On the complexity of feasibility, connectivity, and 2^a-isogeny computation

by

Jesse Allister Kasian Elliott

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Computer Science

Waterloo, Ontario, Canada, 2025

© Jesse Allister Kasian Elliott 2025

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Marc Pouget Researcher, INRIA Nancy
Supervisor(s):	David Jao Professor, Department of Combinatorics & Optimization, University of Waterloo
	Éric Schost Professor, David R. Cheriton School of Computer Science, University of Waterloo
Internal Member:	Mark Giesbrecht Professor, David R. Cheriton School of Computer Science, University of Waterloo
Internal Member:	Stephen Watt Professor, David R. Cheriton School of Computer Science, University of Waterloo
Internal-External Member:	Steve Melczer Professor, Department of Combinatorics & Optimization, University of Waterloo
Other Member(s):	Mohab Safey El Din Professor, LIP6 Sorbonne Université

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

I am the sole author of Chapters 1 and 2. I co-authored the remaining chapters.

Research presented in Chapter 3: The work is co-authored with Mark Giesbrecht and Éric Schost. The paper (Bit complexity for computing one point in each connected component of a smooth real algebraic set [23]) was published in the Journal of Symbolic Computation.

Research presented in Chapters 4 and 5: The work is co-authored with Éric Schost. The work presented in Chapter 4 has been accepted for publication in ISSAC 2025: Proceedings of the 50th International Symposium on Symbolic and Algebraic Computation (July 2025).

Research presented in Chapters 6: The work is co-authored with Aaron Hutchinson and David Jao.

Abstract

Algebraic geometry has several real-world applications, including automation and robot motion planning [2, 8], control theory [38], as well as in the foundations of certain post-quantum cryptosystems. For example, isogeny-based cryptography is built on the computational difficulty of finding isogenies between elliptic curves [5]. This thesis investigates complexity in computational algebraic geometry, particularly bit-complexity questions in real algebraic geometry, and 2^a-isogeny computation on Legendre curves. Analyzing the bit complexity enables the development of modular extensions that manage the growth of intermediate expressions, making implementations more efficient in practice for real-world applications.

The main contributions are as follows.

1. We analyze the bit complexity of an algorithm that computes one point per connected component in a real algebraic set. The analysis is of an algorithm by Safey El Din and Schost (*Polar varieties and computation of one point in each connected component of a smooth real algebraic set*, ISSAC'03). This algorithm uses random changes of variables that are proven to generically ensure certain desirable geometric properties such as *weak transversality* and *Noether position* for polar varieties. The cost of the algorithm was given in an algebraic complexity model. We analyze the bit complexity and the error probability, and we provide a quantitative analysis of the genericity statements.

2. The Chinese Remainder Theorem (CRT) is used in developing modular algorithms, controlling the growth of intemediate expressions by performing computations modulo a number of small primes, and then reconstructing the output modulo the product of the primes. Some *unlucky primes* produce errors by returning incorrect results or no result. We can bound their number by finding a nonzero $U \in \mathbb{Z}$ with the property that all unlucky primes divide U.

   Without allowing for errors, the CRT-based modular algorithms requires all primes to be lucky, which typically results in larger primes. Error-correction techniques (introduced by Pernet and Böhm *et al.*) allow a small number of unlucky primes, therefore reducing the prime size. We provide a quantitative analysis that gives explicit sufficient conditions on the number and size of the primes to ensure a given success probability, when given a height bound H on the output and the integer U bounding the number of unlucky primes. We also demonstrate some applications.

3. We give bit-size estimates for computing *roadmaps* in smooth bounded real hypersurfaces. Roadmaps are used to decide if two points in a real algebraic set are continuously connected by a path from one point to the other, and therefore have applications in robot motion planning by deciding if paths exist between points.

   To obtain our height estimates, we apply tools from intersection theory that involve the *arithmetic Chow ring*, developed by D'Andrea, Krick, and Sombra in [16].

4. We introduce a method for efficiently computing 2^a-isogenies in Legendre form with applications in post-quantum cryptography. The majority of work on isogeny computation uses elliptic curves in Montgomery form; this includes the work on Supersingular Isogeny Diffie-Hellman (SIDH) by Jao, De Feo and Plût [29] and Supersingular Isogeny Key Encapsulation (SIKE) [27]. To the best of our knowledge at the time of writing, elliptic curves in Legendre form had not yet been explored for isogeny-based cryptography. Legendre form is an interesting family to study, having a very simple defining equation, and the simplest possible representation of the 2-torsion subgroup.

Acknowledgements

Firstly, I thank my PhD supervisors, David Jao and Éric Schost, for their exceptional supervision. They provided generous support, guidance, and numerous opportunities throughout the entirety of my program. They were always accessible and offered invaluable feedback and encouragement at every stage.

I also thank everyone from the Symbolic Computation Group and the Cryptography, Security, and Privacy Group for their insights, support, and engaging interactions.

My parents made this journey possible from the beginning. They always supported me in every way. Thank you mom and dad for giving me the best education possible.

I thank all my family and friends for standing by through every challenge and adversity. A special thanks to my housemate for his friendship and support throughout this PhD journey.

Also, a special thanks to my partner for her support and sacrifices, and to her family for their encouragement and kindness.

Finally, I thank David R. Cheriton, the Government of Ontario, and the University of Waterloo for their scholarship support.

Dedication

I dedicate this thesis to my parents.

Table of Contents

Examining Committee ii

Author's Declaration iii

Statement of Contributions iv

Abstract v

Acknowledgements vii

Dedication viii

List of Figures xi

1 Introduction 1
- 1.1 Scientific context . . . 2
- 1.2 Problem statements . . . 3
  - 1.2.1 Feasibility . . . 3
  - 1.2.2 Connectivity . . . 4
  - 1.2.3 2^a-isogeny computation . . . 4
- 1.3 Contributions . . . 5
  - 1.3.1 Bit complexity for computing one point in each connected component of a smooth real algebraic set . . . 5
  - 1.3.2 Some applications of Chinese remainder theorem codes with error-correction . . . 6
  - 1.3.3 Bit-size estimates for computing roadmaps in smooth bounded real hypersurfaces . . . 9
  - 1.3.4 Computing 2^a-isogenies in Legendre form . . . 12
- 1.4 Overview of thesis . . . 13

2 Preliminaries 14
- 2.1 Algebraic sets . . . 14
- 2.2 Polar varieties . . . 17

3 Bit complexity for computing one point in each connected component of a smooth real algebraic set **22**
References . . . 54

4 Some applications of Chinese remainder theorem codes with error-correction **57**
References . . . 74

5 Bit-size estimates for computing roadmaps in smooth bounded real hypersurfaces **76**
5.1 Introduction . . . 76
5.1.1 Problem statement and motivation . . . 76
5.1.2 Roadmaps and control points . . . 77
5.1.3 Overview of the algorithm . . . 77
5.1.4 Data structures . . . 79
5.1.5 Main result . . . 80
5.2 Preliminaries . . . 81
5.2.1 Projective geometry . . . 81
5.2.2 The arithmetic Chow ring . . . 83
5.2.3 Canonical height . . . 83
5.3 The algorithm . . . 84
5.3.1 Genericity assumptions . . . 84
5.3.2 Fibers of projections . . . 85
5.3.3 Subroutines . . . 86
5.4 Proof of Theorem 5.1.3 . . . 88
5.4.1 Nested Lagrange systems . . . 89
5.4.2 Canny roadmap: degree and canonical height . . . 91
5.4.3 Main roadmap: degree and canonical height . . . 101
5.4.4 Height of data structures . . . 107

6 Computing 2^a-isogenies in Legendre form **111**
References . . . 129

References **130**

A Pseudocode **135**

List of Figures

2.1 The algebraic set V is defined by the points where $X_1^2+X_2^2+X_3^2-1=0$, the 3-sphere. On the left, the red points correspond to the first polar $W(\pi_1, V)$. On the right, the red circle corresponds to the second polar $W(\pi_2, V)$. Notice $W(\pi_1, V) \subset W(\pi_2, V)$. 19

Chapter 1

Introduction

The work in this thesis addresses complexity questions in computational algebraic geometry. The primary focus is on real algebraic geometry for which we make the following contributions.

1. We analyze the bit complexity of an algorithm that computes one point per connected component in a real algebraic set.

2. We develop quantitative bounds applying the Chinese remainder theorem in a modular algorithm, accounting for errors caused by certain *unlucky* prime moduli.

   By allowing for some errors (a prime dividing a denominator of a rational number, for instance) as opposed to having all prime moduli *lucky*, we can reduce the size of the primes by an order of magnitude or more in bit-length. We demonstrate applications where CRT-based approaches are preferred because lifting techniques are either computationally more difficult or not possible.

3. We give bit-size estimates for computing *roadmaps* in smooth bounded real hypersurfaces. Roadmaps are used to decide if two points in a real algebraic set are continuously connected by a path from one point to the other, and therefore have applications in robot motion planning by deciding if paths exist between points.

