A Geometric Approach to Cylindrical Algebraic Decomposition

Rizeng Chen
xiaxueqaq@stu.pku.edu.cn

School of Mathematical Sciences, Peking University, China

April 22, 2026

The Speaker

Speaker. Rizeng Chen

- I am Rizeng Chen (pronounced as "ReeTseng Chern").
- I am currently a fifth-year PhD candidate from Peking University, China.
- My advisor is Prof. Bican Xia.
- My research interest lies in the intersection of algebraic geometry, logic, and theoretical computer science.

Question

Question: Have you ever dreamed of someone who proves everything for you?

Leibniz's Dream

Leibniz had dreamed so, hence he invented the first calculator that can perform all four basic arithmetic operations.

Figure 1: Leibniz's monument in Leipzig (Credit: Ad Meskens)

Hilbert's *Entscheidungsproblem*

- Hilbert was more enthusiastic. He called for an algorithm that decides the truth value of any mathematical statements.

Hilbert's *Entscheidungsproblem*

- Hilbert was more enthusiastic. He called for an algorithm that decides the truth value of any mathematical statements.
- The German word "Entscheidungsproblem" means "decision problem".

Hilbert's *Entscheidungsproblem*

- Hilbert was more enthusiastic. He called for an algorithm that decides the truth value of any mathematical statements.
- The German word "Entscheidungsproblem" means "decision problem".
- Hilbert and Ackermann called it "das Hauptproblem der mathematischen Logik" – "the main problem of mathematical logic".

Decision Problem over the Reals

We are interested in the decision problem over the reals.

The Decision Problem

- In the first order theory of the reals, an atom is a polynomial equation $f = 0$ or a polynomial inequality $f > 0$.
- Using logical connectives $(\wedge, \vee, \neg)$, atoms can be combined to get a quantifier-free formula $\Psi(x_1, \ldots, x_n)$.
- A closed formula is a fully quantified formula

$$\Phi = (Q_1 x_1)(Q_2 x_2) \cdots (Q_n x_n)\Psi(x_1, \ldots, x_n),$$

where Q_i are quantifiers $\forall$ or $\exists$.

Goal: Find an algorithm to decide whether Φ is true or not.

Decision Problem over the Reals

We are interested in the decision problem over the reals.

The Decision Problem

- In the first order theory of the reals, an atom is a polynomial equation $f = 0$ or a polynomial inequality $f > 0$.
- Using logical connectives $(\wedge, \vee, \neg)$, atoms can be combined to get a quantifier-free formula $\Psi(x_1, \ldots, x_n)$.
- A closed formula is a fully quantified formula

$$\Phi = (Q_1 x_1)(Q_2 x_2) \cdots (Q_n x_n)\Psi(x_1, \ldots, x_n),$$

where Q_i are quantifiers $\forall$ or $\exists$.

Goal: Find an algorithm to decide whether Φ is true or not.

E.g., for all a, b, c, there exists x such that $x^3 + ax^2 + bx + c = 0$ holds.

$$(\forall a)(\forall b)(\forall c)(\exists x)(x^3 + ax^2 + bx + c = 0).$$

Decision Problem over the Reals

This connects very different areas of mathematics.

- Logic
  e.g. $(\exists x)(\exists y)((x^2 + y^2 - 1 = 0) \wedge (y - 2 = 0))$

Decision Problem over the Reals

This connects very different areas of mathematics.

- Logic
  e.g. $(\exists x)(\exists y)((x^2 + y^2 - 1 = 0) \wedge (y - 2 = 0))$
- Geometry
  e.g. The emptiness of the (semi-)algebraic set $V(x^2 + y^2 - 1, y - 2)$

Decision Problem over the Reals

This connects very different areas of mathematics.

- Logic
  e.g. $(\exists x)(\exists y)((x^2 + y^2 - 1 = 0) \wedge (y - 2 = 0))$
- Geometry
  e.g. The emptiness of the (semi-)algebraic set $V(x^2 + y^2 - 1, y - 2)$
- Algebra (Real Nullstellensatz, Positivstellensatz)
  e.g. $1 + \frac{1}{3}x^2 = \frac{1}{3}(x^2 + y^2 - 1) - \frac{1}{3}(y + 2)(y - 2)$

Decision Problem over the Reals

This connects very different areas of mathematics.

- Logic
  e.g. $(\exists x)(\exists y)((x^2+y^2-1=0)\wedge(y-2=0))$
- Geometry
  e.g. The emptiness of the (semi-)algebraic set $V(x^2+y^2-1, y-2)$
- Algebra (Real Nullstellensatz, Positivstellensatz)
  e.g. $1+\frac{1}{3}x^2=\frac{1}{3}(x^2+y^2-1)-\frac{1}{3}(y+2)(y-2)$
- Computation
  Certainly we need an algorithm!

Decision Problem over the Reals

- Tarski (1951) has shown that this problem is decidable.

Decision Problem over the Reals

- Tarski (1951) has shown that this problem is decidable.
- Why are we still interested in this problem?

Decision Problem over the Reals

- Tarski (1951) has shown that this problem is decidable.
- Why are we still interested in this problem?
- Performance!

Decision Problem over the Reals

- Tarski (1951) has shown that this problem is decidable.
- Why are we still interested in this problem?
- Performance!
- Tarski's result has a non-elementary complexity, which makes it useless in practical computation.

Decision Problem over the Reals

- Tarski (1951) has shown that this problem is decidable.
- Why are we still interested in this problem?
- Performance!
- Tarski's result has a non-elementary complexity, which makes it useless in practical computation.
- Collins (1975) proposed Cylindrical Algebraic Decomposition (CAD), which has been shown to be the most efficient decision algorithm so far.

A Glimpse of Cylindrical Algebraic Decomposition

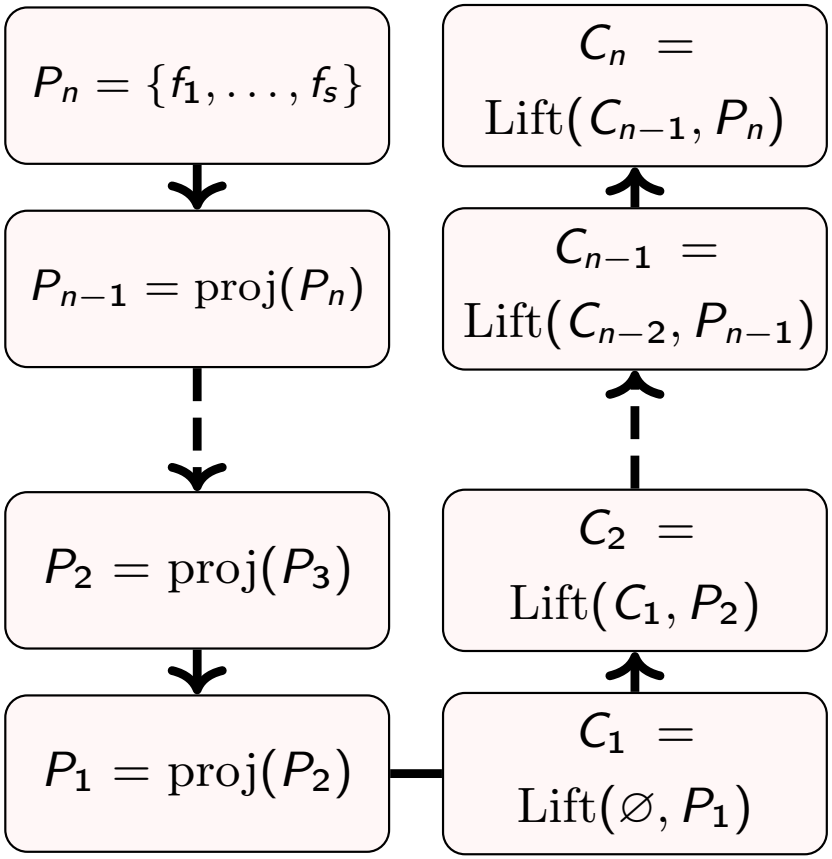


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.

A Glimpse of Cylindrical Algebraic Decomposition

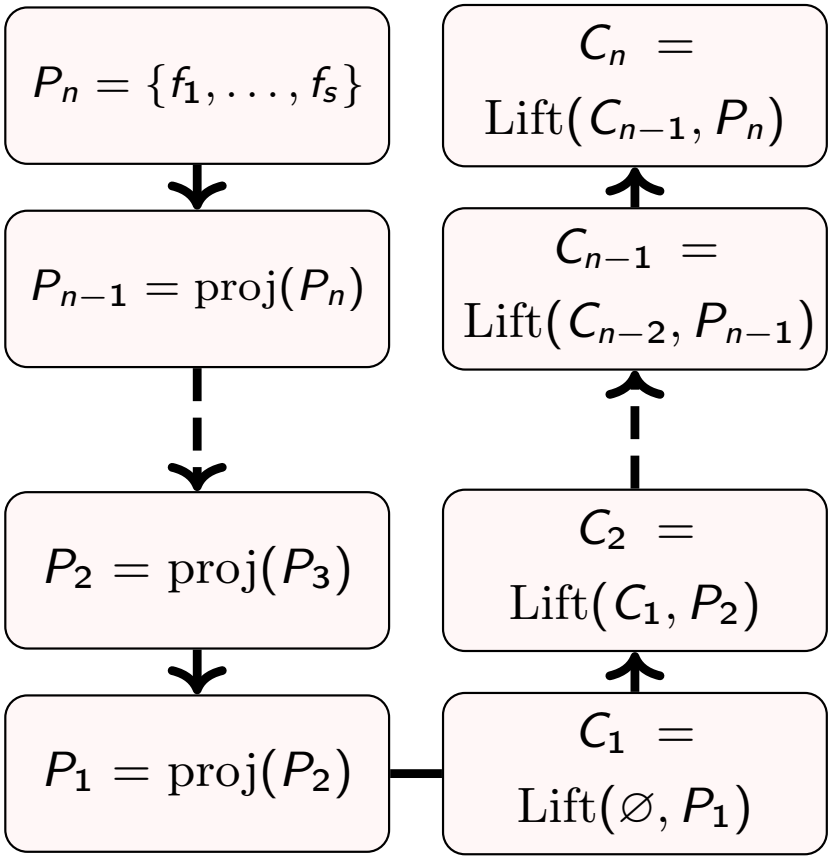


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.
- In the projection phase, the algorithm repeatedly "projects" polynomials.

A Glimpse of Cylindrical Algebraic Decomposition

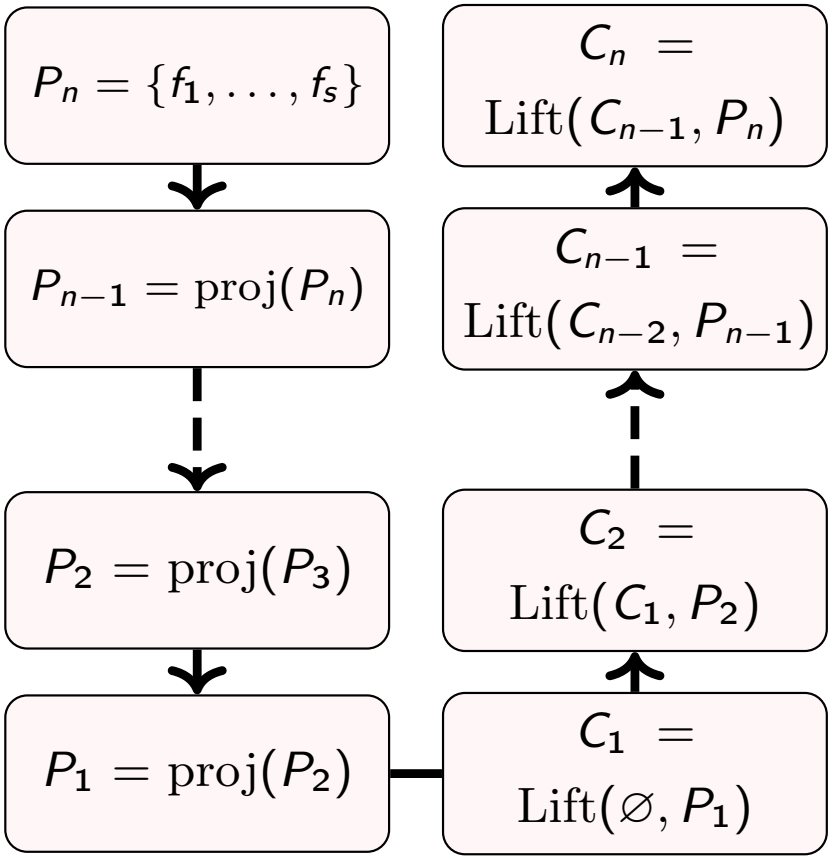


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.
- In the projection phase, the algorithm repeatedly "projects" polynomials.
  - Compute a family of $(n-1)$-variate polynomials P_{n-1} from the input P_n.

