f on $\beta \in \mathbb{R}^{n-1}$ must not be confused with the lex-least valuation at $\alpha \in \mathbb{R}^n$, defined in Definition 7. Notice that if $b = (\beta, b_n) \in \mathbb{R}^n$ then $v_b(f) = (v'_\beta(f), v_n)$ for some integer v_n: in other words, $v'_\beta(f)$ consists of the first $n - 1$ coordinates of the valuation of f at any point above β.

Lex-least semi-valuation and Lazard residue are also dependent on variable ordering as demonstrated by the following example.

Example 4. *Let $f(x, y, z, w) = x^2 + y^2 z - 2yz^2 + zw$ and $\beta = (0, 1, 0)$ with ordering $x \prec y \prec z \prec w$ then $v'_\beta(f) = (0, 0, 1)$ and $f_\beta = w + 1$. If we change the ordering to $x \prec y \prec w \prec z$ then $v'_\beta(f) = (0, 0, 0)$ and $f_\beta = z - 2z^2$.*

2.6 The Lazard Projection and Lazard Lifting

Lazard's idea [13, 18] is to instantiate Figure 1 with ϕ being lex-least-invariance. The nullification problem of [14] is bypassed by "Lazard lifting" L_L: when a polynomial $g \in P^{n-k}(S)$ nullifies on a cell C, its lex-least valuation is non-zero on C (at the sample point $(\alpha_1, \ldots, \alpha_k) \in C$, and in the whole of C). The valuation is also constant over C because C is part of a lex-least-invariant cylindrical algebraic decomposition of $\mathbb{R}^k$. We replace g by $g/(x_1 - \alpha_1)^{v_1} \cdots (x_k - \alpha_k)^{v_k}$ when lifting (analogous to L_C, Collins' sign-invariant CAD lifting) over C if this valuation is $(v_1, \ldots, v_k)$.

For this to work, we need an appropriate projection operator P_L. This is defined in [18] as follows.

Definition 9 ([18, Definition 2.1]). *Let A be a set of irreducible[2] polynomials in $\mathbb{Z}[x_1, \ldots, x_k]$. The Lazard projection $P_L(A)$ is the union of these sets of polynomials in $\mathbb{Z}[x_1, \ldots, x_{k-1}]$.*

1. *All leading coefficients of the elements of A.*
2. *All trailing coefficients of the elements of A.*
3. *All discriminants of the elements of A.*
4. *All resultants of pairs of distinct elements of A.*

$$P_L(A) := \mathrm{lc}(A) \cup \mathrm{tc}(A) \cup \mathrm{disc}(A) \cup \mathrm{res}(A). \tag{3}$$

Though not explicit in [18], any contents of the polynomials (with respect to x_k) also carry forward into $P_L(A)$. We also note that $P_L \subsetneq P_M$ as the middle coefficients do not occur in P_L.

Just as delineability is key to L_C (and to L_M), we need a corresponding definition in this setting.

Definition 10 ([18, Definition 2.10]). *Let f be a nonzero element of $\mathbb{Z}[x_1, \ldots, x_n]$ and S a subset of $\mathbb{R}^{n-1}$. We say that f is Lazard delineable on S if*

1. *the lex-least valuation of f on α is the same for each point $\alpha \in S$;*
2. *there exist finitely many continuous functions $\theta_1 \prec \cdots \prec \theta_k$ from S to $\mathbb{R}$, with $k \geq 0$, such that, for all $\alpha \in S$, the set of real roots of $f_\alpha(x_n)$ is $\{\theta_1(\alpha), \ldots, \theta_k(\alpha)\}$ (empty when $k = 0$);*
3. *if $k > 0$ there exist positive integers $m_1, \ldots m_k$ such that, for all $\alpha \in S$ and all i, m_i is the multiplicity of $\theta_i(\alpha)$ as a root of $f_\alpha(x_n)$.*

If the lex-least valuation is zero, this is the same as ordinary delineability, otherwise it is delineability of the lex-least residue f_α.

[2] [18, Remark 2.2] we could define this for square-free bases instead.

2.7 The Brown–McCallum Improvement

A key result on delineability of sections in the correctness proof for Brown–McCallum CAD is the following.

Theorem 2.1 ([4, Theorem 3]). *Suppose that $f(x_1,\dots,x_n) \in \mathbb{R}[x_1,\dots,x_n]$ has positive degree d in x_n and that $\mathrm{disc}_{x_n}(f)$ and $\mathrm{lc}_{x_n}(f)$ are non-zero (as elements of $\mathbb{R}[x_1,\dots,x_{n-1}]$). Let S be a connected analytic submanifold of $\mathbb{R}^{n-1}$ in which $\mathrm{disc}_{x_n}(f)$ and $\mathrm{lc}_{x_n}(f)$ are lex-least invariant, and at no point of which f vanishes identically. Then f is analytic delineable on S and hence f is lex-least invariant in every Lazard section and sector over S.*

This is less restrictive than [18, Theorem 5.1] for Lazard CAD where the delineability also requires the non-vanishing trailing coefficients and resultants. Conversely we have added the restriction "at no point of which f vanishes identically", which is essentially introdcing the concept of curtains.

For any non-trivial set A of polynomials $P_B(A) \subsetneq P_L(A)$, the difference being the trailing coefficients, which appear in P_L but not P_B. Can we eliminate them? Again, this would not affect the asymptotics, but might be useful in practice. The trailing coefficients are there for two reasons:

(1) To identify curtains;
(2) To ensure that the lex-least-invariant regions within the curtain form stacks.

If the feet of the curtains are isolated points, then reason 2 is irrelevant, and Hypothesis 1 provides a different way of identifying curtains.