The work in the final chapter, while still concerning computational algebraic geometry, is thematically of a different nature. The work introduces a method for efficiently computing 2^a-isogenies in Legendre form with applications in post-quantum cryptography. The majority of work on isogeny computation uses elliptic curves in Montgomery form, includeing the original work on Supersingular Isogeny Diffie-Hellman (SIDH) by Jao, De Feo and Plût [29] and Supersingular Isogeny Key Encapsulation (SIKE) [27]. To the best of our knowledge, at the time of writing, elliptic curves in Legendre form had not yet been explored for isogeny-based cryptography. Legendre form is an interesting family of curves to study, having a simple defining equation and the simplest possible representation of the 2-torsion subgroup.

1.1 Scientific context

Non-linear algebra and applications. Linear algebra has many practical applications across science and engineering. Some of these applications include least squares [48], error-correcting codes [32], and quantum computation [33]. Nevertheless, many real-world problems involve non-linear objects to which linear algebra cannot be applied. Consequently, recent efforts have focused on expanding methods from *non-linear algebra* (i.e. *algebraic geometry*) to address theoretical challenges involving non-linear objects, with applications to real-world problems that include automation and robot motion planning [2, 8], control theory [38], as well as in the foundations of certain post-quantum cryptosystems. For example, isogeny-based cryptography is built on the computational difficulty of finding isogenies between elliptic curves [5].

Cylindrical algebraic decomposition. Tarski and Seidenberg gave the first algorithm for deciding the truth of a first-order formula over the reals, and in particular, for determining the emptiness of a semi-algebraic set [43, 46]. Later, Collins [24] introduced Cylindrical Algebraic Decomposition (CAD), a method docomposing $\mathbb{R}^n$ into cilindrical cells, which improved complexity to polynomial in both the degree and number of polynomials, and doubly exponential in the number of variables. CAD is a very general algorithm that obtains almost all topological information about the algebraic set, but since complexity is doubly exponential in the number of variables, CAD becomes infeasible when the number of variables exceeds two or three.

Critical point methods. Critical point methods are techniques for solving problems in semi-algebraic and real algebraic geometry, including deciding if a semi-algebraic set has real solutions (see for instance [3, 4, 21, 39]). The basic idea of these methods is to define special mappings on an algebraic set, and then to extract topological information about the algebraic set by studying the critical points of the mapping.

Remark 1.1.1. *Critical point methods work with higher dimensional critical point sets called polar varieties. Chapter 2.2 introduces polar varieties.*

Critical point methods have complexity singly exponential in the number of variables, therefore providing an exponential improvement from CAD.[1] The complexity is roughly $(sd)^{O(n^k)}$ for s degree d polynomials in n variables, where k is a constant. Research has focused on improving the theoretical complexity by reducing the exponent k. However, these theoretical improvements do not always result in faster *actual runtime*.

[1] Critical point methods are not always an alternative to CAD. For instance, quantifier elimination over the reals cannot be solved with critical point methods but can be solved with CAD.

Practical algorithms. Real-world applications need algorithms that perform efficiently in practice, where practical efficiency refers to the actual runtime performance. Algorithms with theoretical improvements in complexity are not necessarily efficient in practice. For instance, constants hidden in complexity may be large, or the algorithm may suffer from *intermediate expression swell*, meaning uncontrolled coefficient growth in intermediate expressions when computations are performed over a ground field such as $\mathbb{Q}$.

Bit-complexity. Algebraic algorithms over the rational numbers require an analysis of their *bit complexity* to properly assess actual runtime and practical performance. Bit complexity counts the total number of bit operations. By considering the bit-size of the objects manipulated throughout the algorithm, we gain far more information about both time and space resources needed in practice.

Modular algorithms. Intermediate results of an algorithm can grow larger than the final output, and therefore, even with accurate output bit-size bounds, techniques are needed to control the growth of intermediate expressions. Modular algorithms achieve this by performing computations modulo one or more small primes.[2] Rational reconstruction can then be used to recover the output which is typically a set of polynomials or a matrix of polynomials, given modulo a large integer M. Assuming an upper bound H on the base-2 representation the output, if M is sufficiently large, satisfying

$$\log_2(M) \geq H + 1,$$

rational reconstruction can be applied coefficient-wise to recover the output with rational coefficients.

1.2 Problem statements

Consider a sequence of polynomials $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{Q}[X_1, \ldots, X_n]$ defining an algebraic set $V = V(\boldsymbol{f}) \subset \mathbb{C}^n$.

1.2.1 Feasibility

Problem 1. *Compute at least one point in each connected component of* $V \cap \mathbb{R}^n$.

Problem 1 is a fundamental problem in real algebraic geometry. It has applications for instance in robot motion planning [12]. On its own, it can decide the existence of

[2] Ideally, primes should fit in a single 64-bit machine word (assuming a 64-bit architecture) to avoid the need for multi-precision arithmetic; we call these *machine primes*.

real solutions (*real feasibility*) and estimate an upper bound on the number of connected components of V.

Real feasibility, the problem of determining whether a given system of polynomial equations has a real solution, is central to creating efficient algorithms for polynomial optimization, both constrained and unconstrained [37]. Moreover, computing discriminantal varieties is a closely related problem [31] and involved in both motion planning and robot calibration [8, 11].

Remark 1.2.1. *Here, a connected component refers to the notion of connectedness in the Euclidean topology. A subset is connected if and only if it cannot be partitioned into two disjoint nonempty open subsets.*

1.2.2 Connectivity

A *roadmap* for an algebraic set $V \subset \mathbb{C}^n$ is an algebraic curve $\mathscr{R} \subset V$ with non-empty and connected intersection with each connected components of $V \cap \mathbb{R}^n$. Roadmaps efficiently decide connectivity queries by reducing higher-dimensional computations to one-dimensional computations. In this way, the roadmap is like a skeleton tracking connectivity.

Problem 2. *Compute a roadmap of* $V \cap \mathbb{R}^n$.

Roadmaps are used to decide if two points in a real algebraic set are continuously connected by a path from one point to the other, and therefore have applications in robot motion planning by deciding if paths exist between points [8, 9].

1.2.3 2^a-isogeny computation

Isogenies are mappings between elliptic curves that are compatible with all of their associated internal mathematical structures, in particular the geometric structure (isogenies are rational morphisms) and the algebraic structure (isogenies are group homomorphisms). Several cryptosystems are built on the hypothesized computational hardness of the *supersingular isogeny path problem*: given primes p and l, and two supersingular elliptic curves E and E' over $\mathbb{F}_{p^2}$, find a path from E to E' in the l-isogeny graph.

For additional definitions, see Chapter 6. Introductions can also be found in Silverman [45] and Washington [50].

Elliptic curves have historically played a central role in the development of public-key cryptography. Originally, public-key cryptography was developed using discrete logarithms over finite fields. Over time, for reasons of performance and security, elliptic curve discrete logarithms replaced finite fields for most cryptographic applications. In the context of *post-quantum cryptography*, the study of cryptographic algorithms resistant to quantum computing attacks, discrete logarithms no longer provide any suitable foundation for

public-key cryptography. Accordingly, researchers have turned their attention to using elliptic curve isogenies as a building block for developing quantum-resistant cryptosystems. My work investigates faster ways to compute these isogenies, which could be useful in a variety of applications.

Isogeny-based cryptography remains an active research area within post-quantum cryptography. Isogeny-based schemes are generally compact with smaller key sizes than other available quantum-resistant schemes, although computational time is relatively slower compared to other post-quantum frameworks.

At this time, research in the area of isogeny-based cryptography and its applications is still active, and the full impact of isgoeny-based cryptosystems and algorithms for isogeny computation remains to be determined.

1.3 Contributions

The following notions of height are used in summarizing the results.

- The height of a rational number $r = u/v$ in $\mathbb{Q} \setminus \{0\}$, denoted $\mathrm{ht}(r)$, is the maximum of $\log |u|$ and $\log v$, where $u \in \mathbb{Z}$ and $v \in \mathbb{N}$ are coprime.
- The height of a polynomial with rational coefficients $f \in \mathbb{Q}[X_1, \ldots, X_n]$, denoted $\mathrm{ht}(f)$, where $v \in \mathbb{N}$ is the smallest common denominator of its nonzero coefficients, is the maximum of $\log v$ and the logarithms of the absolute values of the integer coefficients of vf.

1.3.1 Bit complexity for computing one point in each connected component of a smooth real algebraic set

In Chapter 3, I present our work giving the bit complexity of an algorithm that solves Problem 1 when V is smooth and $(n-s)$-equidimensional.

The following is our main theorem.