A Glimpse of Cylindrical Algebraic Decomposition

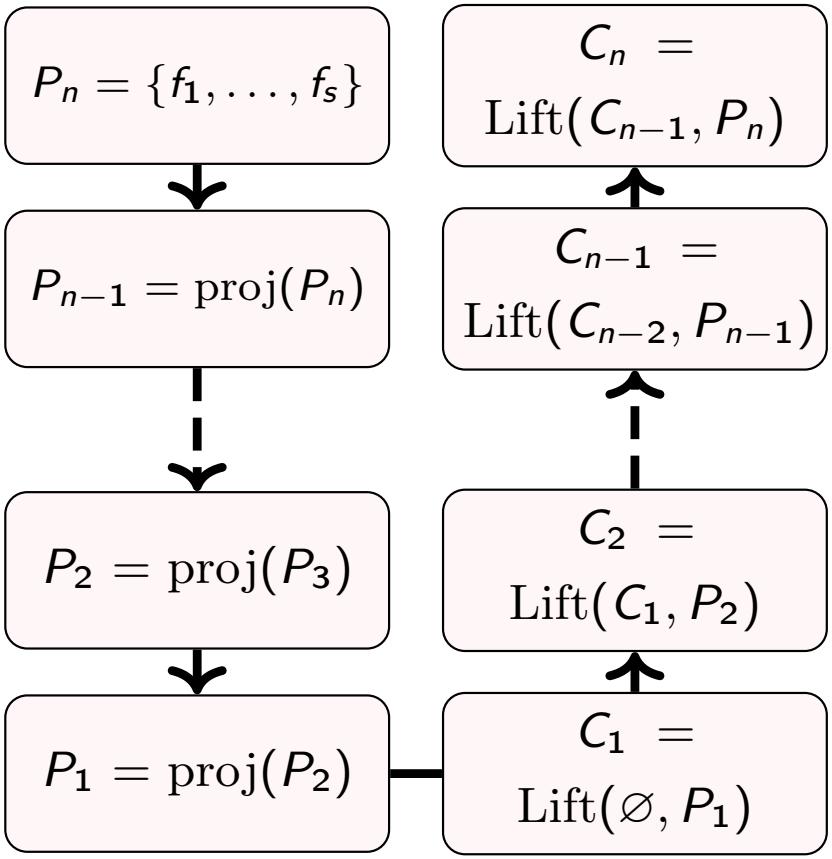


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.
- In the projection phase, the algorithm repeatedly "projects" polynomials.
  - Compute a family of $(n-1)$-variate polynomials P_{n-1} from the input P_n.
  - Then a family of $(n-2)$-variate polynomials P_{n-2} from the previous result P_{n-1}.

A Glimpse of Cylindrical Algebraic Decomposition

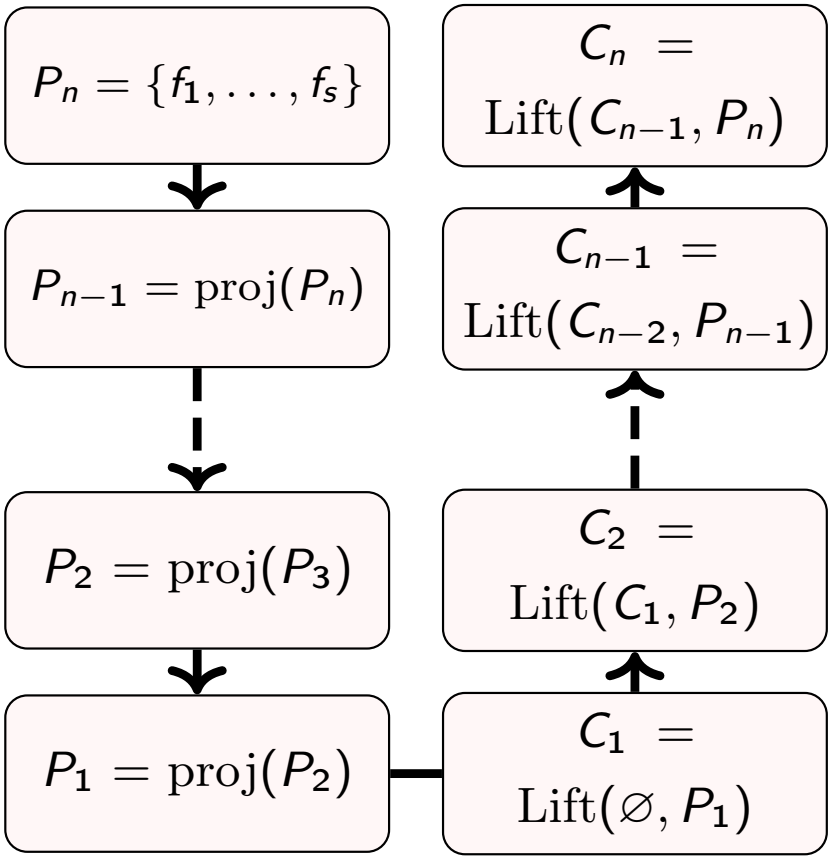


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.
- In the projection phase, the algorithm repeatedly "projects" polynomials.
  - Compute a family of $(n-1)$-variate polynomials P_{n-1} from the input P_n.
  - Then a family of $(n-2)$-variate polynomials P_{n-2} from the previous result P_{n-1}.
  - ....

A Glimpse of Cylindrical Algebraic Decomposition

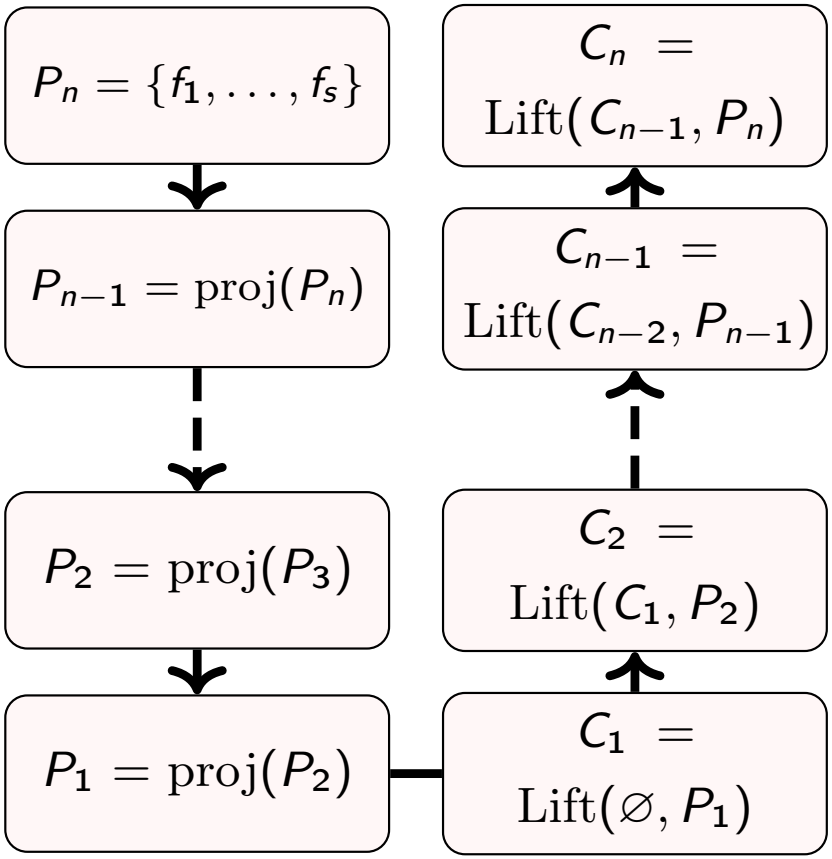


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.
- In the lifting phase, the algorithm successively constructs sign-invariant cells for $P_1, P_2, \ldots, P_n$.

A Glimpse of Cylindrical Algebraic Decomposition

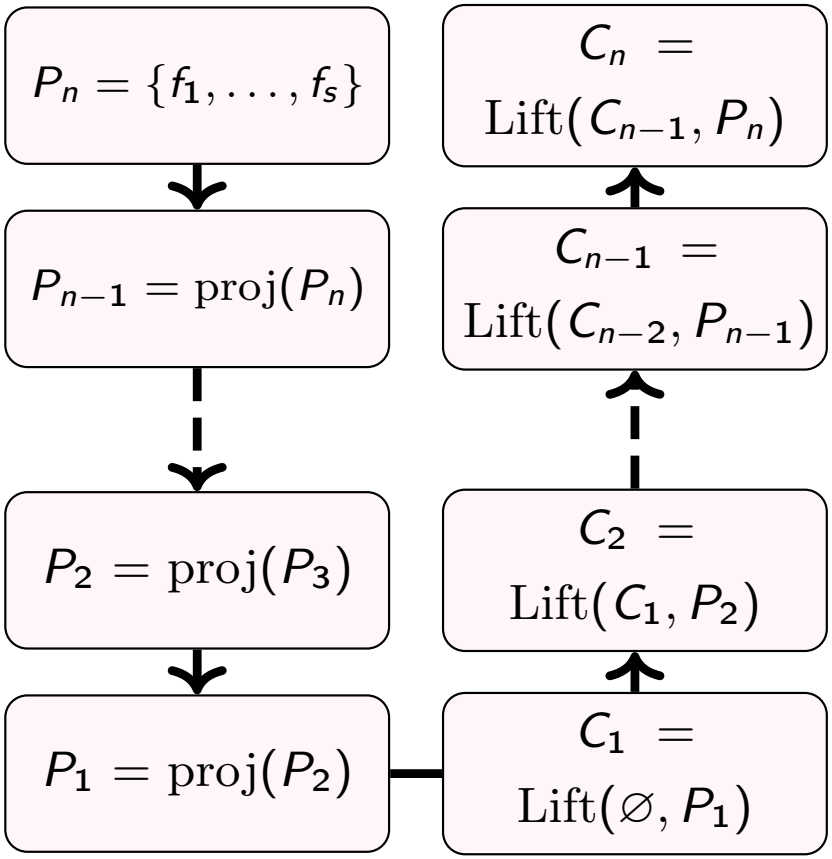


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.
- In the lifting phase, the algorithm successively constructs sign-invariant cells for $P_1, P_2, \ldots, P_n$.
  - Construct sign-invariant regions C_1 in $\mathbb{R}^1$ for P_1,