In [18, p. 64], it is remarked that if $\mathrm{lc}(p)$ is nowhere vanishing, then $\mathrm{tc}(p)$ is not needed. The argument in [4] states that if $\mathrm{lc}(p)$ only vanishes at finitely many points, then $\mathrm{tc}(p)$ is not needed. However, the argument and projection process are more complex.

Hypothesis 1 (Brown–McCallum [4]). *There is a useful and efficient test* $\mathbf{T}$ *which, given $f \in \mathbb{Z}[x_1,\dots,x_m]$, returns either a finite set N of points in $\mathbb{R}^{m-1}$ or $\perp$. If N is returned, then N contains all the points $(\alpha_1,\alpha_2,\dots\alpha_{m-1})$ at which $f(\alpha_1,\alpha_2,\dots,\alpha_{m-1},x_m)$ vanishes identically.*

One way of satisfying the hypothesis is to compute a CAD of $\mathbb{R}^{m-1}$ for the coefficients a_i of $f = \sum a_i(x_1,\dots,x_{m-1})x_m^i$. This is not as expensive at it seems, as it is only of the a_i, irrespective of all the other polynomials in the original problem. As in [23], the authors would actually begin with a Gröbner basis of the a_i, and if this is zero-dimensional (in $\mathbb{C}^{m-1}$) then N is certainly finite.

The improved Lazard projection introduced in [4], denoted P_{BM}, takes, as well as a set of polynomials in m variables, a set γ_m of points in $\mathbb{R}^m$ and returns, as well as a set of polynomials in $m-1$ variables, a set γ_{m-1} of points in $\mathbb{R}^{m-1}$.

Definition 11 ([4, Definition 4]). *Let A be a set of irreducible[2] polynomials in $\mathbb{Z}[x_1,\dots,x_k]$, and Γ a finite set in $\mathbb{R}^k$ which contains points of nullification for elements in A. The polynomial part of the modified Lazard projection $P_{BM}(A)$ is the union of these sets of polynomials in $\mathbb{Z}[x_1,\dots,x_{k-1}]$.*

(1) *All leading coefficients of the elements of A.*
(2) *Those trailing coefficients of the elements of A for which* $\mathbf{T}$ *returns $\perp$.*
(3) *All discriminants of the elements of A.*
(4) *All resultants of pairs of distinct elements of A.*

The "points" part is the projection of the points previously identified together with those identified by $\mathbf{T}$.

$$P_{BM}(A,\Gamma)_{poly} := \mathrm{lc}(A) \cup \bigcup_{\mathbf{T}(f)=\perp} \mathrm{tc}(f) \cup \mathrm{disc}(A) \cup \mathrm{res}(A), \quad (4)$$

$$P_{BM}(A,\Gamma)_{points} := \{(\gamma_1,\dots,\gamma_{k-1}) | (\gamma_1,\dots,\gamma_k) \in \Gamma\} \cup \bigcup_{\mathbf{T}(f)\neq\perp} \{\mathbf{T}(f) | f \in A\}. \quad (5)$$

Though not explicit in [4], any contents of the polynomials (with respect to x_k) also carry forward into $P_{BM}(A,\Gamma)$.

L_{BM} is modified from L_L (Lazard lifting) by analogy with the modifications L_B brings to L_M: at each stage we add to the CAD D_k the points in Γ_k. This is possible because, from the first part of (5), the projections of these points are already point cells in D_{k-1}.

3 A SINGLE EQUATIONAL CONSTRAINT

Definition 12 ([12, Definition 2]). *An equational constraint (EC) is a polynomial equation logically implied by a quantifer-free formula Φ. If it is an atom of the formula, it is said to be explicit; if not, then it is implicit. If the constraint is visibly an equality one from the formula, i.e. the formula Φ is $(f=0) \wedge \Phi'$, we say the constraint is syntactically explicit.*

Φ might have more than one equational constraint, but in this section we fix one of these to be the chosen equational constraint.

3.1 In McCallum's CAD

In [16], McCallum, inspired by [7], adapted the method of [14] to lift an order-invariant cylindrical algebraic decomposition of $\mathbb{R}^{n-1}$ to a sign-invariant cylindrical algebraic decomposition, not of $\mathbb{R}^n$, but rather of the variety V_f of a primitive[3] polynomial $f \in \mathbb{Z}[x_1,\dots,x_n]$ of positive degree in x_n. This inherited the nullification problem of [14], but is useful when the polynomial problem takes the form $f=0 \wedge \Phi$, i.e. had an *equational constraint* (see also [12]). The projection operator from $\mathbb{R}^n$ to $\mathbb{R}^{n-1}$ is much smaller: if E is an irreducible basis for f, then

$$P_M^E(A) = P_M(E) \cup \{\mathrm{res}_{x_n}(f,g) : f \in E, g \in A \setminus E\}. \quad (6)$$

If we compare this with the original $P_M(A)$ from (1) we see that we have no need to compute $\mathrm{disc}(A \setminus E)$ or $\mathrm{res}(A \setminus E)$, which reduces the asymptotic complexity — the double exponent on the number of polynomials drops from n to $n-1$ [12].

3.2 In the Lex-least Case

An obvious question is whether we can do the same with the Brown–McCallum projection[4]. Let E be the set of irreducible factors of the equational constraint. As in (6), we can define a new projection operator from $\mathbb{R}^n$ to $\mathbb{R}^{n-1}$.

Definition 13. *Let E be an irreducible basis for the chosen equational constraint f. Define the modified Brown–McCallum projection*

[3] If f is not primitive with respect to x_n, then when the content vanishes, we have no constraint on x_n. See [8] for more on contents.
[4] This section is based on [19, Chapter 4], which used the Lazard projection instead.