Theorem. *Let $\boldsymbol{f} = (f_1, \ldots, f_s) \in \mathbb{Z}[X_1, \ldots, X_n]^s$ be a sequence of polynomials with $\deg(f_i) \leq d$ and $\mathrm{ht}(f_i) \leq b$. Suppose that the ideal generated by $f_1, \ldots, f_s$ is radical and that $V = V(\boldsymbol{f}) \subset \mathbb{C}^n$ is smooth and equidimensional of dimension $n-s$. Also suppose that $0 < \epsilon < 1$.*

There exists a randomized algorithm that takes $\boldsymbol{f}$ and ϵ as input and produces $n-s+1$ zero-dimensional parameterizations, the union of whose zeros includes at least one point in each connected component of $V(\boldsymbol{f}) \cap \mathbb{R}^n$, with probability at least $1-\epsilon$. Otherwise, the

algorithm either returns a proper subset of the points, or FAIL. In any case, the algorithm uses

$$O^{\sim}(d^{3n+2s+1}\log(1/\epsilon)(b+\log(1/\epsilon)))$$

bit operations. The polynomials in the output have degree at most d^{n+s}*, and height*

$$O^{\sim}(d^{n+s+1}(b+\log(1/\epsilon))).$$

The work extends our analysis of the hypersurface case (*On the bit complexity of finding points in connected components of a smooth real hypersurface, ISSAC'20*). We extend the analysis to more general cases where the input variety is defined by the vanishing of a set of polynomials rather than a single equation. We analyze the bit complexity, the error probability, and provide a quantitative analysis of the genericity statements. The algorithm is bassed on the critical point method and uses polar varieties (see Section 2.2), which we describe using Lagrange systems, simplifying the application of techniques such as weak transversality and an effective Nullstellensatz.

Future work. Together with Mark Giesbrecht, Edern Gillot, Mohab Safey El Din, and Éric Schost, we are refining the bit complexity analysis and expecting an exponential speed up with respect to the number of variables (while the overall complexity still remains exponential). The improvement is based on tools that exploit the multi-homogeneous structure of polynomial systems encoding critical sets (i.e. polar varieties — see Chapter 2.2 for more details on polar varieties).

1.3.2 Some applications of Chinese remainder theorem codes with error-correction

Background and overview. Algorithms using modular techniques based on the Chinese Remainder Theorem (CRT) avoid intermediate expression swell by computing modulo a sequence of small primes

$$p_1, p_2, \ldots, p_\eta.$$

To recover the final result from these modular computations, CRT is applied via an explicit isomorphism between the ring $\mathbb{Z}/M\mathbb{Z}$, where $M = p_1 p_2 \cdots p_\eta$, and the product ring

$$\mathbb{Z}/p_1\mathbb{Z} \times \cdots \times \mathbb{Z}/p_\eta\mathbb{Z}.$$

Each computation modulo p_i gives a residue, and CRT guarantees a unique solution modulo M that is congruent to each of these residues modulo p_i. However, some primes may be *unlucky* in the sense of resulting in incorrect results or no results at all (e.g. division by zero). The number of these primes can be bounded by finding a nonzero $U \in \mathbb{Z}$ with the property that all unlucky primes divide U.

Remark 1.3.1. *One can also avoid intermediate expression swell by choosing a single large prime equal to the total bit-size of the output. However, this approach is more computationally expensive: multi-precision arithmetic may be required, whereas using small 64-bit primes (on a 64-bit architecture) avoids multi-precision arithmetic. Moreover, using small primes improves complexity overall. Indeed, consider η primes of size h and an algorithm performing T steps. With fast arithmetic a step costs $h\log(h)$ [25], giving a total cost of*

$$TNh\log(h).$$

Now, compare with using a single large prime of size Nh. The total cost grows to

$$TNh\log(Nh) = TNh(\log(N) + \log(h)).$$

The small prime method also requires computations for the change of representation. However, this cost can be done in time $N\log^2(N)$ (see [49]) and is not the bottleneck in most cases of interest.

Without using error-correction, CRT-based modular algorithms require that all prime moduli are lucky. To obtain success with high probability, larger primes may be needed, reducing efficiency. In other words, by allowing errors, we can reduce the prime size and improve efficiency.

In Chapter 4, using a bound H on the height of the output, and the integer U, bounding the unlucky primes, we provide a quantitative analysis giving explicit sufficient conditions on the number and size of the primes needed to achieve a given success probability $1-\varepsilon$.

We provide two examples illustrating this approach:

1. computing lexicographic Gröbner bases in two variables, and
2. solving zero-dimensional systems of polynomial equations.

Quantitative bounds. The goal is to reconstruct a rational number $r = f/g \in \mathbb{Q}$, with $g > 0$ and f, g coprime.

- We require an upper bound that we have denoted H on the base-2 representation of r:
$$\log_2(|f|),\ \log_2(g) \leq H.$$
- For primes $p \in \mathbb{N}$,
$$r(p) \in \{0, \ldots, p-1\} \cup \{\infty\},$$
where we let ∞ be a new symbol when no integer value is available. Then we say that p is *lucky* if $r(p) = r \bmod p$ and otherwise we say p is *unlucky*.

- We assume there are finitely many unlucky primes and let U be their product. In addition to the output size bound H, we assume a bound C with

$$\log_2(U) \leq C.$$

- Let σ be a positive integer and let

$$\Sigma = \{\sigma, \ldots, 2\sigma\} \subset \mathbb{N}.$$

- Assume η primes $p_1, \ldots, p_\eta$ in Σ with

$$\eta = \left\lceil \frac{(2+\gamma)H}{\log_2(\sigma)} \right\rceil, \tag{1.1}$$

  and$\gamma > 0$ fixed. Since all chosen primes are in Σ, CRT for rational reconstruction works error-free when $\eta \log_2(\sigma) > 2H$. With errors, a factor of 2 is too tight. However, $2+\gamma$ is sufficient, and therefore when γ is small, modular computations are near the error-free case.

- Define

$$N = \sum_{1 \leq i \leq \eta} \log_2(p_i) \quad \text{and} \quad L = \sum_{\substack{1 \leq i \leq \eta \\ p_i \text{ unlucky}}} \log_2(p_i).$$

  Pernet proved in [34, Lemma 2.5.4] that when

$$L < (N - 2H + 1)/2,$$

  then, given $[p_1, \ldots, p_\eta]$ and $[r(p_1), \ldots, r(p_\eta)]$, we are able to reconstruct r. Böhm *et al.* also gave similar results.

Remark 1.3.2. *For the algorithms we consider, it is reasonable to assume that only finitely many primes are unlucky. Intuitively, a prime is unlucky if it causes a division by zero or if it changes whether a quantity is zero modulo the prime. Heuristically, with only finitely many such quantities, there can be only finitely many unlucky primes.*

Remark 1.3.3. *Both H and C are problem-dependent, and in the applications we demonstrate, we explain how we find H and C.*

We give two quantitative bounds, the first uses Markov's inequality and the second uses Chebyshev's inequality. They are the following.

Proposition. *(Markov) Let r, H, C be as above. For $\varepsilon > 0$ and $\gamma > 0$, let σ be an integer such that*

$$\sigma \geq \max\left(8(2+\gamma)H,\ 8\left(\frac{2+\gamma}{\gamma}\right)\frac{C}{\varepsilon}\right),$$

and take η as in (1.1).

Select pairwise distinct primes $p_1, \ldots, p_\eta$ independently and uniformly at random from the set $\Sigma = \{\sigma, \ldots, 2\sigma\} \subset \mathbb{N}$. Then, with probability at least $1 - \varepsilon$, given $[p_1, \ldots, p_\eta]$ and the vector $[r(p_1), \ldots, r(p_\eta)]$, one can reconstruct r.

Proposition. *(Chebyshev) Let r, H, C be as above. For $\varepsilon > 0$ and $\gamma > 0$, let σ be an integer such that*

$$\sigma \geq \max\left(8(2+\gamma)H,\ 16^{(\gamma+2)/\gamma},\ 8C\left(\frac{2+\gamma}{\gamma}\right),\ 2A\log_2(A)\right),$$

with $A := 128\frac{(2+\gamma)}{\gamma^2}\frac{C}{\varepsilon H}$, and take η as in (1.1). Then the conclusion of the proposition above holds again.

The two propositions use an error-correcting CRT algorithm due to Böhm *et al.* [7] and Pernet [34]; the propositions give explicit sufficient conditions on the number of primes and their size to guarantee an arbitrary probability of success. Our derivations, to our knowledge, while being relatively straightforward, are new.

Future work. The main application we illustrate is the solution of zero-dimensional polynomial systems. We extend Theorem 2.1 in [19] to obtain a bound on U (all unlucky primes divide U), which bounds the number of unlucky primes for the problem of solving zero-dimensional polynomial systems. We would like to extend the result further by removing the assumption that the polynomial system has dimension zero. That is, assuming, as input, we are given polynomials $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{Z}[X_1, \ldots, X_n]$, our goal is to compute a description of the zero-dimensional component of $V(\boldsymbol{f})$, the isolated solutions of $\boldsymbol{f}$. The open problem is obtaining a bound on U in this more general context.