A Glimpse of Cylindrical Algebraic Decomposition

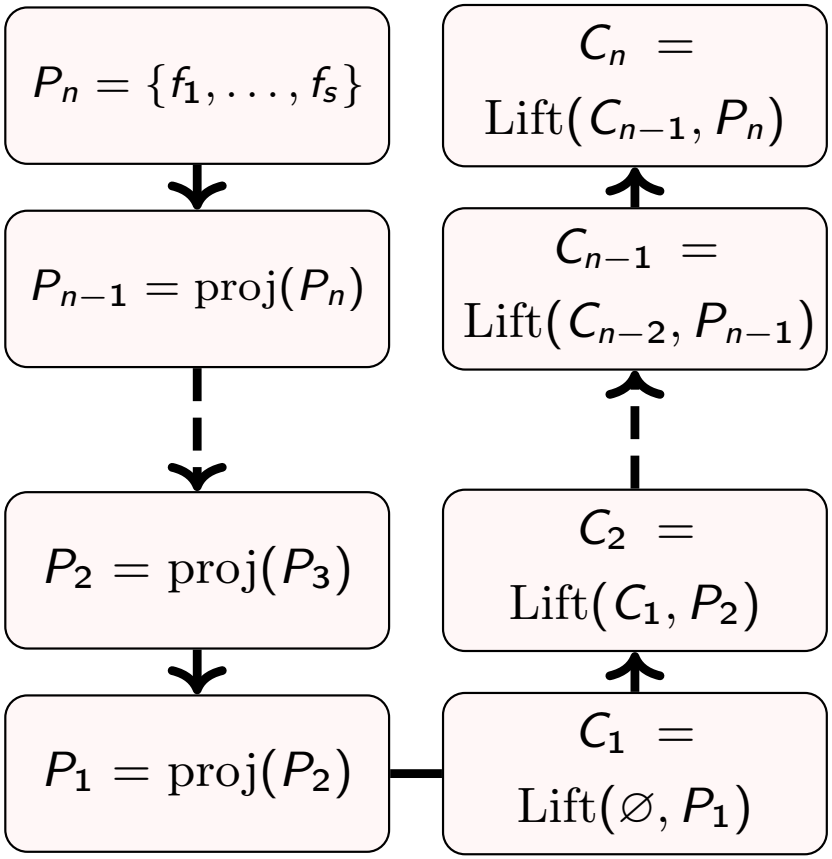


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.
- In the lifting phase, the algorithm successively constructs sign-invariant cells for $P_1, P_2, \ldots, P_n$.
  - Construct sign-invariant regions C_1 in $\mathbb{R}^1$ for P_1,
  - Then lift to sign-invariant regions C_2 in $\mathbb{R}^2$ for P_2,

A Glimpse of Cylindrical Algebraic Decomposition

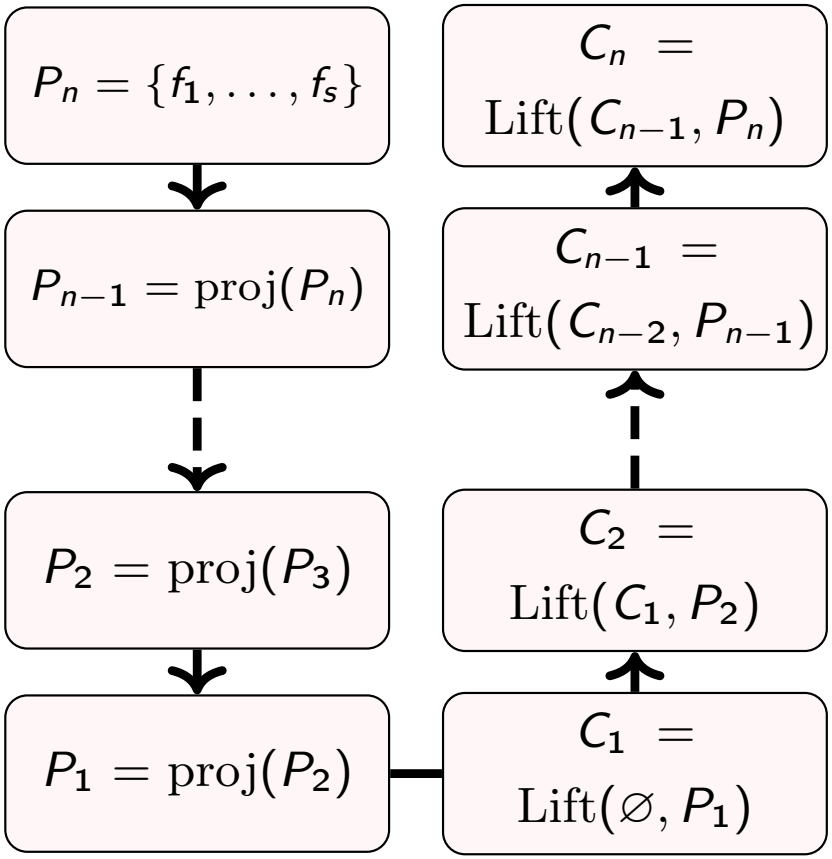


Figure 2: The workflow of CAD

- The classical CAD consists of two phases: Projection Phase and Lifting Phase.
- In the lifting phase, the algorithm successively constructs sign-invariant cells for $P_1, P_2, \ldots, P_n$.
  - Construct sign-invariant regions C_1 in $\mathbb{R}^1$ for P_1,
  - Then lift to sign-invariant regions C_2 in $\mathbb{R}^2$ for P_2,
  - ....