P^E_{BM} *by*

$$P^E_{BM}(A,\Gamma)_{poly} := P_{BM}(E,\Gamma)_{poly} \cup \{\text{res}(f,g) : f \in E, g \in A \setminus E\}. \tag{7}$$

$$P^E_{BM}(A,\Gamma)_{points} := P_{BM}(E,\Gamma)_{points}. \tag{8}$$

Note that, as well as not computing any projection polynomials from $A \setminus E$ alone, we are also only interested in the feet of point curtains from E, not from the whole of A.

The following theorem is the lex-least equivalent of [16, Theorem 2.2], and is proved in [19].

Theorem 3.1 ([19, Theorem 18]). *Let $n \geq 2$ and let $f, g \in \mathbb{R}[x_1, \ldots, x_n]$ be of positive degrees in the main variable x_n. Let S be a connected subset of $\mathbb{R}^{n-1}$. Suppose that f is Lazard delineable on S, that $\text{res}_{x_n}(f,g)$ is lex-least invariant on S, and that V_f does not have a curtain on S. Then g is sign-invariant in each Lazard section of f over S.*

Theorem 3.2 (cf [19, Theorem 25]). *Let $A = \{f_1, \ldots, f_m\}$ be a set of pairwise relatively prime irreducible polynomials in n variables $x_1, ..., x_n$ of positive degrees in x_n, where $n \geq 2$. Let Γ be a finite set of points in $\mathbb{R}^n$. Let S be a subset of $\mathbb{R}^{n-1}$ obtained via the Brown–McCallum algorithm, so that each element of $P^E_{BM}(A,\Gamma)_{poly}$ is lex-least invariant in S. Then every element of A is lex-least invariant in the Lazard sections and sectors of every other element.*

Proof. Set $f = f_1 \ldots f_m$. Then the leading coefficient of f is non-zero and hence lex-least invariant in S. We know $\text{disc}_{x_n}(f)$ can be expanded as follows:

$$\text{disc}_{x_n}(f_1 \ldots f_m) = \Big(\prod_{i=1}^{m} \text{disc}_{x_n}(f_i)\Big)\Big(\prod_{1 \leq i < j \leq m} \text{res}_{x_n}(f_i, f_j)\Big). \tag{9}$$

If everything in the RHS of (9) is lex-least invariant in S, then $\text{disc}_{x_n}(f)$ is lex-least invariant. Hence, by Theorem 2.1, f is Lazard delineable over S. Since $\text{res}_{x_n}(f_i, f_j)$ for all $1 \leq i < j \leq m$ are lex-least invariant in S, the Lazard sections of any two f_i and f_j $(i \neq j)$ are either disjoint or the same. Hence every element of A is lex-least invariant in the Lazard sections of every other element. □

Theorem 3.3 ([19, Theorem 26]). *Let A be a set of pairwise relatively prime irreducible polynomials in n variables $x_1, ..., x_n$ of positive degrees in x_n, where $n \geq 2$. Let E be a subset of A and let Γ be a finite set of points in $\mathbb{R}^n$. Let S be a connected subset of $\mathbb{R}^{n-1}$. Suppose that each element of $P^E_{BM}(A,\Gamma)_{poly}$ is lex-least invariant in S. Then each element of E is Lazard delineable on S and exactly one of the following must be true.*

1. *The hypersurface V_E defined by the product of the elements of E has a curtain over S.*
2. *The Lazard sections over S of the elements of E are pairwise disjoint. Each element of E is lex-least invariant in every such Lazard section. Each element of $A \setminus E$ is sign invariant in every such Lazard section.*

Hence, off the curtains of E, the lifted cells from the $\mathbb{R}^{n-1}$ lex-least-invariant decomposition for $P^E_{BM}(A,\Gamma)$ form a sign-invariant decomposition of V_E for A.

Proof. If there are no points in S such that E is nullified, then Theorem 2.1 implies that each element of E is Lazard delineable since $P_{BM}(E,\Gamma)_{poly} \subseteq P^E_{BM}(A,\Gamma)_{poly}$. Otherwise V_E has a curtain on S and the Lazard sections of the elements of E that do not contain curtains are pairwise disjoint. Now let $f \in E$ (such that f does not have a curtain over S), let σ be a section of f over S and let $g \in A$. If $g \in E$, then by Theorem 2.1, g is lex-least invariant in every section and sector of f. If $g \notin E$ then, $R = \text{res}(f,g) \in P^E_{BM}(A,\Gamma)_{poly}$ and by the assumption R is lex-least invariant in S. Then by Theorem 3.1, g is sign invariant in σ. □

The question of what happens on the curtains of E is taken up in the next section.

4 CURTAINS AND A SINGLE EQUATIONAL CONSTRAINT

McCallum lifts order invariance to sign invariance in [16] and order invariance in [17], but only if the input polynomials do not have any curtains. Since curtains are already an issue for the base algorithm, other problems caused by curtains for equational constraints went unnoticed.

4.1 Classifying Curtains

In the case of a single equational constraint some curtains can be dealt simply. Therefore, we classify curtains further.