1.3.3 Bit-size estimates for computing roadmaps in smooth bounded real hypersurfaces

In Chapter 5, we give bit-size estimates for computing roadmaps of $V \cap \mathbb{R}^n$, Problem 2, when V defines a closed and bounded hypersurface.

The algorithm was given by Safey El Din and Schost in [42] (*A baby steps/giant steps probabilistic algorithm for computing roadmaps in smooth bounded real hypersurface*). Algebraic complexity was given, and bit complexity was left for future work. The algorithm also uses random coordinate changes to establish genericity properties. For a degree d, square-free polynomial f in $\mathbb{Q}[X_1, \ldots, X_n]$ with

1. $V(f) \subset \mathbb{C}^n$ smooth, and
2. $V(f) \cap \mathbb{R}^n$ bounded,

the main result roughly states: there exists a randomized algorithm that, upon success, if all genericity properties are satisfied, computes a roadmap of $V(f)$ after performing $(nd)^{O(n^{1.5})}$ arithmetic operations. Assuming bounds on:

1. the height of f, and

2. the height of entries in the changes of coordinates we apply to f,

we give bit-size estimates showing that the height of the output follows the same order of complexity.

This algorithm is again based on the *critical point method* (see Section 2.2). Critical point sets of projection mappings on the hypersurface are calculated to extract connectivity information. We represent these critical sets, *polar varieties*, through Lagrange systems. By using the Lagrangian formulation, we obtain equations that are a complete intersection. We are then able to apply techniques from intersection theory that incorporate the *arithmetic Chow ring* (see [16]), which we use to obtain our height bounds.

Data structures. The output of the main algorithm is a data structure representing an algebraic curve called a one-dimensional parameterization. Let $C \subset \mathbb{C}^n$ be a one-dimensional algebraic set defined over $\mathbb{Q}$. A *one-dimensional parametrization*

$$\mathscr{Q} = ((q, v_1, \ldots, v_n), (\lambda, \lambda')),$$

is a data structure consisting of

- bi-variate polynomials $(q, v_1, \ldots, v_n) \in \mathbb{Q}[T, U]$, q square-free and monic in T and U, with $\mathrm{GCD}(q, v_1) = 1$ and $\deg(v_i) < \deg(q)$;
- linear forms $\lambda = \lambda_1 X_1 + \ldots + \lambda_n X_n$, and $\lambda' = \lambda'_1 X_1 + \ldots + \lambda'_n X_n$ with
  - $$\lambda(v_1, \ldots, v_n) = T\frac{\partial q}{\partial T} \mod q;$$
  - $$\lambda'(v_1, \ldots, v_n) = U\frac{\partial q}{\partial T} \mod q.$$

Then C is the Zariski closure

$$\mathsf{Z}(\mathscr{Q}) := \overline{\left\{ \left(\frac{v_1(\tau,\rho)}{\partial q/\partial T(\tau,\rho)}, \ldots, \frac{v_n(\tau,\rho)}{\partial q/\partial T(\tau,\rho)} \right) \in \mathbb{Q}^n \mid q(\tau,\rho) = 0, \frac{\partial q}{\partial T}(\tau,\rho) \neq 0 \right\}}.$$

Main result. The algorithm uses a finite set of *control points* $\mathscr{P}$ included in the input along with V which are a subset of V. These points are used to test if the points of $\mathscr{P}$ are connected on $V \cap \mathbb{R}^n$.

Our main result is the following.

Theorem. *Consider a square-free polynomial f in $\mathbb{Q}[X_1, \ldots, X_n]$ of degree d and height h, such that*

1. *$V(f) \subset \mathbb{C}^n$ is smooth, and*
2. *$V(f) \cap \mathbb{R}^n$ is bounded.*

Also let $\mathscr{P}$ be a set of control points of degree $\mathfrak{d}$ and height $\mathfrak{h}$. Then, upon success, if all genericity properties are satisfied, the height of the one-dimensional parameterization of the roadmap of $(V(f), \mathscr{P})$ in the output of [42, MainRoadmap] is

$$O^{\sim}\left(\mathfrak{h} d^{24n^{3/2}} + \mathfrak{d}(h + h_{\boldsymbol{A}} + h_{\boldsymbol{u}} + h_{\boldsymbol{A}^{-1}}) d^{30n^{3/2}+1}\right)$$

where

- *$h_{\boldsymbol{A}}$ bounds the height of entries in the change of coordinates matrix $\boldsymbol{A}$;*
- *$h_{\boldsymbol{u}}$ bounds the height of the coefficients in the linear forms of the Lagrange systems;*
- *$h_{\boldsymbol{A}^{-1}}$ bounds the height of entries in the inverse change of coordinates matrix $\boldsymbol{A}^{-1}$.*

Future work. We aim to fully analyze the bit complexity of the algorithm which requires the following steps.

1. We aim to give quantitative versions of the genericity statements. By quantitative, this means bounding the degrees of hypersurfaces containing changes of coordinates we need to avoid in order to establish genericity. We can reuse the analysis from Chapter 3 in analyzing Noether position and transversality, but now we also need to take into account singular points. Furthermore, an additional genericity property needs to be analyzed which is not in Chapter 3. We can revisit the proof in [40, Appendix D] which shows the property holds in generic coordinates, and extend the result to obtain a quantitative version.

   This analysis would obtain bounds for $h_{\boldsymbol{A}}$ and $h_{\boldsymbol{u}}$.

2. The subroutines need to be analyzed and given modular extensions. The work in Chapter 4 can be used for solving zero-dimensional polynomial systems, and the bit-size estimates from [15, Theorem 1] can be used for bounding the height of data structures for solving one-dimensional polynomials systems. However, there are a number of additional subroutines that also need to be analyzed.

1.3.4 Computing 2^a-isogenies in Legendre form

We introduce a method for efficiently computing 2^a-isogenies in Legendre form with applications in post-quantum cryptography (e.g. the Charles-Goren-Lauter (CGL) hash function [13, 17]).

Legendre form has a simple defining equation, and the simplest possible representation of the 2-torsion subgroup, making this family of curves particularly interesting to study. As far as we know, elliptic curves in Legendre form had not yet been explored for isogeny-based cryptography at the time of writing.

Isogeny computation generally uses elliptic curves in Montgomery form, including the original work on supersingular isogeny diffie-hellman (SIDH) by Jao, De Feo and Plût [29] and the most recent implementation of Supersingular Isogeny Key Encapsilation (SIKE) [27].

Remark 1.3.4. *SIDH is a key exchange algorithm proposed by Jao and De Feo 2011 [28]. It was conjectured to be post-quantum secure. However, it was broken in 2022 by Castryck and Descru [10]. Our isogeny computation builds on the SIDH framework, and we reference SIDH throughout the paper using the same terminology.*

In Chapter 6, we develop a new framework for constructing 2^a-isogenies using elliptic curves in Legendre form. We provide a detailed description of how a degree 2^a isogeny can be constructed between elliptic curves in Legendre form with applications in isogeny-based cryptography (i.e. the CGL hash function [13, 17]). The work derives several results for elliptic curves in Legendre form and degree 2 isogenies between curves in Legendre form.

The naive construction of a degree 2 isogeny between Legendre curves requires computing a square root. We avoid the square root computation using an order 4 point above the kernel generator. We also describe how to convert between Montgomery and Legendre form giving explicit isomorphisms.

Future work. We compute the isogeny through the SIDH framework. Let φ be the target 2^a-isogeny. The construction proceeds with a sequence of 2-isogenies $\varphi_i : E_i \to E_{i+1}$ with $\ker(\varphi_i) = \langle R_i \rangle$. The points R_i are computed through *strategy traversal*, which is a costly task and serving only to construct φ_i.

However, R_i has a simple form: each R_i is a nontrivial 2-torsion point on the curve E_i, and when E_i is a Legendre curve L_{λ_i}, then

$$L_{\lambda_i}[2] = \{\mathcal{O}, (0,0), (1,0), (\lambda, 0)\}$$

for a parameter λ defining the Legendre curve.

Since the 2-torsion of Legendre curves is known in advance and is identical for all curves, one may consider an alternative approach for constructing the isogeny φ. If each

φ_i is constructed with R_i chosen as $(0,0)$ or $(1,0)$ (we explain in Chapter 6 how to avoid the $(\lambda,0)$ case), then the value of each R_i is a bit β_i with $R_i = (\beta_i, 0)$. Therefore, one could use some bitstring $(\beta_1, \ldots, \beta_e)$ and *choose* to use the value of $R_i = (\beta_i, 0)$ for each kernel generator rather than derive R_i through strategy traversal. This would have the effect of skipping strategy traversal altogether—improving efficiency by a $O(\log d)$ factor where d is the tree depth in the strategy traversal—since all points R_i are known from the start.