A Glimpse of Cylindrical Algebraic Decomposition

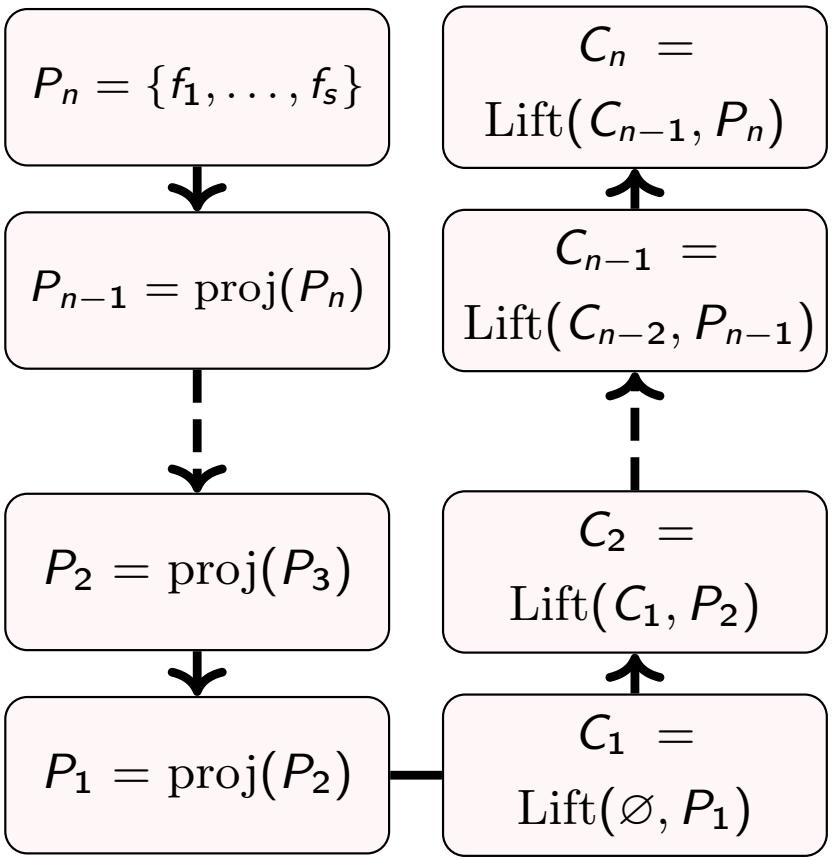


Figure 2: The workflow of CAD

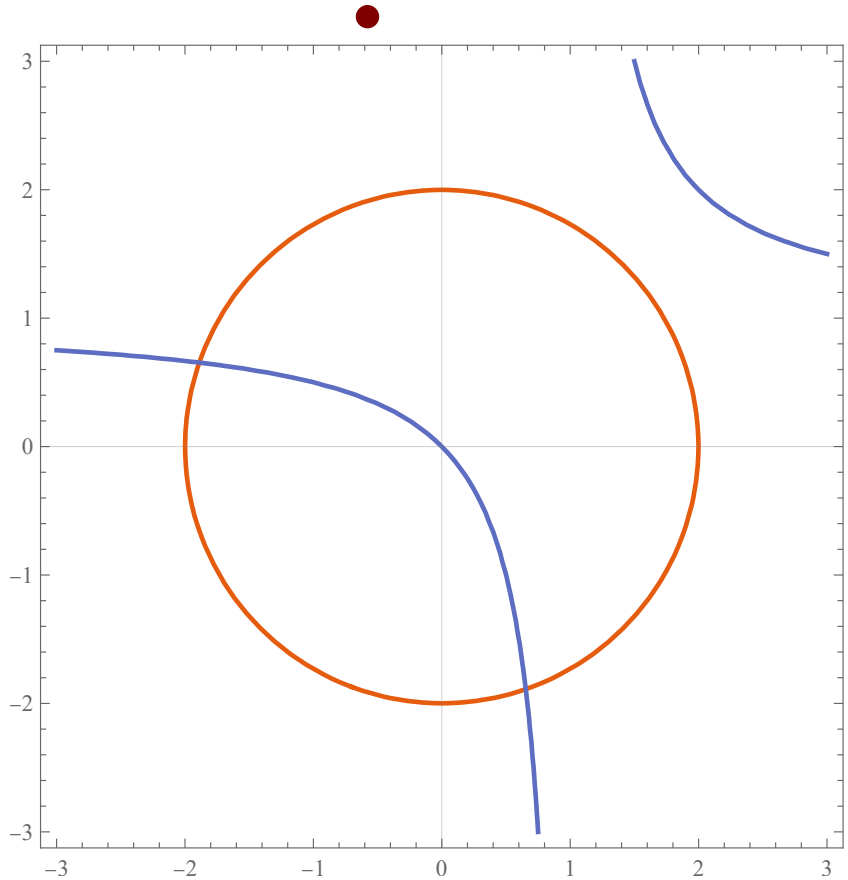


Figure 3: An Example Input

A Glimpse of Cylindrical Algebraic Decomposition

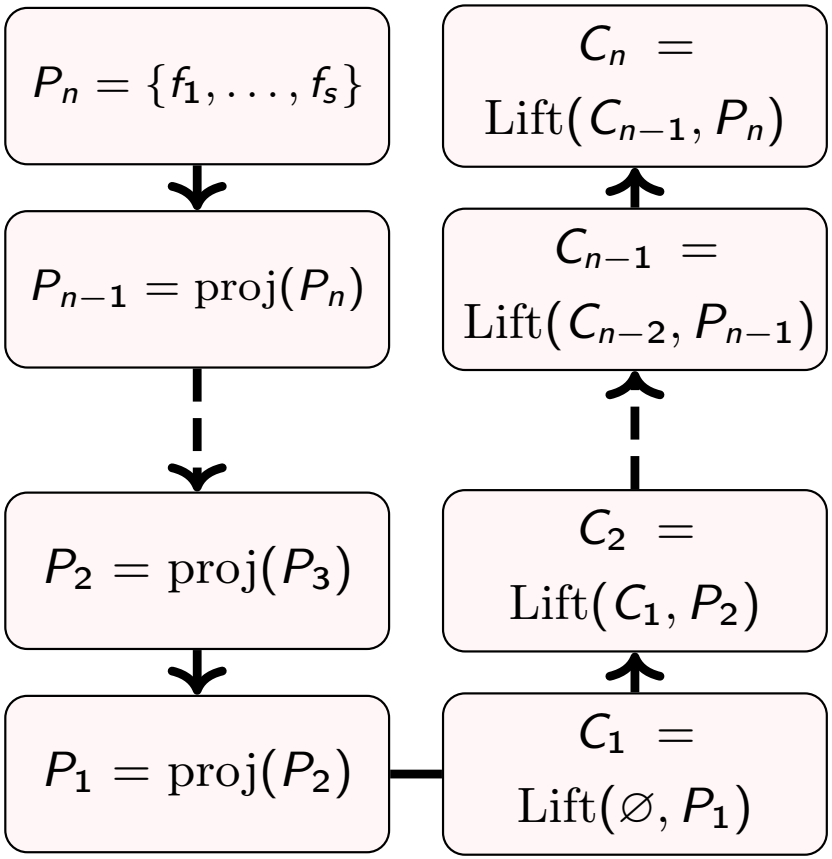


Figure 2: The workflow of CAD

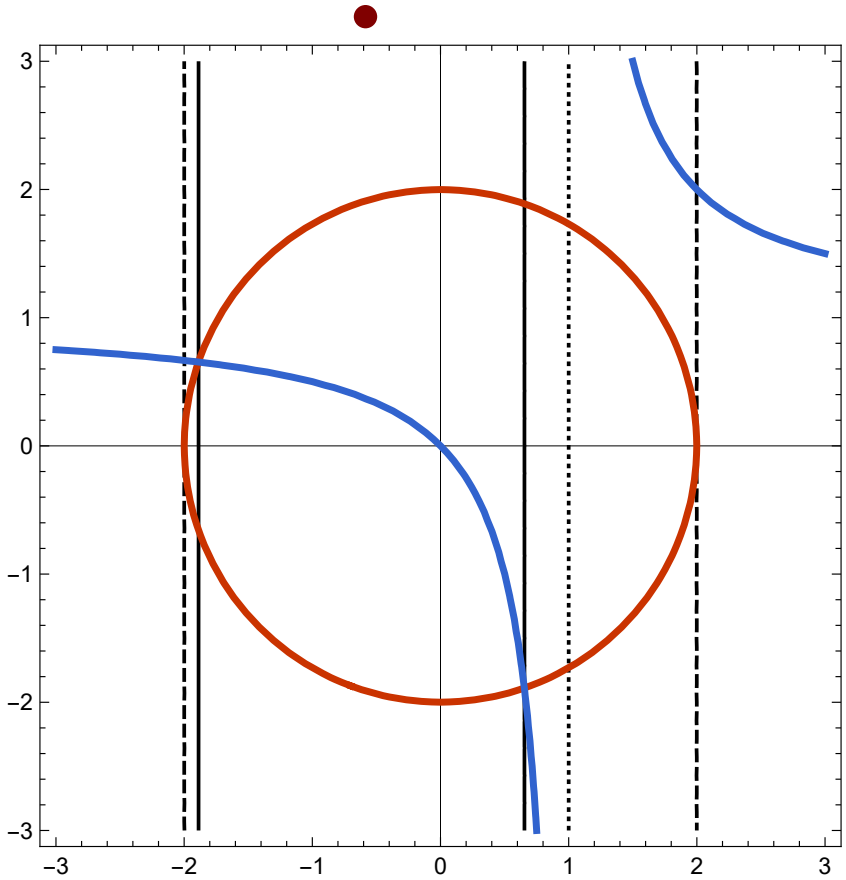


Figure 3: Projection Phase

A Glimpse of Cylindrical Algebraic Decomposition

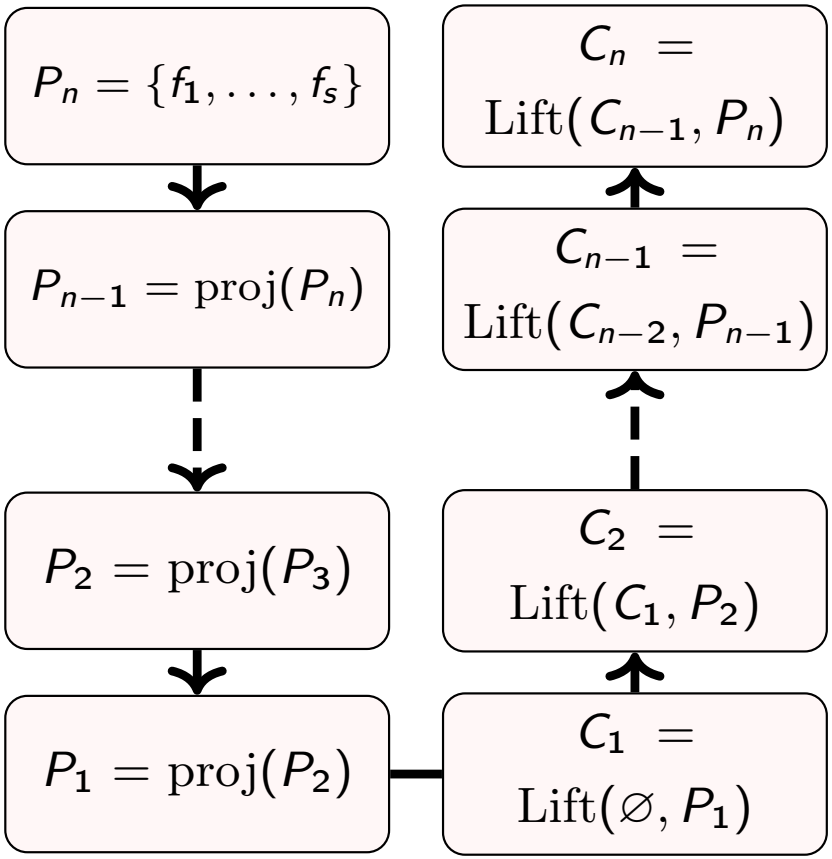


Figure 2: The workflow of CAD

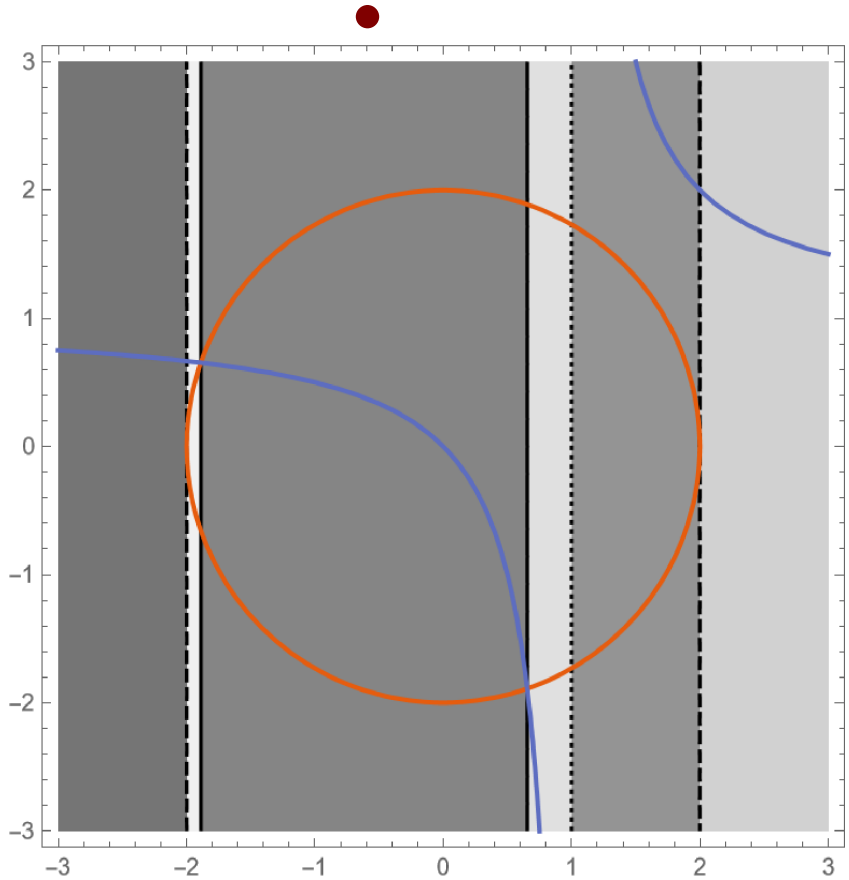


Figure 3: P_1-sign-invariant regions

A Glimpse of Cylindrical Algebraic Decomposition

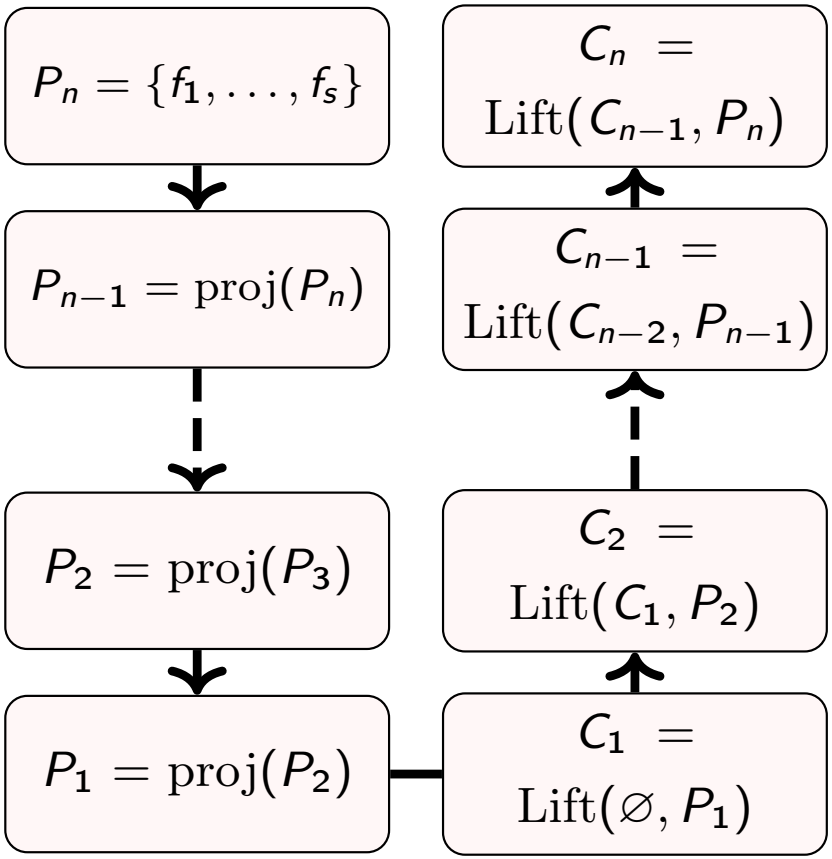


Figure 2: The workflow of CAD

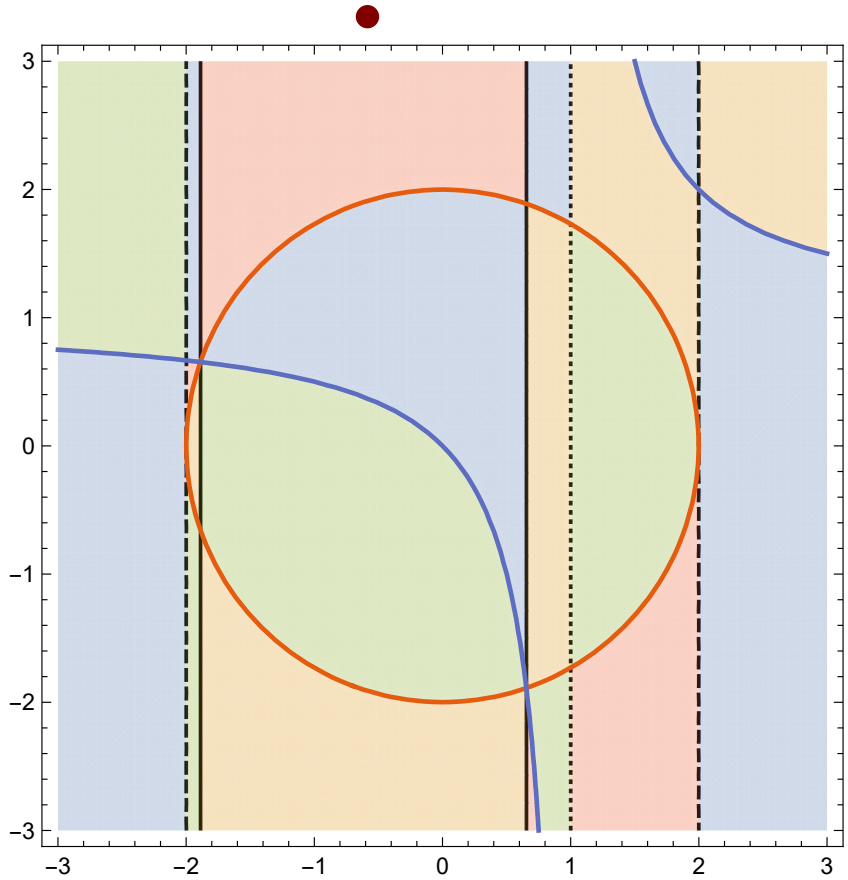


Figure 3: P_2-sign-invariant regions

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"
- One answer was, "it means eliminating variables".

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"
- One answer was, "it means eliminating variables".
- I did not find the answer satisfying.