Definition 14. *Let $f \in \mathbb{R}[x_1, \ldots, x_n]$ and $\alpha \in \mathbb{R}^{n-1}$. We say that f has a point curtain at α if $f(\alpha, y) = 0$ for all $y \in \mathbb{R}$, and there exists a Euclidean open neighbourhood $U \subset \mathbb{R}$ of α such there exists no $\beta \in U \setminus \{\alpha\}$ such that $f(\beta, y) = 0$ for all $y \in \mathbb{R}$.*

Example 5. *We provide one example to illustrate the difference.*

- *Point Curtain: $f(x,y,z) = x^2 + zy^2 - z$ has point curtains at $(0,1)$ and $(0,-1)$.*
- *Non-Point Curtain: Consider $f(x,y,z) = -x^3y^3z - xy^4z + xy^3z^2 + x^4 + 2x^3z + x^2y - x^2z + 2xyz - 2xz^2$. It has a non-point curtain on $(0,y)$ for all $y \in \mathbb{R}$.*

Note that in $\mathbb{R}^3$, all non-point curtains are explicit curtains. The example for non-point curtains in Example 5 is of the form $x \cdot$(quintic) so it is an explicit curtain (see Definition 3). The curtains in Example 1 are explicit non-point, and implicit point curtains, respectively. These curtains are plotted in Figure 2.

Now let us explain how curtains cause a problem when we try to exploit the hypersurfaces defined by an equational constraint.

Example 6. *Let $f = x^2 + y^2 - 1$ (which we assume to be an EC), $g_1 = z - x - 1$ and $g_2 = z - y - 1$. Then $\text{res}_z(f, g_1) = x^2 + y^2 - 1 = \text{res}_z(f, g_2)$, and this gives us no information about $\text{res}(g_1, g_2)$. In such cases, when the EC has a non-point curtain, it becomes impossible to use P^E_{BM} to detect the intersections of the other constraints on that curtain. In this example, f does not have positive degree in z, so would not be considered anyway.*

Example 7. *Let $f = x - yz$ (which we assume to be an EC), $g_1 = z - x$ and $g_2 = z - y$ (which we assume are not ECs). Then $\text{res}(f, g_1) = yx - x$ and $\text{res}(f, g_2) = y^2 - x$ and $\text{res}(\text{res}(f, g_1), \text{res}(f, g_2)) = x^2(1 - x)$. This gives us information about the interaction of g_1 and g_2. In such cases, when the EC has a point curtain, P^E_{BM} detects the intersections of the other constraints on that curtain.*

Being able to classify curtains is just as important. For a given point $\beta \in \mathbb{R}^{n-1}$, where $f \in \mathbb{R}[x_1, \ldots, x_n]$ nullifies, one can classify a curtain with β in its foot by looking at the lex-least semi-valuation. If $\nu'_\beta(f) \neq 0$ one checks the 1-cell neighbours. If any one of those cells also has non-zero semi-valuation at this point, then the curtain is a non-point curtain.

4.2 Point Curtains

Point curtains do not pose a problem for the lifting process.

Proposition 2 ([19, Proposition 6]). *Let $f, g \in \mathbb{R}[x_1, \ldots, x_n]$ and $\alpha \in \mathbb{R}^{n-1}$ and suppose that the variety V_f defined by f contains a curtain at α. If $\mathrm{res}(f, g)(\alpha) = 0$ then V_g contains a curtain at α or intersects V_f at finitely many points over α.*

The modified projection operator P^E_{BM} only projects information about the equational constraint and the resultants between the equational and non-equational constraints. Test T in P^E_{BM} will detect curtains with point feet in $\mathbb{R}^{n-1}$, and will return $\perp$ if it thinks there are non-point curtains. Any curtains, point or non-point, detected by P_{BM} at lower levels lift to curtains with non-point feet in $\mathbb{R}^{n-1}$. The following theorem shows that P^E_{BM} actually produces a sin-invariant CAD for curtains with point feet in $\mathbb{R}^{n-1}$.

Theorem 4.1 ([19, Theorem 20]). *Let $f, g \in \mathbb{R}[x_1, \ldots, x_n]$, E be an irreducible basis for f our equational constraint, and suppose that $\alpha \in \mathbb{R}^{n-1}$ is a cell of $P^E_{BM}(\{f, g\}, \Gamma)_{points}$. Then g is sign invariant in the sections and sectors of $f_\alpha g_\alpha$.*

Proof. By assumption V_f has a point curtain at α. This implies that there exists a neighbourhood B of α in $\mathbb{R}^{n-1}$ such that $\nu'_\beta(f) = 0^{n-1}$ for $\beta \in B \setminus \{\alpha\}$ and $\nu'_\alpha(f) \neq 0^{n-1}$. Because we are using the projection operator $P_{BM}(E)$, where E is the set of irreducible factors of f, the polynomial f is lex-least invariant with respect to the CAD produced by the projection operator $P_{BM}(E)$. Since the polynomial f is lex-least invariant with respect to the CAD and V_f has a curtain at α, the CAD consists of cells of the form $\alpha \times (a, b)$ or point cells $\alpha \times \{c\}$, where $a, b, c \in \mathbb{R}$.

We know $\mathrm{res}(f, g)(\alpha) = 0$. Then by Proposition 2, V_g has a point curtain at α or intersects V_f at finitely many points. If V_g has a curtain at α then g is sign invariant (zero) in all cells over α. Suppose V_f only intersects V_g at finitely many points and let $Z = \{\beta_1, \ldots, \beta_k\}$ (arranged in increasing order) be the roots of $f_\alpha g_\alpha$. Then the cells above α will be of the form $\alpha \times (a, b)$ or point cells $\alpha \times \{c\}$, where $a, b, c \in Z$. Note that β_i has to be either a root of f_α or g_α for each i. Since $g = 0$ intersects $f = 0$ at only finitely many points and all roots of g_α are in Z, g is non-zero in $\alpha \times (\beta_i, \beta_{i+1})$. g is zero on the point cells (α, β_i) where β_i is a root of g, and non-zero otherwise. □

Theorem 4.1 is the first point at which the merger of Lazard-style CAD and equational constraint theory is not straight-forward.