1.4 Overview of thesis

In Chapter 2, we provide preliminaries on algebraic sets and polar varieties. In Chapter 3, we present our work on analyzing the bit complexity for computing one point in each connected component of a smooth real algebraic set. In Chapter 4, we present our work on Chinese Remainder Theorem codes with error correction. In Chapter 5, we give bit-size estimates for computing roadmaps in smooth bound real hypersurfaces. Finally, in Chapter 6, we present our work on computing 2^a-isogenies in Legendre form.

Chapter 2

Preliminaries

Let $\mathbb{Q}, \mathbb{R}$, and $\mathbb{C}$ denote the sets of rational, real, and complex numbers, and let $\mathbb{Z}$ denote the integers.

2.1 Algebraic sets

Let $\boldsymbol{f} = (f_1, \ldots, f_s)$ be a sequence of polynomials in $\mathbb{C}[X_1, \ldots, X_n]$. The *algebraic set* $V = V(\boldsymbol{f})$ associated with $\boldsymbol{f}$ is the set of common zeros:

$$\{(x_1, \ldots, x_n) \in \mathbb{C}^n \mid f_1(x_1, \ldots, x_n) = \cdots = f_s(x_1, \ldots, x_n) = 0\}.$$

In this work, we use the terms algebraic set and *variety* interchangeably.

For an algebraic set $V \subset \mathbb{C}^n$, the set of polynomials in $\mathbb{C}[X_1, \ldots, X_n]$ that vanish on every point of V,

$$\{f \in \mathbb{C}[X_1, \ldots, X_n] \mid f(x_1, \ldots, x_n) = 0,\ \forall (x_1, \ldots, x_n) \in V\},$$

denoted $I(V)$, is called the *ideal* of V.

For more preliminaries on algebraic sets see for instance [14, 44].

Irreducibility. An algebraic set $V \subset \mathbb{C}^n$ is *irreducible* if it cannot be written as the union of two proper algebraic subsets. That is, V is irreducible if

$$V = V_1 \cup V_2 \quad \text{implies that either } V = V_1 \text{ or } V = V_2.$$

If $V \subset \mathbb{C}^n$ is an algebraic set then V can be uniquely decomposed into a finite union of irreducible components:

$$V = V_1 \cup V_2 \cup \cdots \cup V_r,$$

where each V_j is irreducible.

Dimension. We let $\dim(V)$ denote the *dimension* of an algebraic set $V \subset \mathbb{C}^n$; it can be defined as the unique integer d with

$$V \cap H_1 \cap \cdots \cap H_d$$

finite but not empty, for a generic choice of hyperplanes $H_1, \ldots, H_d$. The *codimension* of V is $n - \dim(V)$.

An algebraic set V is *equidimensional* if each of its irreducible components has the same dimension; if each component has dimension d, then we say that V is d-equidimensional.

Regular sequences. Consider a sequence of polynomials $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{C}[X_1, \ldots, X_n]$ and let $V = V(\boldsymbol{f}) \subset \mathbb{C}^n$ be their corresponding algebraic set. Then, $\boldsymbol{f}$ is a *reduced regular sequence* when: for $i = 1, \ldots, s$, the ideal

$$\langle f_1, \ldots, f_i \rangle \subset \mathbb{C}[X_1, \ldots, X_n]$$

is radical and

$$V(\langle f_1, \ldots, f_i \rangle) \subset \mathbb{C}^n$$

has dimension $n - i$. If the radicality assumption is dropped, then $\boldsymbol{f}$ is simply called a *regular sequence*.

An algebraic set $V \subset \mathbb{C}^n$ is a *complete intersection* when its defining ideal $I \subset \mathbb{C}[X_1, \ldots, X_n]$ is generated by a regular sequence $f_1, \ldots, f_s$ of polynomials with $\dim(V) = n - s$.

In Chapters 3 and 5 we use Lagrange systems describing higher-dimensional critical point sets (see polar varieties next in Section 2.2). We use the Lagrangian formulation because the resulting polynomial equations define a complete intersection, and our methods rely on this structure. See [20] for more details on regular sequences.

Degree. Let V be an irreducible algebraic set. The *degree* of V is the number of intersection points with $\dim(V)$ generic hyperplanes, and the degree of a reducible variety is the sum of the degrees of its irreducible components.

Tangent spaces and singular points. Consider a d-equidimensional algebraic set $V \subset \mathbb{C}^n$. The *Zariski tangent space* to V at a point $\boldsymbol{x} \in V$ is the vector space $T_{\boldsymbol{x}}V$ defined by equations

$$\nabla f(\boldsymbol{x}) \cdot \mathbf{v} = 0, \forall f \in I(V), \quad \boldsymbol{v} \in \mathbb{C}^{n \times 1},$$

where $\nabla f(\boldsymbol{x})$ is the gradient of f evaluated at $\boldsymbol{x}$.

Remark 2.1.1. *We also write*

$$\nabla_{(X_1, \ldots, X_n)} f$$

when we want to explicitly state the variables of differentiation.

A point $\boldsymbol{x} \in V$ is a *regular point* when

$$\dim(T_{\boldsymbol{x}}V) = d,$$

and otherwise a *singular point*. The set of *regular* points of V is denoted $\mathrm{reg}(V)$, and the set of *singular points* of V is denoted $\mathrm{sing}(V)$. If $\mathrm{sing}(V)$ is empty then we say that V is *smooth*.

Proposition 2.1.2. *(Jacobian Criterion) If $V \subset \mathbb{C}^n$ is d-equidimensional and has defining ideal $I(V)$ generated by polynomials*

$$\boldsymbol{g} = (g_1, \ldots, g_s) \in \mathbb{C}[X_1, \ldots, X_n]^s,$$

then for $\boldsymbol{x} \in \mathrm{reg}(V)$, the Jacobian matrix $\mathrm{jac}_{\boldsymbol{x}}(\boldsymbol{g})$ has full rank $n - d$, and its kernel is equal to $T_{\boldsymbol{x}}V$.

Proof. See [20, Corollary 16.20]. □

Noether position. Consider a sequence of polynomials $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{C}[X_1, \ldots, X_n]$ defining a d-equidimensional algebraic set $V \subset \mathbb{C}^n$. Also consider the projection

$$\begin{aligned} \pi_i : \mathbb{C}^n &\to \mathbb{C}^i \\ (x_1, \ldots, x_n) &\mapsto (x_1, \ldots, x_i), \end{aligned}$$

for $i \in \{1, \ldots, n - s + 1\}$. Then, V is in *Noether position* for π_d if the field extension

$$\mathbb{C}[X_1, \ldots, X_d] \to \mathbb{C}[X_1, \ldots, X_n]/I(V)$$

is integral. Consequently, for any choice of $(\beta_1, \ldots, \beta_d) \in \mathbb{C}^d$, the fiber

$$V \cap \pi_d^{-1}(\beta_1, \ldots, \beta_d)$$

is finite and non-empty.

Genericity and changes of coordinates. Denote by $\mathrm{GL}(n)$ the set of $n \times n$ invertible matrices. Also denote by $\mathrm{GL}(n, e)$ the subset of matrices in $\mathrm{GL}(n)$ which leave the first e coordinates invariant and act only on the last $n - e$.

Consider an algebraic set $V \subset \mathbb{C}^n$, and a matrix $\boldsymbol{A} \in \mathrm{GL}(n, \mathbb{Q})$. Let $V^{\boldsymbol{A}}$ denote the image of V under the map

$$\boldsymbol{x} \to \boldsymbol{A}^{-1}\boldsymbol{x}.$$

For $f \in \mathbb{C}[X_1, \ldots, X_n]$, let $f^{\boldsymbol{A}}$ denote the polynomial $f(\boldsymbol{AX})$, where $\boldsymbol{X}$ is the column vector $[X_1 \cdots X_n]^{\perp}$. Considering a sequence of polynomials $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{C}[X_1, \ldots, X_n]$, let $\boldsymbol{f}^{\boldsymbol{A}}$ denote the subset $\{f^{\boldsymbol{A}} \mid f \in \boldsymbol{f}\}$. Notice

$$V(\boldsymbol{f})^{\boldsymbol{A}} = V(\boldsymbol{f}^{\boldsymbol{A}}).$$

The algorithms we consider in Chapters 3 and 5 require that the input satisfy certain geometric properties in generic coordinates. We ensure these properties through applying a randomly chosen change of coordinates to the input polynomial system. *Generic* means there exists a non-empty Zariski open subset $\mathcal{O} \subset \mathbb{C}^{n\times n}$ such that if $\boldsymbol{A} \in \mathcal{O}$, then $V^{\boldsymbol{A}}$ has the desired geometric properties.

2.2 Polar varieties

The critical point method uses *polar varieties* to identify and analyze critical points of projection mappings. These critical points are then used to extract connectivity information. Polar varieties are fundamental to the algorithms covered in Chapters 3 and 5.