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"
- One answer was, "it means eliminating variables".
- I did not find the answer satisfying.
- Collins-Hong projection operator $\mathrm{proj}(P)$ includes three types of polynomials:

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"
- One answer was, "it means eliminating variables".
- I did not find the answer satisfying.
- Collins-Hong projection operator $\mathrm{proj}(P)$ includes three types of polynomials:
  1. Leading coefficients of reducta of $f \in P$,

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"
- One answer was, "it means eliminating variables".
- I did not find the answer satisfying.
- Collins-Hong projection operator $\mathrm{proj}(P)$ includes three types of polynomials:
  1. Leading coefficients of reducta of $f \in P$,
  2. Sub-discriminants of reducta of $f \in P$,

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"
- One answer was, "it means eliminating variables".
- I did not find the answer satisfying.
- Collins-Hong projection operator $\mathrm{proj}(P)$ includes three types of polynomials:
  1. Leading coefficients of reducta of $f \in P$,
  2. Sub-discriminants of reducta of $f \in P$,
  3. Pairwise subresultants of reducta of $f, g \in P$.

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"
- One answer was, "it means eliminating variables".
- I did not find the answer satisfying.
- Collins-Hong projection operator $\mathrm{proj}(P)$ includes three types of polynomials:
  1. Leading coefficients of reducta of $f \in P$,
  2. Sub-discriminants of reducta of $f \in P$,
  3. Pairwise subresultants of reducta of $f, g \in P$.
- "Resultant eliminates a variable." This is fine.

Interlude: a Question that Haunts me

- When I was a naïve undergraduate, I asked the following question:
- "What do you mean by 'projecting' polynomials?"
- One answer was, "it means eliminating variables".
- I did not find the answer satisfying.
- Collins-Hong projection operator $\mathrm{proj}(P)$ includes three types of polynomials:
  1. Leading coefficients of reducta of $f \in P$,
  2. Sub-discriminants of reducta of $f \in P$,
  3. Pairwise subresultants of reducta of $f, g \in P$.
- "Resultant eliminates a variable." This is fine.
- "Leading coefficient or discriminant eliminates a variable." ?????

The Challenge from Equations

- It was soon noticed that some information can be used to accelerate the computation of c.a.d., especially the equations.

The Challenge from Equations

- It was soon noticed that some information can be used to accelerate the computation of c.a.d., especially the equations.
- For some applications, there are already equations in the input (e.g. Real Root Classification, Automatic Theorem Proving).

The Challenge from Equations

- It was soon noticed that some information can be used to accelerate the computation of c.a.d., especially the equations.
- For some applications, there are already equations in the input (e.g. Real Root Classification, Automatic Theorem Proving).
- Also, equations are generated during the projection phase.

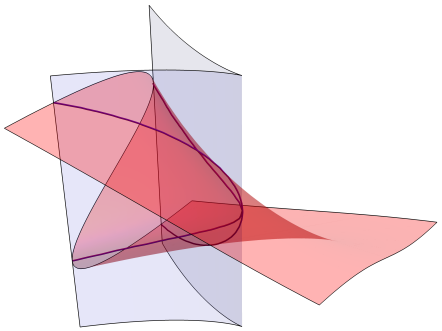
The Challenge from Equations

- It was soon noticed that some information can be used to accelerate the computation of c.a.d., especially the equations.
- For some applications, there are already equations in the input (e.g. Real Root Classification, Automatic Theorem Proving).
- Also, equations are generated during the projection phase.
- To see this, let us consider a toy polynomial system

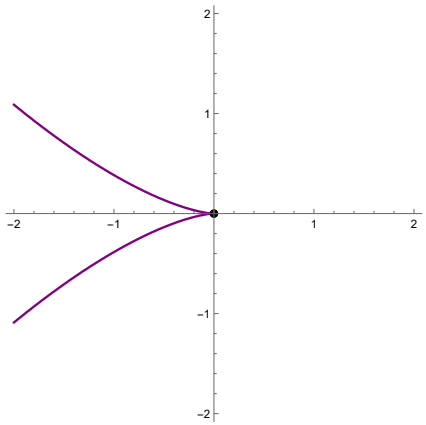
$$x^3 + px + q = 4p^3 + 27q^2 = 0$$

in parameters p, q and variable x.

The Challenge from Equations



(a) The solution set

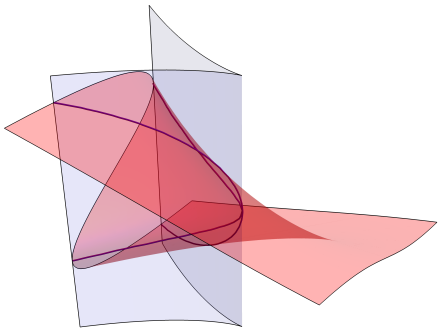


(b) The projection

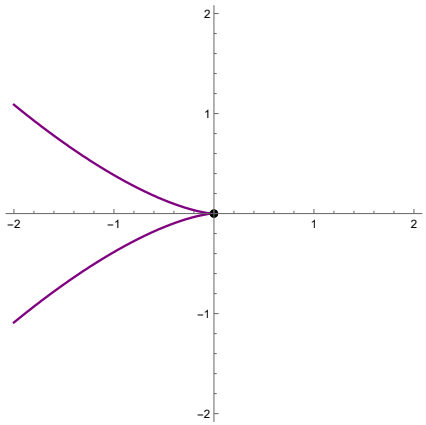
Figure 4: Equations arise

- The system $x^3 + px + q = 4p^3 + 27q^2 = 0$ defines a curve consisting of two irreducible components, see the purple curve on the upper left.

The Challenge from Equations



(a) The solution set

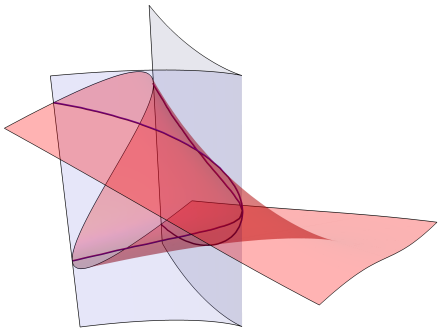


(b) The projection

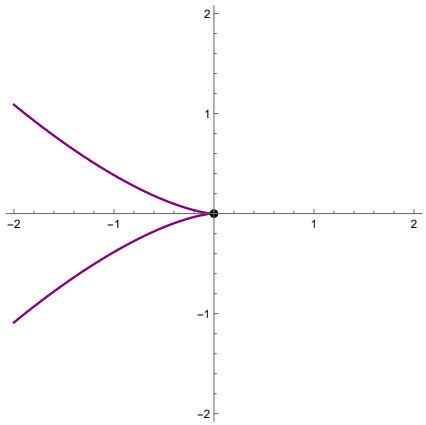
Figure 4: Equations arise

- The system $x^3 + px + q = 4p^3 + 27q^2 = 0$ defines a curve consisting of two irreducible components, see the purple curve on the upper left.
- These two irreducible components intersect at the origin.

The Challenge from Equations



(a) The solution set

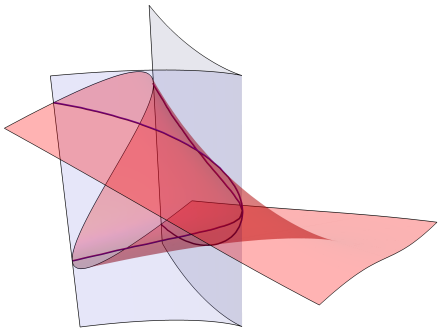


(b) The projection

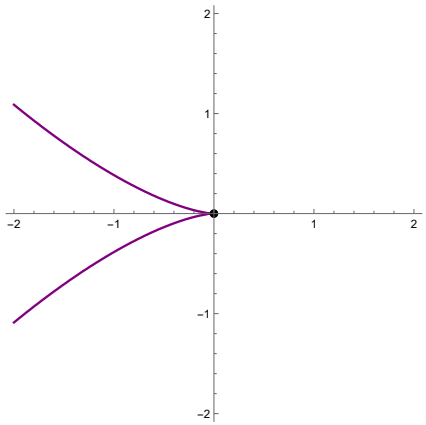
Figure 4: Equations arise

- The system $x^3 + px + q = 4p^3 + 27q^2 = 0$ defines a curve consisting of two irreducible components, see the purple curve on the upper left.
- These two irreducible components intersect at the origin.
- So extra equations $p = q = 0$ are generated during the projection.

The Challenge from Equations



(a) The solution set

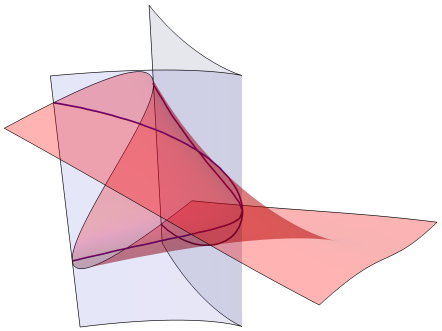


(b) The projection

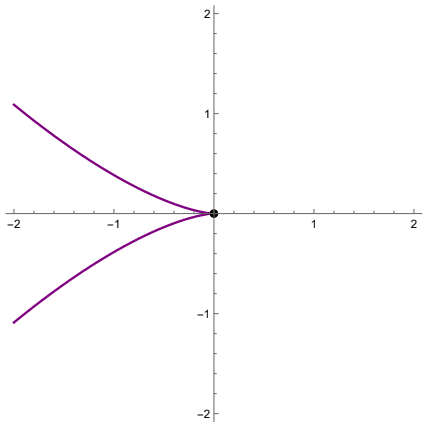
Figure 4: Equations arise

- The system $x^3 + px + q = 4p^3 + 27q^2 = 0$ defines a curve consisting of two irreducible components, see the purple curve on the upper left.
- These two irreducible components intersect at the origin.
- So extra equations $p = q = 0$ are generated during the projection.
- For larger non-trivial inputs, this phenomenon becomes much more frequent.

The Challenge from Equations



(a) The solution set

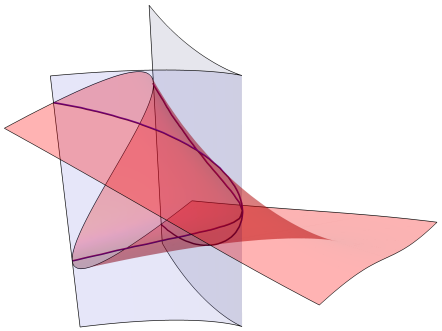


(b) The projection

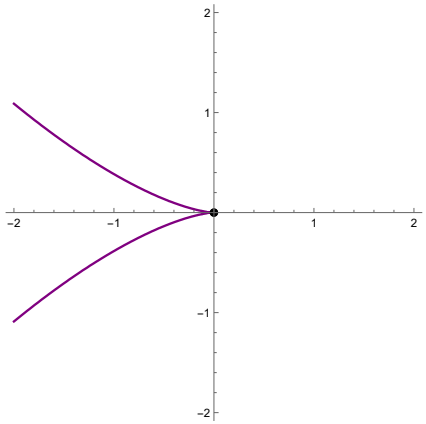
Figure 3: Equations arise

- For larger non-trivial inputs, this phenomenon becomes much more frequent.

The Challenge from Equations



(a) The solution set

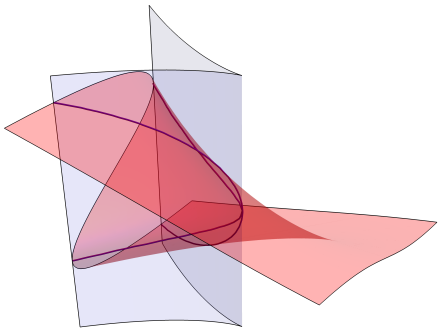


(b) The projection

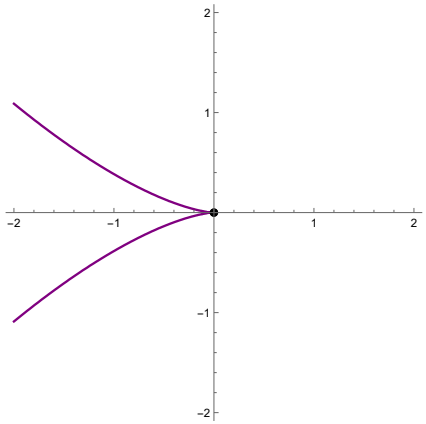
Figure 3: Equations arise

- For larger non-trivial inputs, this phenomenon becomes much more frequent.
- There are already many equations in the input and many additional equations are systematically generated during the projection phase.