4.3 Non-point Curtains

Even if there are non-point curtains present, we know from Theorems 3.3 and 4.1 that use of P^E_{BM} at the first (projection) layer and P_{BM} at subsequent layers generates a sign-invariant CAD away from the non-point curtains. So the challenge is purely confined to the non-point curtains (if any). Example 6 shows that we need to consider the resultants of polynomials other than equational constraints, as P^E_{BM} has lost too much information compared with P_{BM}. Hence our strategy is the following.

(1) Project with P^E_{BM} at the first layer and P_{BM} at subsequent layers, and lift to $\mathbb{R}^{n-1}$ to get decompositions $D_1, \ldots, D_{n-1}$, and to D_n, where the other constraints are sign-invariant over the non-curtain and point-curtain cells where $f = 0$.
(2) If there are no non-point curtains we are done. Otherwise let $(\alpha^{(i)}_1, \ldots, \alpha^{(i)}_{n-1})$ be the sample points of the cells $C_i \in D_{n-1}$ forming the non-point curtains. We know the chosen equational constraints is zero on these cells.
(3) Project all other constraints to $\mathbb{R}^1$ using P_{BM}. Let P_i be the polynomial part of the projection at level $\mathbb{R}^i$.
(4) In $\mathbb{R}^1$, consider the cells whose sample points are $\alpha^{(i)}_1$, and decompose them further according to P_1, refining D_1 to $\widehat{D}_1$, introducing the necessary sample points.
(5) In $\mathbb{R}^2$, consider those cells whose sample points are $(\alpha^{(i)}_1, \alpha^{(i)}_2)$. Their base in $\mathbb{R}^1$ has been split in the previous stage, but they may also need to be split further in the x_2 direction by the polynomials in P_2. Appropriate new sample points will need to be computed.
(6) Continue to $\mathbb{R}^{n-1}$ and then lift these cells to $\mathbb{R}^n$.
(7) We have subdivided some cells in each D_i to get $\widehat{D}_i$. If we want a completely cylindrical decomposition, these subdivisions may need to be extended to the whole of each $\widehat{D}_i$.

Step (7) is purely a data structure manipulation: no new polynomial computations need to be done beyond the computation of sample points for the newly-split cells.

To decompose a curtain of an equational constraint is the same as decomposing a set $S \times \mathbb{R}$ where $S \subset \mathbb{R}^{n-1}$. When working in a curtain, the equational constraint does not give us any information on how the rest of the polynomial constraints interact. The following theorem validates our method of decomposing curtains separately.

Theorem 4.2 ([19, Theorem 21]). *Let $A = \{f, g_1, \ldots g_m\}$ be a set of polynomial constraints where f is the equational constraint. Let D be the CAD obtained by the applying the steps (1)–(7). Then $g_1, \ldots, g_m$ are sign-invariant in the cells of D where $f = 0$.*

Proof. Let S be a connected subset of $\mathbb{R}^{n-1}$ such that f is Lazard delineable over S and the set $\{\mathrm{res}_{x_n}(f, g_j) | 1 \leq j \leq m\}$ is lex-least invariant over S. If V_f does not contain a non-point curtain on S, then Theorems 3.3 and 4.1 imply the result. Let us assume the contrary. Since V_f contains a non-point curtain on S, the aforementioned steps splits up S into connected, non-intersecting subsets, where f is Lazard delineable and $P_{BM}(\{g_1, \ldots, g_m\}, \Gamma)$ is lex-least invariant [4]. Hence by Theorem 3.1, $g_1, \ldots, g_m$ are lex-least invariant in the cells of the curtains of V_f, hence also sign-invariant. □

5 MULTIPLE EQUATIONAL CONSTRAINTS

We now generalise the previous ideas to include multiple equational constraints, which involves modifying several levels of projection.

5.1 Higher-dimensional Curtains

Definition 15 (cf [19, Definition 43]). *A variety $C \subseteq \mathbb{R}^n$ is called an m-curtain if, whenever $(x_1, \ldots, x_n) \in C$, then $(x_1, \ldots, x_{n-m}, y_{n-m+1}, \ldots, y_n) \in C$ for all $(y_{n-m+1}, \ldots, y_n) \in \mathbb{R}^m$.*

In other words, C is an m-curtain if it is a union of fibres of $\mathbb{R}^n \to \mathbb{R}^{n-m}$.

Definition 16 (cf [19, Definition 44]). *Suppose $S \subseteq \mathbb{R}^{n-m}$ and $f \in \mathbb{R}[x_1, \ldots, x_n]$. We say that V_f (or f) has a m-curtain at S if for all $(\alpha_1, \ldots, \alpha_{n-m}) \in S$ and for all $(y_{n-m+1}, \ldots, y_n) \in \mathbb{R}^m$ we have $f(\alpha_1, \ldots, \alpha_{n-m}, y_{n-m+1}, \ldots, y_n) = 0$. We call S the foot of the curtain.*

5.2 The Modified Projection

For multiple equational constraints, we modify our projection operator further, inspired by [17]. We now assume that we are given a quantifier-free formula of the form

$$(f_1 = 0) \wedge \ldots \wedge (f_k = 0) \wedge \Phi,$$

where Φ contains polynomial constraints g_j.

In this formula, the equational constraints are given by the polynomials $f_1, \ldots, f_k$. Since a solution set to this Tarski formula must satisfy the equational constraints, we focus our attention on the intersection of equational constraints.