Background. Polar varieties are higher-dimensional critical loci. They were first introduced in the 1930's for the purpose of defining characteristic classes (see [35, 47]). Since then, polar varieties have played an important role in effective real algebraic geometry where they are used as tools for algorithm design and complexity improvements (see [3, 4, 21, 39]).

Consider a sequence of polynomials $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{C}[X_1, \ldots, X_n]$ defining an algebraic set $V = V(\boldsymbol{f}) \subset \mathbb{C}^n$. For $i \in \{1, \ldots, n-s+1\}$, let π_i denote the canonical projection

$$\begin{array}{ccc} \mathbb{C}^n & \to & \mathbb{C}^i \\ (x_1, \ldots, x_n) & \mapsto & (x_1, \ldots, x_i). \end{array}$$

A point $\boldsymbol{x} \in V$ is a *critical point* if the dimension of $\pi_i(T_{\boldsymbol{x}}V)$ is less than i, where $T_{\boldsymbol{x}}V$ denotes the tangent space of V at $\boldsymbol{x}$. The i-th *open polar variety*

$$W^o(\pi_i, V) := \{\boldsymbol{x} \in \mathrm{reg}(V) \mid \dim(\pi_i(T_{\boldsymbol{x}}V)) < i\}$$

is the set of critical points of π_i on the regular points of V. Then, the i-th polar variety is the Zariski closure of $W^\circ(\pi_i, V)$ which we denote as $W(\pi_i, V)$, and $K(\pi_i, V)$ is the union of $W^\circ(\pi_i, V)$ and $\mathrm{sing}(V)$. Moreover, $K(\pi_i, V)$ is an algebraic set that also contains $W(\pi_i, V)$. Therefore we have that

$$W^\circ(\pi_i, V) \;\subset\; W(\pi_i, V) \;\subset\; K(\pi_i, V).$$

When V is smooth, these three sets coincide. In what follows assume V is smooth.

Determinantal modeling. Polar varieties arise from Jacobian rank conditions and therefore can also be described as determinantal varieties. Consider the Jacobian matrix

$$\mathrm{jac}(\boldsymbol{f}) = \begin{bmatrix} \frac{\partial f_1}{\partial X_1} & \cdots & \frac{\partial f_1}{\partial X_n} \\ \vdots & & \vdots \\ \frac{\partial f_s}{\partial X_1} & \cdots & \frac{\partial f_s}{\partial X_n} \end{bmatrix}.$$

Recall the i-th polar variety, $W(\pi_i, V)$, is the set of critical points of π_i on V. These points correspond to having the truncated Jacobian matrix

$$\operatorname{jac}(\boldsymbol{f}, i) = \begin{bmatrix} \frac{\partial f_1}{\partial X_{i+1}} & \cdots & \frac{\partial f_1}{\partial X_n} \\ \vdots & & \vdots \\ \frac{\partial f_s}{\partial X_{i+1}} & \cdots & \frac{\partial f_s}{\partial X_n} \end{bmatrix},$$

rank deficient, implying that the maximal, s-minors vanish. Let

$$\operatorname{Minors}(\boldsymbol{f}, s) = (M_{i,1}, \ldots, M_{i,S_i})$$

be the minors of size s of $\operatorname{jac}(\boldsymbol{f}, i)$, where

$$S_i = \binom{s}{s}\binom{n-i}{s} = \binom{n-i}{s}.$$

Proposition 2.2.1. *Consider $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{C}[X_1, \ldots, X_n]$. If the ideal defined by $\boldsymbol{f}$ is radical and $V = V(\boldsymbol{f})$ is smooth, then the polar variety*

$$W(\pi_i, V) = \{\boldsymbol{x} \in V \mid \dim \pi_i(T_{\boldsymbol{x}}V) < i\}$$

is defined by the vanishing of $\boldsymbol{f}$ and Minors$(\boldsymbol{f}, s)$.

Proof. See [22]. □

Consequently, when $\boldsymbol{f} = (f)$ and $V = V(f)$ is a hypersurface then it follows from Proposition 2.2.1 that the equations defining the polar variety conveniently simplify into a regular sequence. The polar variety $W(\pi_i, V(f))$ is then defined by the vanishing of f and its partial derivatives:

$$\left(f, \frac{\partial f}{\partial X_{i+1}}, \ldots, \frac{\partial f}{\partial X_n}\right).$$

Consider the following example.

Example 2.2.2. *Consider a polynomial*

$$f = X_1^2 + X_2^2 + X_3^2 - 1 \in \mathbb{C}[X_1, X_2, X_3]$$

and consider the hypersurface

$$V = V(X_1^2 + X_2^2 + X_3^2 - 1) \subset \mathbb{C}^3.$$

The second polar variety is the set of critical points of the projection

$$\begin{aligned} \pi_2 : \mathbb{C}^3 &\to \mathbb{C}^2 \\ (x_1, x_2, x_3) &\mapsto (x_1, x_2) \end{aligned}$$

on $V(f)$, defined by

$$V\left(f, \frac{\partial f}{\partial X_3}\right).$$

Therefore, the second polar variety $W(\pi_2, V(f))$ is defined by those points where

$$X_1^2 + X_2^2 + X_3^2 - 1 = X_3 = 0.$$

The first polar variety is the set of critical points of the projection

$$\begin{aligned} \pi_2 : \mathbb{C}^3 &\to \mathbb{C} \\ (x_1, x_2, x_3) &\mapsto (x_1) \end{aligned}$$

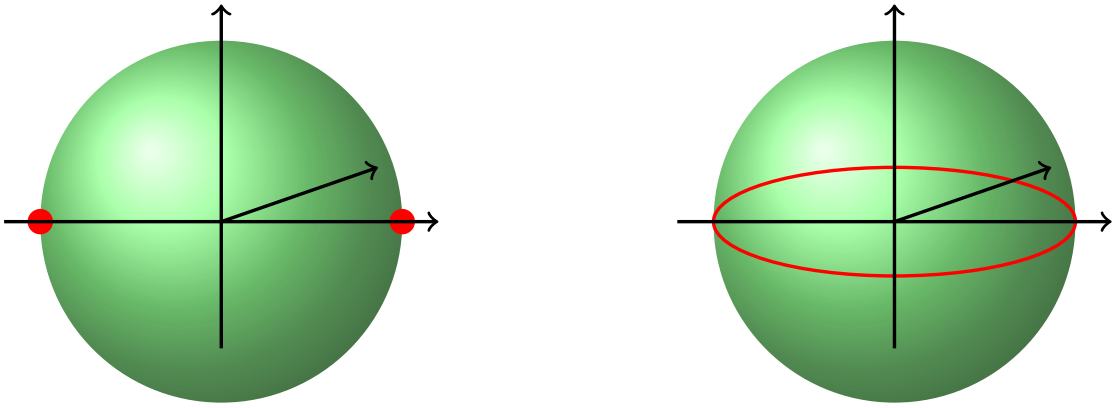
on $V(f)$ are defined by

$$V\left(f, \frac{\partial f}{\partial X_2}, \frac{\partial f}{\partial X_3}\right).$$

The first polar variety $W(\pi_1, V(f))$ is therefore defined by those points where

$$X_1^2 + X_2^2 + X_3^2 - 1 = X_2 = X_3 = 0.$$

Refer to Figure 2.1, which illustrates the real points of the polar varieties $W(\pi_1, V(f))$ and $W(\pi_2, V(f))$ which respectively correspond to the critical points of projections onto a line and a plane.



(a) $W(\pi_1, V(X_1^2 + X_2^2 + X_3^2 - 1))$ (b) $W(\pi_2, V(X_1^2 + X_2^2 + X_3^2 - 1))$

Figure 2.1: The algebraic set V is defined by the points where $X_1^2 + X_2^2 + X_3^2 - 1 = 0$, the 3-sphere. On the left, the red points correspond to the first polar $W(\pi_1, V)$. On the right, the red circle corresponds to the second polar $W(\pi_2, V)$. Notice $W(\pi_1, V) \subset W(\pi_2, V)$.

Lagrangian modeling. Unfortunately, (except for the hypersurface case) the equations in the determinantal modeling of polar varieties are not guaranteed to be a regular sequence. There are too many equations. Indeed, when $n - i \sim 2s$, the length of the sequence of minors, Minors$(\boldsymbol{f}, s)$, is

$$S_i = \binom{n-i}{s} \sim 2^{n-i+1}.$$

To address this challenge, we adopt an alternative modeling of polar varieties introducing Lagrange multipliers. Rather than explicitly representing the polar variety W in $\mathbb{C}^n$, we construct an algebraic set $\mathscr{W}$ in a higher-dimensional space $\mathbb{C}^n \times \mathbb{C}^s$, for $s > 0$, which then maps onto W: the polar variety W is the Zariski closure of the projection of $\mathscr{W}$ onto $\mathbb{C}^n$, and $\mathscr{W}$ is expected to be a complete intersection in generic coordinates. The *Lagrange multipliers* $L_1, \ldots, L_s$, are introduced through a system of homogeneous equations in $L_1, \ldots, L_s$, along with an affine linear form to dehomogenize them.