The Challenge from Equations



(a) The solution set



(b) The projection

Figure 3: Equations arise

- For larger non-trivial inputs, this phenomenon becomes much more frequent.
- There are already many equations in the input and many additional equations are systematically generated during the projection phase.
- Therefore it is an important challenge to exploit these equations.

Previous Results

There has been intensive efforts regarding this issue.

- McCallum (1999, 2001) proposed restricted projection operators to utilize the equations in the input.

Previous Results

There has been intensive efforts regarding this issue.

- McCallum (1999, 2001) proposed restricted projection operators to utilize the equations in the input.
- The main idea is to

Previous Results

There has been intensive efforts regarding this issue.

- McCallum (1999, 2001) proposed restricted projection operators to utilize the equations in the input.
- The main idea is to
  - select an equation (if there are any) at each stage of projection,

Previous Results

There has been intensive efforts regarding this issue.

- McCallum (1999, 2001) proposed restricted projection operators to utilize the equations in the input.
- The main idea is to
  - select an equation (if there are any) at each stage of projection,
  - decompose the hypersurface defined by the equation only,

Previous Results

There has been intensive efforts regarding this issue.

- McCallum (1999, 2001) proposed restricted projection operators to utilize the equations in the input.
- The main idea is to
  - select an equation (if there are any) at each stage of projection,
  - decompose the hypersurface defined by the equation only,
  - discover implicit equations by successive resultants of known equations.

Previous Results

There has been intensive efforts regarding this issue.

- McCallum (1999, 2001) proposed restricted projection operators to utilize the equations in the input.
- The main idea is to
  - select an equation (if there are any) at each stage of projection,
  - decompose the hypersurface defined by the equation only,
  - discover implicit equations by successive resultants of known equations.
- England, Bradford and Davenport (2015) found that savings can be made during the lifting phase.

Still, more to be done...

- The previous results did not make full use of the equations.

Still, more to be done...

- The previous results did not make full use of the equations.
  - Only one equation is used as "pivot" in each stage of projection.

Still, more to be done...

- The previous results did not make full use of the equations.
  - Only one equation is used as "pivot" in each stage of projection.
  - And they ignored many equations emerging in the middle of projection.

Still, more to be done...

- The previous results did not make full use of the equations.
  - Only one equation is used as "pivot" in each stage of projection.
  - And they ignored many equations emerging in the middle of projection.
- McCallum (2001) works with irreducible polynomials,

Still, more to be done...

- The previous results did not make full use of the equations.
  - Only one equation is used as "pivot" in each stage of projection.
  - And they ignored many equations emerging in the middle of projection.
- McCallum (2001) works with irreducible polynomials,
  - a complicated preprocessing needs to be done before applying their result.

Still, more to be done...

- The previous results did not make full use of the equations.
  - Only one equation is used as "pivot" in each stage of projection.
  - And they ignored many equations emerging in the middle of projection.
- McCallum (2001) works with irreducible polynomials,
  - a complicated preprocessing needs to be done before applying their result.
- Their theory is based on McCallum (1998), the McCallum projection operator.

Still, more to be done...

- The previous results did not make full use of the equations.
  - Only one equation is used as "pivot" in each stage of projection.
  - And they ignored many equations emerging in the middle of projection.
- McCallum (2001) works with irreducible polynomials,
  - a complicated preprocessing needs to be done before applying their result.
- Their theory is based on McCallum (1998), the McCallum projection operator.
  - Its correctness is conditional only.

Still, more to be done...

- The previous results did not make full use of the equations.
  - Only one equation is used as "pivot" in each stage of projection.
  - And they ignored many equations emerging in the middle of projection.
- McCallum (2001) works with irreducible polynomials,
  - a complicated preprocessing needs to be done before applying their result.
- Their theory is based on McCallum (1998), the McCallum projection operator.
  - Its correctness is conditional only.
  - When the projection is not quasi-finite (a rare case), their algorithms would raise an error (the "nullification problem").

A new simple algorithm

We develop a new conceptually simple and efficient algorithm that exploits all the equations systematically in the recent paper

R. Chen, *A geometric approach to cylindrical algebraic decomposition*, Math. Comp. **95** (2026), pp. 2025-2059.

The big idea

- We do not "project" polynomials anymore. We will project varieties.

The big idea

- We do not "project" polynomials anymore. We will project varieties.
- Very roughly speaking, our algorithm starts with some varieties.

The big idea

- We do not "project" polynomials anymore. We will project varieties.
- Very roughly speaking, our algorithm starts with some varieties.
- Then we stratify their projection images into

pieces.

The big idea

- We do not "project" polynomials anymore. We will project varieties.
- Very roughly speaking, our algorithm starts with some varieties.
- Then we stratify their projection images into
  - "flat" ($\approx$ fibers changes continuously) and

  pieces.

Flatness

- Flatness is a kind of continuity in algebraic geometry.

Flatness

- Flatness is a kind of continuity in algebraic geometry.
- Roughly speaking, a parametric system is flat, if the variables change continuously w.r.t. the parameters.

Flatness

- Flatness is a kind of continuity in algebraic geometry.
- Roughly speaking, a parametric system is flat, if the variables change continuously w.r.t. the parameters.
- Example: $x^2 + bx + c = 0$. The unknown x is continuous w.r.t. the parameters b, c (quadratic formula).

Flatness

- Flatness is a kind of continuity in algebraic geometry.
- Roughly speaking, a parametric system is flat, if the variables change continuously w.r.t. the parameters.
- Example: $x^2 + bx + c = 0$. The unknown x is continuous w.r.t. the parameters b, c (quadratic formula).
- Non-example: $ax^2 + bx + c = 0$. Because a might be 0, $\frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ is not continuous near $a = 0$.

The big idea

- We do not "project" polynomials anymore. We will project varieties.
- Very roughly speaking, our algorithm starts with some varieties.
- Then we stratify their projection images into
  - "flat" ($\approx$ fibers changes continuously) and
  - "unramified" ($\approx$ fibers do not break into different branches)

  pieces.

Unramified

- Roughly speaking, a parametric system is unramified, if the number of distinct **complex** solutions is a constant, regardless of the parameter.

Unramified

- Roughly speaking, a parametric system is unramified, if the number of distinct **complex** solutions is a constant, regardless of the parameter.
- Example: $(x-t)^2-1=0$. There are always two distinct solutions $x=t+1$ and $x=t-1$.

Unramified

- Roughly speaking, a parametric system is unramified, if the number of distinct **complex** solutions is a constant, regardless of the parameter.
- Example: $(x-t)^2-1=0$. There are always two distinct solutions $x=t+1$ and $x=t-1$.
- Example: $(x-t)^2=0$. There is always a double root $x=t$.

Unramified

- Roughly speaking, a parametric system is unramified, if the number of distinct **complex** solutions is a constant, regardless of the parameter.
- Example: $(x-t)^2-1=0$. There are always two distinct solutions $x=t+1$ and $x=t-1$.
- Example: $(x-t)^2=0$. There is always a double root $x=t$.
- Non-example: $x^2-t=0$. Generically there are two distinct solutions, except when $t=0$.

Remarks

Although we use the word "unramified" differently from scheme theory, the difference is minor.

- (Scheme-theoretic) $\Omega_{X/Y} = 0$.

Remarks

Although we use the word "unramified" differently from scheme theory, the difference is minor.

- (Scheme-theoretic) $\Omega_{X/Y} = 0$.
- (Ours) $n(y) := \#X_{\overline{y}}$ (locally) constant.

Remarks

Although we use the word "unramified" differently from scheme theory, the difference is minor.

- (Scheme-theoretic) $\Omega_{X/Y} = 0$.
- (Ours) $n(y) := \#X_{\overline{y}}$ (locally) constant.
- For a finite, flat morphism $\pi : X \to Y$ with $\Omega_{X/Y} = 0$, $n(y)$ is locally constant.

Remarks

Although we use the word "unramified" differently from scheme theory, the difference is minor.

- (Scheme-theoretic) $\Omega_{X/Y} = 0$.
- (Ours) $n(y) := \#X_{\overline{y}}$ (locally) constant.
- For a finite, flat morphism $\pi : X \to Y$ with $\Omega_{X/Y} = 0$, $n(y)$ is locally constant.
- Conversely, one can show that for a quasi-finite flat morphism $\pi : X \to Y$ with locally constant $n(y)$, the reduced map $X_{\mathrm{red}} \to Y_{\mathrm{red}}$ is finite, étale, hence $\Omega_{X_{\mathrm{red}}/Y_{\mathrm{red}}} = 0$.

Convert to an algorithm

- In general, given a variety $X \subseteq \mathbb{A}_R^n$, its projection downto $\mathbb{A}_R^{n-1}$ is neither "flat" nor "unramified".

Convert to an algorithm

- In general, given a variety $X \subseteq \mathbb{A}_R^n$, its projection downto $\mathbb{A}_R^{n-1}$ is neither "flat" nor "unramified".
- An effective version of Grothendieck's generic freeness remedies this: the projection can be partitioned into finitely many pieces such that the projection is "flat".

Effective generic freeness

- The generic freeness lemma is an important tool in algebraic geometry due to Grothendieck.

Effective generic freeness

- The generic freeness lemma is an important tool in algebraic geometry due to Grothendieck.
- The theory of Gröbner basis gives it a constructive proof.

Folklore. Effective Generic Freeness

Let A be a k-algebra of finite type, $B = A[x_1, \ldots, x_n]/I$ is an A-algebra of finite type. Let $\mathcal{G}$ be a Gröbner basis of I, over A. Set $w = \prod_{f \in \mathcal{G}} \mathrm{LC}(g) \in A$, then B_w is a free A_w-module.

Effective generic freeness

- The generic freeness lemma is an important tool in algebraic geometry due to Grothendieck.
- The theory of Gröbner basis gives it a constructive proof.

Folklore. Effective Generic Freeness

Let A be a k-algebra of finite type, $B = A[x_1, \ldots, x_n]/I$ is an A-algebra of finite type. Let $\mathcal{G}$ be a Gröbner basis of I, over A. Set $w = \prod_{f \in \mathcal{G}} \mathrm{LC}(g) \in A$, then B_w is a free A_w-module.

- Hence, after inverting an element $w \in A$ (localizing), the projection from B_w to A_w is flat.

Effective generic freeness

- The generic freeness lemma is an important tool in algebraic geometry due to Grothendieck.
- The theory of Gröbner basis gives it a constructive proof.

Folklore. Effective Generic Freeness

Let A be a k-algebra of finite type, $B = A[x_1, \ldots, x_n]/I$ is an A-algebra of finite type. Let $\mathcal{G}$ be a Gröbner basis of I, over A. Set $w = \prod_{f \in \mathcal{G}} \mathrm{LC}(g) \in A$, then B_w is a free A_w-module.

- Hence, after inverting an element $w \in A$ (localizing), the projection from B_w to A_w is flat.
- Applying effective generic freeness recursively, we can stratify the projection image into "flat" pieces.

Convert to an algorithm

- In general, given a variety $X \subseteq \mathbb{A}_R^n$, its projection downto $\mathbb{A}_R^{n-1}$ is neither "flat" nor "unramified".
- An effective version of Grothendieck's generic freeness remedies this: the projection can be partitioned into finitely many pieces such that the projection is "flat".
- Then on each piece, a trace pairing technique (Hermite's Quadratic Form) can be used to count the complex solutions, so each piece is further stratified into "unramified" pieces.

Hermite's Quadratic Form

- Suppose $A = k[x_1, \ldots, x_n]/I$ is a zero-dimensional k-algebra ($\operatorname{char} k = 0$). The Hermite's Quadratic Form is the trace pairing

$$\begin{array}{rcl} H : A \times A & \to & k \\ (f, g) & \mapsto & \operatorname{tr} L_{fg} \end{array},$$

where L_{fg} is the k-linear multiplication map $A \to A, a \mapsto fg \cdot a$.