Definition 17. *Let A be a finite set of irreducible polynomials in $\mathbb{R}[x_1, \ldots, x_n]$ with $n \geq 2$ and let E be a subset of A. The modified Brown–McCallum projection operator for multiple equational constraints $\hat{P}^E_{BM}(A)$ is the subset of $\mathbb{R}[x_1, \ldots, x_{n-1}]$ consisting of the following polynomials.*

1. *All leading coefficients of the elements of A.*
2. *The trailing coefficients of the elements of A required by test* **T**.
3. *All discriminants of the elements of A.*
4. *All resultants $\mathrm{res}_{x_n}(f, g)$, where $f \in E$ and $g \in A$ and $f \neq g$.*

This differs from Definition 13 precisely as [17] differs from [16] — by adding item (3). As before, we shall choose equational constraints and set E as the set of their irreducible factors.

The way we take advantage of multiple equational constraints is by using this operator recursively. The resultants between the chosen equational constraint and the remaining equational constraints are equational constraints of the projected polynomials. Hence we take the equational constraints one at a time, so if we have k equational constraints we will use $\hat{P}^E_{BM}(A)$ for the first k projections: see [12] for more on multiple equational constraints. The following theorems validate the use of $\hat{P}^E_{BM}(A)$ recursively.

Theorem 5.1 (cf [19, Theorem 22]). *Suppose $f, g \in \mathbb{R}[x_1, \ldots, x_n]$ both have positive degree in x_n. Let $S \subset \mathbb{R}^{n-1}$ be a connected subset such that f does not have a curtain over S. If $\mathrm{disc}_{x_n}(g)$, $\mathrm{lc}_{x_n}(g)$ and $\mathrm{res}_{x_n}(f, g)$ are all lex-least invariant over S, then g is lex-least invariant on every section and sector of f over S.*

Proof. From Theorem 3.1 we know that g is sign invariant in the sections of f. Since $\mathrm{disc}_{x_n}(g)$ and $\mathrm{lc}_{x_n}(g)$ are lex-least invariant over S, g is Lazard delineable over S from Theorem 2.1. Hence g is lex-least invariant in the sections and sectors of f. □

5.3 Correctness off Curtains

Theorem 5.2 (cf [19, Theorem 27]). *Let A be a set of irreducible polynomials in $x_1, \ldots, x_n$ of positive degree in x_n, where $n \geq 2$. Let E be a subset of A and let Γ be a finite set of points in $\mathbb{R}^n$. Let S be a connected subset of $\mathbb{R}^{n-1}$. Suppose that each element of $\hat{P}^E_{BM}(A, \Gamma)$ is lex-least invariant in S. Then each element of E is Lazard delineable over S and exactly one of the following is true.*

1. *The hypersurface defined by the product of the elements of E has a curtain whose foot is S.*
2. *The Lazard sections of the elements of E are pairwise disjoint over S. Each element of A is lex-least invariant in such Lazard sections.*

Proof. If we are in case 1, then at least one element of E is zero over the whole of the curtain, so case 2 is meaningless. Hence the two cases are disjoint.

So assume that S is not a foot of a curtain for any element of E. Then, since $\hat{P}^E_{BM}(A, \Gamma) \supset P_{BM}(E)$, the Lazard sections of the elements of E are pairwise disjoint over S. Then by Theorem 5.1 the elements of A are lex-least invariant in such Lazard sections. □

Hence, if we never encounter curtains, we have a lex-least invariant stack over S. Assuming we have (after computing resultants to propagate equational constraints) an equational constraint in each of $x_n, \ldots, x_{n-k+1}$, then k applications of $\hat{P}^E_{BM}$ followed by $n-k-1$ applications of P_{BM} and finally $n-1$ applications of Lazard lifting, will give us a lex-least invariant (for A) CAD of $V_{f_1,\ldots,f_k}$ in the absence of curtains.

5.4 Curtains

But what if we encounter a curtain of an equational constraint (either an original equation constraint, or a resultant of equational constraints) in the lifting? Here we have to take a leaf out of §4.3, where we projected the whole of $A \setminus E$ over the curtain. There may, however, be several curtains, possibly at different levels of lifting, and these curtains may be of derived equationsal constraints.

Hence we suggest that, *if* curtains are detected in the lifting phase, we follow the methodology of §4.3 but do a complete projection of A to $\mathbb{R}^1$ using P_{BM}, and then follow steps (4)–(6) to refine the decomposition over those stacks that have curtains in them.

6 EXAMPLES

More examples are given in [22, §7.2] and [10]. The example we consider here is the first example in [22, §7.2], taken from [11, §2.2.3]. As originally posed, the problem is

$$\exists u\, \exists v \;\; -uv + x = 0 \wedge -uv^2 + y = 0 \wedge -u^2 + z = 0. \tag{10}$$

The order chosen by the heuristics described in [22] is $u \prec v \prec y \prec z \prec x$. Because we have multiple equational constraints, we can use Gröbner bases, as suggested in [9, 12, 23]. This converts the Cylindrical Algebraic Decomposition problem into

$$\{vx - y, uv - x, \mathbf{ux - vz}\}, \tag{11}$$

where the emboldened equation is taken as the equational constraint. This projects as follows:

$$\{-v^2 z + uy, \mathbf{u^2 - z}\}; \{\mathbf{uv^2 - y}\}; \{v\}; \{u\}. \tag{12}$$

Having chosen $uz - vz$ as the initial equational constraint, the curtains are $u = 0 \wedge v = 0$ and $u = 0 \wedge z = 0$, but in fact they are decomposed into 15 cells (each into 9 but three are duplicates) as given in [22, Table 7.1]. As described in §5.4, we have to lift $A \setminus E$ i.e. $\{vx - y, uv - x\}$ over these curtains.

$$\{vx - y, uv - x\}; \emptyset, ; \{y, uv^2 - y\}; \{v\}; \{u\} \tag{13}$$

The combined result is a CAD with 591 cells.