Definition. *Let* $\boldsymbol{L} = (L_1, \ldots, L_s)$ *be new indeterminants. Then* $\boldsymbol{Lagrange}(\boldsymbol{f}, i, (L_1, \ldots, L_s))$ *denotes the entries of the vector*

$$[L_1 \cdots L_s] \cdot \mathrm{jac}(\boldsymbol{f}, i),$$

where the Jacobian is in the variables $(X_{i+1}, \ldots, X_n)$.

We want to enforce that $\boldsymbol{l} = (0, \ldots, 0)$ is not included in any solutions; we add an affine linear form

$$\sum_{i=1}^{s} u_i \cdot L_i - 1$$

to dehomogenize them, for a randomly chosen point $\boldsymbol{u} = (u_1, \ldots, u_s) \subset \mathbb{C}^s$. We let $\mathscr{W}_{\boldsymbol{u}}(\pi_i, V)$ denote the variety

$$V\left(\boldsymbol{f}, \boldsymbol{Lagrange}(\boldsymbol{f}, i, (L_1, \ldots, L_s)), \sum_{i=1}^{s} u_i \cdot L_i - 1\right) \subset \mathbb{C}^{n+s},$$

again in indeterminants $\boldsymbol{X} = (X_1, \ldots, X_n)$ and $\boldsymbol{L} = (L_1, \ldots, L_s)$. We further let $\mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f})$ denote the polynomials

$$\left(\boldsymbol{f}, \boldsymbol{Lagrange}(\boldsymbol{f}, i, (L_1, \ldots, L_s)), \sum_{i=1}^{s} u_i L_i - 1\right)$$

in $\mathbb{C}[X_1, \ldots, X_n, L_1, \ldots, L_s]^{s+n-i+1}$, for $\boldsymbol{u} = (u_1, \ldots, u_s) \in \mathbb{C}^s$.

From Proposition 3.1 given in Chapter 3, for polynomials $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{C}[X_1, \ldots, X_n]$ and $\boldsymbol{u} \in \mathbb{C}^s$ chosen generically,

$$\pi_{\boldsymbol{X}}\left(\overline{\mathscr{W}_{\boldsymbol{u}}(\pi_i, V(\boldsymbol{f}))}\right) = W(\pi_i, V(\boldsymbol{f})),$$

for $i \in \{1, \ldots, n - s + 1\}$.

Example 2.2.3. *Consider again Example 2.2.2, but this time with the Lagrangian formulation. Consider again the polynomial*

$$f = X_1^2 + X_2^2 + X_3^2 - 1 \in \mathbb{C}[X_1, X_2, X_3]$$

and the sphere

$$V\left(X_1^2 + X_2^2 + X_3^2 - 1\right) \subset \mathbb{C}^3.$$

Since

$$\mathrm{jac}(X_1^2 + X_2^2 + X_3^2 - 1, 2) = 2X_3,$$

the Lagrange system defines the variety

$$V(X_1^2 + X_2^2 + X_3^2 - 1, LX_3, L - 1) = V(X_1^2 + X_2^2 - 1, X_3, L - 1).$$

The equations on the right hand side are a lexicographically ordered Gröbner basis of the ideal

$$\langle X_1^2 + X_2^2 + X_3^2 - 1, LX_3, L - 1\rangle.$$

Therefore

$$\pi_{\boldsymbol{X}}\left(V\left(X_1^2 + X_2^2 - 1, X_3, L - 1\right)\right)$$

describes the set where

$$X_1^2 + X_2^2 + X_3^2 - 1 = X_3 = 0$$

and therefore describes the polar variety $W(\pi_2, V(f))$.

Chapter 3

Bit complexity for computing one point in each connected component of a smooth real algebraic set

Bit complexity for computing one point in each connected component of a smooth real algebraic set

Jesse Elliott, Mark Giesbrecht, Éric Schost
David R. Cheriton School of Computer Science, University of Waterloo, On, Canada

Overview

We analyze the bit complexity of an algorithm for the computation of at least one point in each connected component of a smooth real algebraic set. This work is a continuation of our analysis of the hypersurface case (*On the bit complexity of finding points in connected components of a smooth real hypersurface*, ISSAC'20). In this paper, we extend the analysis to more general cases.

Let $F = (f_1, \ldots, f_p)$ in $\mathbb{Z}[X_1, \ldots, X_n]^p$ be a sequence of polynomials with $V = V(F) \subset \mathbb{C}^n$ a smooth and equidimensional variety and $\langle F \rangle \subset \mathbb{C}[X_1, \ldots, X_n]$ a radical ideal. To compute at least one point in each connected component of $V \cap \mathbb{R}^n$, our starting point is an algorithm by Safey El Din and Schost (*Polar varieties and computation of one point in each connected component of a smooth real algebraic set*, ISSAC'03). This algorithm uses random changes of variables that are proven to generically ensure certain desirable geometric properties. The cost of the algorithm was given in an algebraic complexity model; here, we analyze the bit complexity and the error probability, and we provide a quantitative analysis of the genericity statements. In particular, we are led to use Lagrange systems to describe polar varieties, as they make it simpler to rely on techniques such as weak transversality and an effective Nullstellensatz.

Keywords— Real algebraic geometry; weak transversality; Noether position; complexity

1 Introduction

Background and problem statement. Computing one point in each connected component of a real algebraic set S is a basic subroutine in real algebraic and semi-algebraic geometry; it is also useful in its own right, since it allows one to decide if S is empty or not.

We consider the case where S is given as $S = V \cap \mathbb{R}^n$, where $V = V(F) \subset \mathbb{C}^n$ is a complex algebraic set defined by a sequence of polynomials $F = (f_1, \ldots, f_p)$ in $\mathbb{Z}[X_1, \ldots, X_n]^p$. Algorithms for this task have been known for decades, and their complexity is to some extent well understood. Suppose that all f_i's have degree at most d, and coefficients of bit-size at most b. Without making any assumption on these polynomials, the algorithm given in [8, Section 13.1] solves our problem using $d^{O(n)}$ operations in $\mathbb{Q}$; in addition, the output of the

algorithm is represented by polynomials of degree $d^{O(n)}$, with coefficients of bit-size $hd^{O(n)}$. The key idea behind this algorithm goes back to [22]: sample points are found through the computation of critical points of well-chosen functions on V.

The number of connected components of V admits the lower bound $d^{\Omega(n)}$, so up to polynomial factors this result is optimal. However, due to the generality of the algorithm, the constant hidden in the exponent $O(n)$ in its runtime turns out to be rather large: the algorithm relies on infinitesimal deformations, that affect runtime non-trivially.

In this paper, we will work under the additional assumption that $V = V(f_1, \ldots, f_p)$ is a *smooth* complex algebraic set, equidimensional of dimension $\delta = n - p$, and that $f_1, \ldots, f_p$ generate a radical ideal (we explain these terms in the next section). We place ourselves in the continuation of the line of work initiated by [5]: that reference deals with cases where V is a smooth hypersurface and $V \cap \mathbb{R}^n$ is compact, pointing out how *polar varieties* (that were introduced in the 1930's in order to define characteristic classes [32, 41]) can play a role in effective real geometry. This paper was extended in several directions: to V being a smooth complete intersection, still with $V \cap \mathbb{R}^n$ compact [6], then without the compactness assumption [35, 7]; the smoothness assumption was then partly dropped in [2, 4].

Our starting point is the algorithm in [35], whose assumptions are slightly more general than ours (V is not required to have dimension $\delta = n - p$). In the cases we consider in this paper, its runtime is $\binom{n}{p}^{2+o(1)} d^{(4+o(1))n}$ operations in $\mathbb{Q}$. As with many results in this vein, the algorithm is randomized, as we need to assume that we are in generic coordinates; this is done by applying a random change of coordinates prior to all computations. In addition, the algorithm relies on procedures for solving systems of polynomial equations that are themselves randomized. Altogether, we choose $n^{O(1)}$ random vectors, each of them in an affine space of dimension $n^{O(1)}$; every time a choice is made, there exists a hypersurface of the parameter space that one has to avoid in order to guarantee success. In this paper, we revisit this algorithm, modify it in part, and give a complete analysis of its probability of success and its bit complexity.

This work is a continuation of the analysis of the hypersurface case that we gave in [17] (that is, the case $p = 1$). A very useful property in the hypersurface case is that polar varieties can be described by straightforward equations (the partial derivatives of the input polynomial) that form a regular sequence, at least in generic coordinates. In higher codimension, this is not the case anymore: the natural description of polar varieties now involves minors of the Jacobian matrix of the input equations (this is the approach used in [35]). The resulting equations are in general not a complete intersection anymore, which makes it impossible to extend directly several arguments we used in [17].