Hermite's Quadratic Form

- Suppose $A = k[x_1, \ldots, x_n]/I$ is a zero-dimensional k-algebra ($\operatorname{char} k = 0$). The Hermite's Quadratic Form is the trace pairing

$$\begin{array}{rcl} H : A \times A & \to & k \\ (f, g) & \mapsto & \operatorname{tr} L_{fg} \end{array},$$

  where L_{fg} is the k-linear multiplication map $A \to A, a \mapsto fg \cdot a$.
- $\operatorname{rank} H = \#V_C(I)$, $\operatorname{sign} H = \#V_R(I)$.

Hermite's Quadratic Form

- Suppose $A = k[x_1, \ldots, x_n]/I$ is a zero-dimensional k-algebra ($\operatorname{char} k = 0$). The Hermite's Quadratic Form is the trace pairing

$$\begin{array}{rcl} H : A \times A & \to & k \\ (f, g) & \mapsto & \operatorname{tr} L_{fg} \end{array},$$

  where L_{fg} is the k-linear multiplication map $A \to A, a \mapsto fg \cdot a$.
- $\operatorname{rank} H = \# V_C(I)$, $\operatorname{sign} H = \# V_R(I)$.
- This can be generalized to the parametric case. If $\operatorname{rank} H$ is a constant, regardless of the parameters, then the system is "unramified".

Convert to an algorithm

- In general, given a variety $X \subseteq \mathbb{A}_R^n$, its projection downto $\mathbb{A}_R^{n-1}$ is neither "flat" nor "unramified".
- An effective version of Grothendieck's generic freeness remedies this: the projection can be partitioned into finitely many pieces such that the projection is "flat".
- Then on each piece, a trace pairing technique (Hermite's Quadratic Form) can be used to count the complex solutions, so each piece is further stratified into "unramified" pieces.
- Now we can conclude the existence of semi-algebraic sections on (connected components) of each stratum.

Convert to an algorithm

- In general, given a variety $X \subseteq \mathbb{A}_R^n$, its projection downto $\mathbb{A}_R^{n-1}$ is neither "flat" nor "unramified".
- An effective version of Grothendieck's generic freeness remedies this: the projection can be partitioned into finitely many pieces such that the projection is "flat".
- Then on each piece, a trace pairing technique (Hermite's Quadratic Form) can be used to count the complex solutions, so each piece is further stratified into "unramified" pieces.
- Now we can conclude the existence of semi-algebraic sections on (connected components) of each stratum.
- So a c.a.d. adapted to X can be lifted from a c.a.d. adapted to the strata.

Foundation Rebuilt

Theorem (Chen, 2026)

Suppose that

$$\pi : X \underset{\text{closed}}{\hookrightarrow} \mathbb{A}^1_Y \to Y$$

is finite free morphism between affine R-varieties.
If $S \subseteq Y(R)$ is a semi-algebraically connected semi-algebraic set on which $\#X_{\overline{y}}$ is a constant, then there are semi-algebraic continuous functions

$$\xi_1, \ldots, \xi_l : S \to X(R)$$

such that $\pi \circ \xi_i = \mathrm{id}_S$ and $\{\xi_i(y)\}_{i=1}^l$ is the real fiber of y in $X(R)$.

This is a non-trivial generalization of Collins (1975) in the algebro-geometric context.

The Meaning

- In short, this theorem says that for a "flat" and "unramified" parametric system, the real solutions are continuous functions in the parameters and the projection is a covering map on the real points.

The Meaning

- In short, this theorem says that for a "flat" and "unramified" parametric system, the real solutions are continuous functions in the parameters and the projection is a covering map on the real points.
- This is non-trivial because it links the number of complex solutions (geometric invariant) to being a covering on the real points (semi-algebraic property).

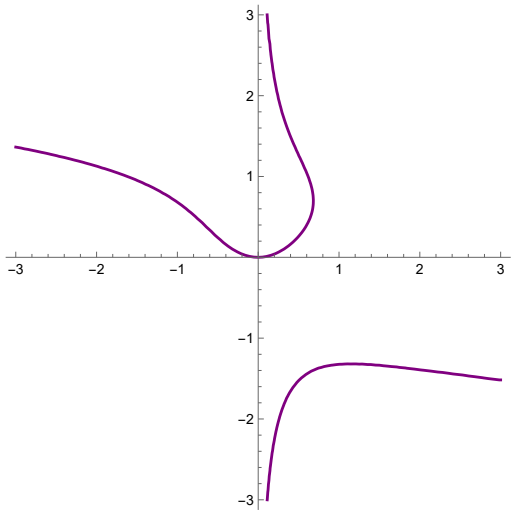
The Meaning

- In short, this theorem says that for a "flat" and "unramified" parametric system, the real solutions are continuous functions in the parameters and the projection is a covering map on the real points.
- This is non-trivial because it links the number of complex solutions (geometric invariant) to being a covering on the real points (semi-algebraic property).
- Therefore, it bridges complex algebraic geometry and real algebraic geometry.

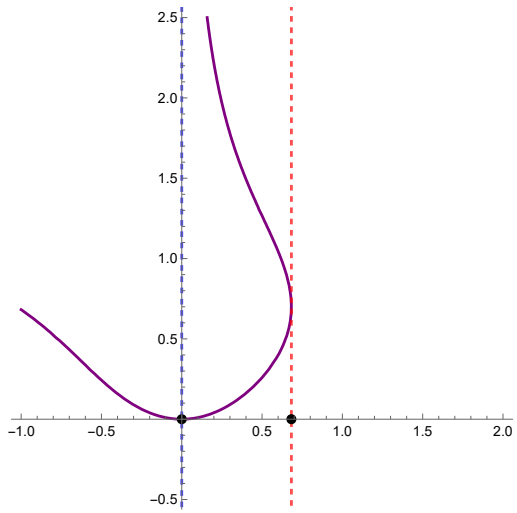
The Meaning

- In short, this theorem says that for a "flat" and "unramified" parametric system, the real solutions are continuous functions in the parameters and the projection is a covering map on the real points.
- This is non-trivial because it links the number of complex solutions (geometric invariant) to being a covering on the real points (semi-algebraic property).
- Therefore, it bridges complex algebraic geometry and real algebraic geometry.
- The shortest path between two truths in the real domain passes through the complex domain.

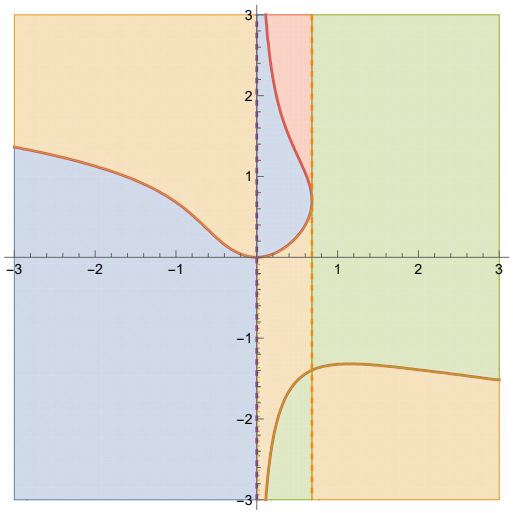
Illustration



(a) Locus of $f = xy^3 - y + x^2$



(b) Where "flatness" or "unramifiedness" fails



(c) The final c.a.d.

Figure 4: An example of c.a.d.

There is still a naïve undergraduate waiting for an answer

- Recall the Collins-Hong projection operator $\mathrm{proj}(P)$.

There is still a naïve undergraduate waiting for an answer

- Recall the Collins-Hong projection operator $\mathrm{proj}(P)$.
- It includes three types of polynomials:

There is still a naïve undergraduate waiting for an answer

- Recall the Collins-Hong projection operator $\mathrm{proj}(P)$.
- It includes three types of polynomials:
  1. Leading coefficients of reducta of $f \in P$,

There is still a naïve undergraduate waiting for an answer

- Recall the Collins-Hong projection operator $\mathrm{proj}(P)$.
- It includes three types of polynomials:
  1. Leading coefficients of reducta of $f \in P$,
  2. Sub-discriminants of reducta of $f \in P$,

There is still a naïve undergraduate waiting for an answer

- Recall the Collins-Hong projection operator $\mathrm{proj}(P)$.
- It includes three types of polynomials:
  1. Leading coefficients of reducta of $f \in P$,
  2. Sub-discriminants of reducta of $f \in P$,
  3. Pairwise subresultants of reducta of $f, g \in P$.

The answer

1. **Leading coefficients**
   This is to construct the generic freeness. Let $f \in A[x]$ and a is the leading coefficient of f, then $A_a[x]/\langle f \rangle$ is a free A_a-module of finite rank.

The answer

1. **Leading coefficients**
   This is to construct the generic freeness. Let $f \in A[x]$ and a is the leading coefficient of f, then $A_a[x]/\langle f \rangle$ is a free A_a-module of finite rank.
2. **Sub-discriminants**
   This is to count the geometric fibers of $\operatorname{Spec} A_a[x]/\langle f \rangle \to \operatorname{Spec} A_a$.

The answer

1. **Leading coefficients**
   This is to construct the generic freeness. Let $f \in A[x]$ and a is the leading coefficient of f, then $A_a[x]/\langle f \rangle$ is a free A_a-module of finite rank.
2. **Sub-discriminants**
   This is to count the geometric fibers of $\operatorname{Spec} A_a[x]/\langle f \rangle \to \operatorname{Spec} A_a$.
3. **Pairwise subresultants**
   This is to construct the generic freeness of $A[x]/\langle f, g \rangle$.

The answer

1. **Leading coefficients**
   This is to construct the generic freeness. Let $f \in A[x]$ and a is the leading coefficient of f, then $A_a[x]/\langle f \rangle$ is a free A_a-module of finite rank.
2. **Sub-discriminants**
   This is to count the geometric fibers of $\operatorname{Spec} A_a[x]/\langle f \rangle \to \operatorname{Spec} A_a$.
3. **Pairwise subresultants**
   This is to construct the generic freeness of $A[x]/\langle f, g \rangle$.

The answer

1. **Leading coefficients**
   This is to construct the generic freeness. Let $f \in A[x]$ and a is the leading coefficient of f, then $A_a[x]/\langle f \rangle$ is a free A_a-module of finite rank.
2. **Sub-discriminants**
   This is to count the geometric fibers of $\operatorname{Spec} A_a[x]/\langle f \rangle \to \operatorname{Spec} A_a$.
3. **Pairwise subresultants**
   This is to construct the generic freeness of $A[x]/\langle f, g \rangle$.

So Collins-Hong projection operator contains all the polynomials (possibly more than) needed to encode a stratification of the projection from a hypersurface $V(f)$ into "flat" and "unramified" pieces.

The answer

1. **Leading coefficients**
   This is to construct the generic freeness. Let $f \in A[x]$ and a is the leading coefficient of f, then $A_a[x]/\langle f \rangle$ is a free A_a-module of finite rank.
2. **Sub-discriminants**
   This is to count the geometric fibers of $\operatorname{Spec} A_a[x]/\langle f \rangle \to \operatorname{Spec} A_a$.
3. **Pairwise subresultants**
   This is to construct the generic freeness of $A[x]/\langle f, g \rangle$.

So Collins-Hong projection operator contains all the polynomials (possibly more than) needed to encode a stratification of the projection from a hypersurface $V(f)$ into "flat" and "unramified" pieces.
Therefore proj is not really eliminating variables, but stratifying the projection image. :)

Comparison

- Let $f = w^2 + ((x+y)^2 u - yz) \in R[x, y, z, u, w]$.
  The goal is to build an f-sign-inv. cylindrical decomposition of R^5.

Comparison

- Let $f = w^2 + ((x+y)^2 u - yz) \in R[x, y, z, u, w]$.
  The goal is to build an f-sign-inv. cylindrical decomposition of R^5.
- Using Collins-Hong projection or McCallum-Brown projection, the problem is reduced to computing lower-dimensional sign-invariant c.a.d.s of $R^4, \ldots, R^1$.