If we merely adopt the "Single Equational Constraint" methodology, the Lazard projection (as on [22, p. 238]) is

$$\{-v^2z + uy, u^2 - z\}; \{y, uv^2 - y\}; \{v\}; \{u\} \tag{14}$$

which gives a total of 951 cells, due to the extra y polynomial.

This y is a trailing coefficient which the Brown–McCallum projection does not need, so in this case the behaviour would be the same as multiple equation constraints (which is to be expected, as the savings after the first projection only manifest when there are more than two polynomials in all.

7 COMPLEXITY ANALYSIS

Definition 18. *Let $\{f_i\}$ be a finite set of polynomials. The combined maximum degree of $\{f_i\}$ is the maximum element of the set*

$$\bigcup_j \left\{ \deg_{x_j} \prod_i f_i \right\}.$$

Definition 19 ([14, Section 6.1]). *A set of polynomials has the (m, d)-property if it can be partitioned into m sets, such that the combined maximum degree of each set is less than or equal to d.*

The outcome of the complexity analysis carried out in [19, Chapter 8] (based on [14]) is summarised in Table 1. If the set of inputs has the (m, d)-property then the set of projected polynomials after projection has the $(M, 2d^2)$-property, with M as in Table 1.

Table 1: Growth of polynomials in CAD

Theory by		Original	Single EC	Multiple EC
McCallum	[14]	$\left\lfloor \frac{(m+1)^2}{2} \right\rfloor$	$\left\lfloor \frac{5m+4}{4} \right\rfloor$	$\left\lfloor \frac{11m}{4} \right\rfloor$
Lazard	[18]	$\left\lfloor \frac{(m+1)^2}{2} \right\rfloor$	$\left\lfloor \frac{5m+3}{4} \right\rfloor$	$\left\lfloor \frac{9m-1}{4} \right\rfloor$
Brown-McCallum	[4]	$\frac{m(m+1)}{2}$	$\left\lfloor \frac{5m+2}{4} \right\rfloor$	$\left\lfloor \frac{9m-2}{4} \right\rfloor$

We note that there is very little improvement in the asymptotics as we move down the table: the only leading term to change is in the Multiple EC column, from 11 to 9. The differences in practice are more substantial: [4, p 145] shows that their method ("Improved Lazard") only requires 27% of the cells of the basic Lazard method.

This shows that, if we have only one equational constraint, the "Single EC" methods are better. However, if we have multiple equational constraints, the "Multiple EC" methods are better: for projections with two (or more) equational constraints, the number of polynomials is bounded by $O(m^4)$ for the original methods, $O(m^2)$ for the "Single EC" and $O(m)$ for the "Multiple EC" methods.

8 CONCLUSION

We have analysed both single and multiple equational constraints. Table 1 shows that, if we only have one equation constraint, the "single" method is distinctly better at the cost of only producing a sign-invariant decomposition. However only a few users would wary of such change.

8.1 Single Equational Constraint

Theorem 3.3 tells us that we can transport the single equational work of [16] (which can fail because of nullification) to the Brown–McCallum projection in the cases where [16] works. The complexity gain [12] is the same: the double exponent of the number of polynomials is reduced by 1. Theorem 4.1 extends this to point curtains, with the same complexity improvement.

Theorem 4.2 shows that we can work with an equational constraint even in the presence of non-point curtains, but the efficiency gain *on the curtain* is more modest, and does not lend itself to a simple formulation.

8.2 Multiple Equational Constraints

Theorem 5.2 tells us that we can transport the single equation al work of [17] (which failed with nullification) to the Brown- -McCallum projection in the cases where [17] worked. The compl exity gain [12] is the same: the double exponent of the number of polynomials is reduced by k, the number of consecutive layers where we can apply $\hat{P}^E_{BM}$.

In the presence of curtains, our method is probably more efficient than purely using [4], but this is hard to quantify. See §6.

8.3 Future Work

There are several open questions.

(1) We have talked about "the number of consecutive layers where we" have an equational constraint. This is implicit in [17] as well. But it appears that, in the absence of curtains, it should be possible to "mix and match" between *PBM* and $\hat{P}^E_{BM}$, using $\hat{P}^E_{BM}$ at every layer where there is an equations constraint. But formalising this would require a suitable induction hypothesis.
(2) §4.2 deals more neatly with point curtains in the case of a single equational constraint. When this curtain occurs in the lift from $\mathbb{R}^{n-1}$ to $\mathbb{R}^n$ we can probably do the same for multiple equational constraints. But what if the point curtain is elsewhere? We currently propose treating is as a non-point curtain — can we do better?
(3) [12] suggested using Gröbner bases rather than resultants to propagate equational constraints, in the setting of [17]. That has been shown to work in §6.
(4) [1, see also [2]] introduced the concept of a "truth table invariant CAD", where equational constraints might only govern part of a formula, as in $((f_1 = 0) \wedge \Phi_1) \vee ((f_2 = 0) \wedge \Phi_2)$. This theory build on [16], so should transfer.

In addition, [19, §9.1] asks whether lex-least is the only other valuation (besides sign in [6] and order in [14]) that we should consider. It is noted there that "The property of upper-semicontinuity guarantees that if two polynomials are valuation invariant in a set, then their product is valuation invariant in the same set and vice-versa, which helps us take advantage of equational constraints". So what other upper-semicontinuous valuations might we consider?

REFERENCES

[1] R.J. Bradford, J.H. Davenport, M. England, S. McCallum, and D.J. Wilson. 2013. Cylindrical Algebraic Decompositions for Boolean Combinations. In *Proceedings ISSAC 2013*. 125–132.