Comparison

- Let $f = w^2 + ((x+y)^2 u - yz) \in R[x, y, z, u, w]$.
  The goal is to build an f-sign-inv. cylindrical decomposition of R^5.
- Using Collins-Hong projection or McCallum-Brown projection, the problem is reduced to computing lower-dimensional sign-invariant c.a.d.s of $R^4, \ldots, R^1$.
  - Collins-Hong projection generates a c.a.d. consisting of 297 cells.

Comparison

- Let $f = w^2 + ((x+y)^2 u - yz) \in R[x, y, z, u, w]$.
  The goal is to build an f-sign-inv. cylindrical decomposition of R^5.
- Using Collins-Hong projection or McCallum-Brown projection, the problem is reduced to computing lower-dimensional sign-invariant c.a.d.s of $R^4, \ldots, R^1$.
  - Collins-Hong projection generates a c.a.d. consisting of 297 cells.
  - McCallum-Brown projection yields the same projection polynomials, but the nullification problem occurs. In this case, their correctness is not guaranteed.

Comparison

- Let $f = w^2 + ((x+y)^2 u - yz) \in R[x, y, z, u, w]$.
  The goal is to build an f-sign-inv. cylindrical decomposition of R^5.
- Using Collins-Hong projection or McCallum-Brown projection, the problem is reduced to computing lower-dimensional sign-invariant c.a.d.s of $R^4, \ldots, R^1$.
  - Collins-Hong projection generates a c.a.d. consisting of 297 cells.
  - McCallum-Brown projection yields the same projection polynomials, but the nullification problem occurs. In this case, their correctness is not guaranteed.
- Using our geometric method, this problem is reduced to compute a lower-dimensional c.a.d. adapted to another family of varieties recursively.

Comparison

- Let $f = w^2 + ((x+y)^2 u - yz) \in R[x, y, z, u, w]$.
  The goal is to build an f-sign-inv. cylindrical decomposition of R^5.
- Using Collins-Hong projection or McCallum-Brown projection, the problem is reduced to computing lower-dimensional sign-invariant c.a.d.s of $R^4, \ldots, R^1$.
  - Collins-Hong projection generates a c.a.d. consisting of 297 cells.
  - McCallum-Brown projection yields the same projection polynomials, but the nullification problem occurs. In this case, their correctness is not guaranteed.
- Using our geometric method, this problem is reduced to compute a lower-dimensional c.a.d. adapted to another family of varieties recursively.
  - Our algorithm generates a c.a.d. consisting of **75** cells.

Comparison

To see where the gain comes from, we need to look deeper into the projection phase.

The left column shows the projection polynomials computed by classical methods, while varieties generated by our geometric method are shown in the right column.

	classical	geometric
Level 5	$w^2 + (x+y)^2 u - yz$	$V(0), V(f)$
Level 4	$(x+y)^2 u - yz$	$V(0), V((x+y)^2 u - yz)$
Level 3	$(x+y)^2, yz$	$V(0), V(x+y), V(x+y, yz)$
Level 2	$x+y, y$	$V(0), V(x+y), V(x, y)$
Level 1	x	$V(0), V(x)$

Table 1: A comparison of computation of f-sign-invariant c.a.d.

More comparison

- Compute a c.a.d. adapted to a curve $\mathcal{C} \subseteq \mathbb{A}^5_R$,

More comparison

- Compute a c.a.d. adapted to a curve $\mathcal{C} \subseteq \mathbb{A}^5_R$,
- $\mathcal{C}$ is defined by the locus of polynomials

$$
\begin{aligned}
f_1 &= wu - 1, \\
f_2 &= w^2 - 2wy + u^2 - 2uz - x, \\
f_3 &= 32y^2 + 168yz + 40yx - 270y + 8z^2 + 20zx - 390z + 4x^2 - 105x + 450 \\
f_4 &= 240yz - 16yx^2 - 532yx - 4480y - 800z^3 - 1240z^2x - 17720z^2 - 408zx^2 \\
&\quad -6214zx + 25240z - 40x^3 - 550x^2 + 5695x + 1050 \text{ and} \\
f_5 &= 320yzx + 8320yz + 32yx^2 + 264yx - 14840y + 240z^2 + 16zx^2 - 372zx \\
&\quad -23380z - 140x^2 - 2575x + 36750.
\end{aligned}
$$

More comparison

- Compute a c.a.d. adapted to a curve $\mathcal{C} \subseteq \mathbb{A}^5_R$,
- $\mathcal{C}$ is defined by the locus of polynomials

$$
\begin{aligned}
f_1 &= wu - 1, \\
f_2 &= w^2 - 2wy + u^2 - 2uz - x, \\
f_3 &= 32y^2 + 168yz + 40yx - 270y + 8z^2 + 20zx - 390z + 4x^2 - 105x + 450 \\
f_4 &= 240yz - 16yx^2 - 532yx - 4480y - 800z^3 - 1240z^2x - 17720z^2 - 408zx^2 \\
&\quad -6214zx + 25240z - 40x^3 - 550x^2 + 5695x + 1050 \text{ and} \\
f_5 &= 320yzx + 8320yz + 32yx^2 + 264yx - 14840y + 240z^2 + 16zx^2 - 372zx \\
&\quad -23380z - 140x^2 - 2575x + 36750.
\end{aligned}
$$

- Our algorithm (`GeometricCAD`): finished in 10.67 seconds, generating 200 cells,

More comparison

- Compute a c.a.d. adapted to a curve $\mathcal{C} \subseteq \mathbb{A}^5_R$,
- $\mathcal{C}$ is defined by the locus of polynomials

$$
\begin{aligned}
f_1 &= wu - 1, \\
f_2 &= w^2 - 2wy + u^2 - 2uz - x, \\
f_3 &= 32y^2 + 168yz + 40yx - 270y + 8z^2 + 20zx - 390z + 4x^2 - 105x + 450 \\
f_4 &= 240yz - 16yx^2 - 532yx - 4480y - 800z^3 - 1240z^2x - 17720z^2 - 408zx^2 \\
&\quad -6214zx + 25240z - 40x^3 - 550x^2 + 5695x + 1050 \text{ and} \\
f_5 &= 320yzx + 8320yz + 32yx^2 + 264yx - 14840y + 240z^2 + 16zx^2 - 372zx \\
&\quad -23380z - 140x^2 - 2575x + 36750.
\end{aligned}
$$

- Our algorithm (`GeometricCAD`): finished in 10.67 seconds, generating 200 cells,
- Classical algorithm (`QEPCAD`): quit after 14 minutes because of a software failure.

Implementation

We implement our algorithm on Mathematica, the source code is available at

https://github.com/xiaxueqaq/GeometricCADv2.

Experimental Results

Time	Hong-4	Hong-5	Hong-6	Hong-7
Geometric CAD	0.56	2.53	9.77	49.52
RC-CAD	1.19	5.16	20.00	133.86
QEPCAD	1.35	1.35	1.70	>3600

Time	Hong-8	cyclic-4	cyclic-5	cyclic-6
Geometric CAD	270.33	0.03	0.58	5.48
RC-CAD	Error	0.06	4.58	>3600
QEPCAD	>3600	1.56	>3600	>3600

Table 2: Running time in seconds

Experimental Results

Time	FIS-1	FIS-2	FIS-3	FIS-4
Geometric CAD	16.11	2.66	1654.84	0.19
RC-CAD	40.25	1.59	644.91	1.83
QEPCAD	3.68	1101.60	>3600	4.76

Time	FIS-5	FIS-6	FIS-7	FIS-8
Geometric CAD	0.20	2834.00	10.67	12.75
RC-CAD	>3600	>3600	>3600	122.58
QEPCAD	>3600	>3600	ERROR	ERROR

Table 3: Running time in seconds

The CAD Olympiad

	Total	Solved	Gold:Silver:Bronze
Geometric CAD	20	20	15:5:0
RC-CAD	20	14	3:8:3
QEPCAD	20	11	3:1:7

Table 4: Summary

Inside the Discriminant: Multiplicity and Number of Solutions

- Recall that solution number plays an important role in our construction. It drops when two solutions merge. The system becomes singular in this case.

Inside the Discriminant: Multiplicity and Number of Solutions

- Recall that solution number plays an important role in our construction. It drops when two solutions merge. The system becomes singular in this case.
- The fewer solutions for the system, the more singular the system is.

Inside the Discriminant: Multiplicity and Number of Solutions

- Recall that solution number plays an important role in our construction. It drops when two solutions merge. The system becomes singular in this case.
- The fewer solutions for the system, the more singular the system is.
- Instead of directly counting the solutions, McCallum (1998) improved CAD by using the multiplicity of discriminant to detect how singular the system is in the hypersurface case.

Inside the Discriminant: Multiplicity and Number of Solutions

- Recall that solution number plays an important role in our construction. It drops when two solutions merge. The system becomes singular in this case.
- The fewer solutions for the system, the more singular the system is.
- Instead of directly counting the solutions, McCallum (1998) improved CAD by using the multiplicity of discriminant to detect how singular the system is in the hypersurface case.
- **Theorem (McCallum, 1998)**

  Let $f(y_1, \ldots, y_n, x) \in \mathbb{R}[y_1, \ldots, y_n, x]$, $D(y) = \mathrm{Res}_x(f, \frac{\partial f}{\partial x}) \neq 0$. Let $S \subseteq \mathbb{R}^n$ be a connected submanifold on which f is degree-invariant in x and does not vanish identically, and in which D is multiplicity-invariant. Then f is analytically delineable on S and order-invariant in each f-section over S.

Inside the Discriminant: Multiplicity and Number of Solutions

- Recall that solution number plays an important role in our construction. It drops when two solutions merge. The system becomes singular in this case.
- The fewer solutions for the system, the more singular the system is.
- Instead of directly counting the solutions, McCallum (1998) improved CAD by using the multiplicity of discriminant to detect how singular the system is in the hypersurface case.
- **Conjecture**

  Let $\pi : X \to Y$ be a finite flat morphism of reduced k-varieties. The discriminant D is a closed subscheme of Y. Suppose the Hilbert-Samuel multiplicity $\mathrm{mult}_z(D)$ of D is locally constant along a non-singular subscheme Z of Y. Then the geometric fiber cardinality $n(y) = \#X_{\overline{y}}$ is also locally constant along Z.

Inside the Discriminant: Multiplicity and Number of Solutions

Let us work on a concrete example.

- Recall that the discriminant of $x^3 + px + q = 0$ is $4p^3 + 27q^2$.

Inside the Discriminant: Multiplicity and Number of Solutions

Let us work on a concrete example.

- Recall that the discriminant of $x^3 + px + q = 0$ is $4p^3 + 27q^2$.
- This is a cuspidal curve.

Inside the Discriminant: Multiplicity and Number of Solutions

Let us work on a concrete example.

- Recall that the discriminant of $x^3 + px + q = 0$ is $4p^3 + 27q^2$.
- This is a cuspidal curve.
- The multiplicity of a smooth point (p, q) on the discriminant is 1, which corresponds to a double root.

Inside the Discriminant: Multiplicity and Number of Solutions

Let us work on a concrete example.

- Recall that the discriminant of $x^3 + px + q = 0$ is $4p^3 + 27q^2$.
- This is a cuspidal curve.
- The multiplicity of a smooth point (p, q) on the discriminant is 1, which corresponds to a double root.
- The multiplicity of the cusp $(0, 0)$ is 2, which corresponds to the triple root $x^3 = 0$.

Inside the Discriminant: Multiplicity and Number of Solutions

Dimca and Rosian (1984) showed that for the universal family of monic equations $F(x) = x^n + a_{n-1}x^{n-1} + \cdots + a_0 = 0$ of degree n,

the number of distinct roots + the multiplicity of $D = n$.

Moreover, the Samuel multiplicity stratification is exactly the canonical Whitney stratification of D.

Inside the Discriminant: Multiplicity and Number of Solutions

Dimca and Rosian (1984) showed that for the universal family of monic equations $F(x) = x^n + a_{n-1}x^{n-1} + \cdots + a_0 = 0$ of degree n,

the number of distinct roots + the multiplicity of $D = n$.

Moreover, the Samuel multiplicity stratification is exactly the canonical Whitney stratification of D.

Question: does this pullback to arbitrary finite free corank 1 projection $\pi : X \to Y$ of degree n?

Thank you!

Find more about me on xiaxueqaq.github.io