[2] R.J. Bradford, J.H. Davenport, M. England, S. McCallum, and D.J. Wilson. 2016. Truth table invariant cylindrical algebraic decomposition. *J. Symbolic Comp.* 76 (2016), 1–35.

[3] C.W. Brown. 2001. Improved Projection for Cylindrical Algebraic Decomposition. *J. Symbolic Comp.* 32 (2001), 447–465.

[4] C.W. Brown and S. McCallum. 2020. Enhancements to Lazard's Method for Cylindrical Algebraic Decomposition. In *Computer Algebra in Scientific Computing CASC 2020 (Springer Lecture Notes in Computer Science, Vol. 12291)*, F. Boulier, M. England, T.M. Sadykov, and E.V. Vorozhtsov (Eds.). 129–149. https://doi.org/10.1007/978-3-030-60026-6_8

[5] C. Chen, M. Moreno Maza, B. Xia, and L. Yang. 2009. Computing Cylindrical Algebraic Decomposition via Triangular Decomposition. In *Proceedings ISSAC 2009*, J. May (Ed.). 95–102.

[6] G.E. Collins. 1975. Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition. In *Proceedings 2nd. GI Conference Automata Theory & Formal Languages (Springer Lecture Notes in Computer Science, Vol. 33)*. 134–183.

[7] G.E. Collins. 1998. Quantifier elimination by cylindrical algebraic decomposition — twenty years of progess. In *Quantifier Elimination and Cylindrical Algebraic Decomposition*, B.F. Caviness and J.R. Johnson (Eds.). Springer Verlag, Wien, 8–23.

[8] J.H. Davenport and M. England. 2016. Need Polynomial Systems be Doubly-exponential?. In *International Congress on Mathematical Software ICMS 2016 (Springer Lecture Notes in Computer Science, Vol. 9725)*, G-M. Greuel, T. Koch, P. Paule, and A. Sommese (Eds.). 157–164.

[9] J.H. Davenport and M. England. 2017. The Potential and Challenges of CAD with Equational Constraints for SC-Square. In *Proceedings MACIS 2017: Mathematical Aspects of Computer and Information Sciences*. 280–285.

[10] J.H. Davenport, Z.P. Tonks, and A.K. Uncu. 2023. A Poly-algorithmic Approach to Quantifier Elimination. https://arxiv.org/abs/2302.06814.

[11] A. Dolzmann, A. Seidl, and Th. Sturm. 2004. Efficient Projection Orders for CAD. In *Proceedings ISSAC 2004*, J. Gutierrez (Ed.). 111–118.

[12] M. England, R.J. Bradford, and J.H. Davenport. 2020. Cylindrical Algebraic Decomposition with Equational Constraints. In *Symbolic Computation and Satisfiability Checking: special issue of Journal of Symbolic Computation*, J.H. Davenport, M. England, A. Griggio, T. Sturm, and C. Tinelli (Eds.). Vol. 100. 38–71.

[13] D. Lazard. 1994. An Improved Projection Operator for Cylindrical Algebraic Decomposition. In *Proceedings Algebraic Geometry and its Applications: Collections of Papers from Shreeram S. Abhyankar's 60th Birthday Conference*, C.L. Bajaj (Ed.). 467–476.

[14] S. McCallum. 1985. *An Improved Projection Operation for Cylindrical Algebraic Decomposition.* Technical Report 578. Computer Science University Wisconsin at Madison. https://minds.wisconsin.edu/bitstream/handle/1793/58594/TR578.pdf?sequence=1

[15] S. McCallum. 1998. An Improved Projection Operation for Cylindrical Algebraic Decomposition. Springer-Verlag Vienna, 242–268.

[16] S. McCallum. 1999. On Projection in CAD-Based Quantifier Elimination with Equational Constraints. In *Proceedings ISSAC '99*, S. Dooley (Ed.). 145–149.

[17] S. McCallum. 2001. On Propagation of Equational Constraints in CAD-Based Quantifier Elimination. In *Proceedings ISSAC 2001*, B. Mourrain (Ed.). 223–230.

[18] S. McCallum, A. Parusiński, and L. Paunescu. 2019. Validity proof of Lazard's method for CAD construction. *J. Symbolic Comp.* 92 (2019), 52–69.

[19] A.S. Nair. 2021. *Curtains in Cylindrical Algebraic Decomposition.* Ph. D. Dissertation. University of Bath. https://researchportal.bath.ac.uk/en/studentTheses/curtains-in-cylindrical-algebraic-decomposition

[20] A.S. Nair, J.H. Davenport, and G.K. Sankaran. 2020. Curtains in CAD: Why Are They a Problem and How Do We Fix Them? In *Mathematical Software — ICMS 2020*, A.M. Bigatti, J. Carette, J.H. Davenport, M. Joswig, and T. de Wolff (Eds.). Springer Lecture Notes in Computer Science, Vol. 12097. Springer, 17–26.

[21] A. Nair, J. Davenport, G. Sankaran, and S. McCallum. 2019. Lazard's CAD exploiting equality constraints. *ACM Comm. Computer Algebra* 53 (2019), 138–141.

[22] Z. Tonks. 2021. *Poly-algorithmic Techniques in Real Quantifier Elimination.* Ph. D. Dissertation. University of Bath. https://researchportal.bath.ac.uk/en/studentTheses/poly-algorithmic-techniques-in-real-quantifier-elimination

[23] D.J. Wilson, R.J. Bradford, and J.H. Davenport. 2012. Speeding up Cylindrical Algebraic Decomposition by Gröbner Bases. In *Proceedings CICM 2012*, J. Jeuring et al. (Ed.). 279–293.