Our solution is to use a description of polar varieties by means of so-called Lagrange equations. These equations are complete intersections (in generic coordinates), but they involve more variables. As such, they describe algebraic sets that cover polar varieties; we will discuss in detail the relationship between these two presentations, using in particular several results from [3, 37].

Data structures. The output of the algorithm is a finite set in $\overline{\mathbb{Q}}^n$. To represent it, we rely on a widely used data structure based on univariate polynomials [28, 29, 18, 21, 1, 19, 20, 33]. For a zero-dimensional algebraic set $S \subset \mathbb{C}^n$ defined over $\mathbb{Q}$, a *zero-dimensional parameterization* $\mathscr{Q} = ((q, v_1, \ldots, v_n), \lambda)$ of S consists in polynomials $(q, v_1, \ldots, v_n)$, such that $q \in \mathbb{Q}[T]$ is monic and squarefree, all v_i's are in $\mathbb{Q}[T]$ and satisfy $\deg(v_i) < \deg(q)$, and in a $\mathbb{Q}$-linear form λ in variables $X_1, \ldots, X_n$, such that

- $\lambda(v_1, \ldots, v_n) = Tq' \bmod q$;
- we have the equality $S = \left\{ \left(\frac{v_1(\tau)}{q'(\tau)}, \ldots, \frac{v_n(\tau)}{q'(\tau)} \right) \mid q(\tau) = 0 \right\}$.

The constraint on λ says that the roots of q are the values taken by λ on S. The parameterization of the coordinates by rational functions having q' as a denominator goes back to [28, 29]: as pointed out in [1], it allows one to control precisely the size of the coefficients of $v_1, \ldots, v_n$.

Main result. To state our main result, we need to define the *height* of a rational number, and of a polynomial with rational coefficients.

The *height* of a non-zero $a = u/v \in \mathbb{Q}$ is the maximum of $\ln(|u|)$ and $\ln(v)$, where $u \in \mathbb{Z}$ and $v \in \mathbb{N}$ are coprime. For a polynomial f with rational coefficients, if $v \in \mathbb{N}$ is the minimal common denominator of all non-zero coefficients of f, then the *height* $\mathrm{ht}(f)$ of f is defined as the maximum of the logarithms of v and of the absolute values of the coefficients of vf.

Theorem 1.1. *Let $F = (f_1, \ldots, f_p) \in \mathbb{Z}[X_1, \ldots, X_n]^p$ be a sequence of polynomials with $\deg(f_i) \leq d$ and $\mathrm{ht}(f_i) \leq b$. Suppose that the ideal generated by $f_1, \ldots, f_p$ is radical and that $V = V(F) \subset \mathbb{C}^n$ is smooth and equidimensional of dimension $n - p$. Also suppose that $0 < \epsilon < 1$.*

There exists a randomized algorithm that takes F and ϵ as input and produces $n - p + 1$ zero-dimensional parameterizations, the union of whose zeros includes at least one point in each connected component of $V(F) \cap \mathbb{R}^n$, with probability at least $1 - \epsilon$. Otherwise, the algorithm either returns a proper subset of the points, or FAIL. In any case, the algorithm uses

$$O^{\sim}(d^{3n+2p+1} \log(1/\epsilon)(b + \log(1/\epsilon)))$$

bit operations. The polynomials in the output have degree at most d^{n+p}, and height

$$O^{\sim}(d^{n+p+1}(b + \log(1/\epsilon))).$$

Remark 1.2. *When $d = 2$, that is, in the quadratic case, we obtain notably better complexity (see Section 9.2 for details). We thank Mohab Safey El Din for pointing this out to us.*

Here we assume that F is given as a sequence of polynomials in dense representation. Following references such as [21, 19, 20, 5, 35], it would be possible to refine the runtime estimate by assuming that F is given by a *straight-line program* (that is, a sequence of operations $+, -, \times$ that takes as input $X_1, \ldots, X_n$ and evaluates F). Any polynomial of degree d in n

variables can be computed by a straight-line program that does $O(d^n)$ operations: evaluate all monomials of degree up to d in n variables, multiply them by their respective coefficients and sum the results. However, some inputs may be given by a shorter straight-line program, and the algorithm would actually benefit from this.

The algorithm itself is rather simple. To describe it, we need to define *polar varieties*, which will play a crucial role in this paper. Let $V = V(F)$, for $F = (f_1, \ldots, f_p)$ as in the theorem. For $i \in \{1, \ldots, n-1\}$, denote by $\pi_i : \mathbb{C}^n \to \mathbb{C}^i$ the projection $(x_1, \ldots, x_n) \mapsto (x_1, \ldots, x_i)$. The i-th *polar variety*

$$W(i, F) := \{\boldsymbol{x} \in V \mid \dim \pi_i(T_{\boldsymbol{x}}V) < i\}$$

is the set of critical points of π_i on V. We will recall below that it is defined by the vanishing of all p-minors $M_{i,1}, \ldots, M_{i,S_i}$ of the last $n-i$ columns of the Jacobian matrix of F, together with the equations F themselves (here, S_i is simply the binomial number $\binom{n-i}{p}$).

In general, we cannot say much about the geometry of $W(i, F)$, but if we apply a generic change of coordinates $\boldsymbol{A}$ to F, then $W(i, F)$ is known to be equidimensional of dimension $(i-1)$ or empty [5, 3, 37], and to be in so-called *Noether position* [35] (background notions in algebraic geometry are in [30, 39, 16]; we will recall key definitions). If this is the case, the algorithm in [35] chooses arbitrary $\sigma_1, \ldots, \sigma_{n-1}$ in $\mathbb{Q}$ and solves the systems defined by

$$X_1 - \sigma_1 = \cdots = X_{i-1} - \sigma_{i-1} = f_1 = \cdots = f_p = M_{i,1} = \cdots = M_{i,S_i} = 0 \tag{1}$$

for $i = 1, \ldots, n-p+1$. They all admit finitely many solutions, and Theorem 2 in [35] proves that the union of their solution sets contains one point on each connected component of $V \cap \mathbb{R}^n$.

One of our contributions is to analyze precisely what conditions on the change of coordinates $\boldsymbol{A}$ guarantee success. This is done by revisiting the key ingredients in the proofs given in [3, 35], and giving quantitative versions of these results, bounding the degrees of the hypersurfaces we have to avoid.

We actually do not solve the equations (1), since the (large) number of minors S_i makes this analysis difficult. Instead, we replace (1) by equations involving Lagrange multipliers. Proving correctness requires us to guarantee further genericity properties, but once this is done, we can rely on the algorithm in [38] to solve these equations, for which a complete bit complexity analysis is available.

Further work. This paper is an extension of [17], where the analysis was done for the hypersurface case. In addition, this work should also be seen as a step toward the analysis of further randomized algorithms in real algebraic geometry. In particular, randomized algorithms for deciding *connectivity queries* on smooth, compact algebraic sets have been developed in a series of papers [36, 9], and could be revisited using the techniques introduced here. The techniques would apply to algorithms in real algebraic geometry where transversality or Noether position are required geometric properties established by a random change of coordinates.

Outline. The next section summarizes the main concepts from algebraic geometry needed in this paper. In Section 3, we compare the descriptions of determinantal varieties by the vanishing of matrix minors, and through the use of Lagrange multipliers; these results, while rather simple, are used throughout. A first application is in Section 4, where we give a quantitative form of Thom's "weak transversality lemma".

Section 5 introduces polar varieties and discusses the algorithm sketched above and the genericity conditions required for it to succeed. These conditions are studied in detail in Sections 6, 7 and 8; this allows us to complete the analysis of the algorithm in Section 9, thereby proving Theorem 1.1.

2 Preliminaries

In this section, we gather several basic definitions and properties of algebraic sets and locally closed sets. General references for this material are [30, 39, 16].

Algebraic sets. An algebraic set $V \subset \mathbb{C}^n$ is the set of common zeros of an ideal I in $\mathbb{C}[X_1, \ldots, X_n]$. Conversely, the ideal of a subset V of $\mathbb{C}^n$, that is, the set of polynomials in $\mathbb{C}[X_1, \ldots, X_n]$ that vanish at all points of V, is called the *ideal* of V; this is a radical ideal, which we write $I(V)$.

The smallest algebraic set containing an arbitrary set Y is called the *Zariski closure* of Y and written $\overline{Y}$.

Irreducible decomposition. An algebraic set $V \subset \mathbb{C}^n$ is *irreducible* when $V = V_1 \cup V_2$, with V_1, V_2 algebraic sets, implies $V = V_1$ or $V = V_2$; this is the case if and only if $I(V)$ is prime. An algebraic set $V \subset \mathbb{C}^n$ can be decomposed into a finite union of irreducible algebraic sets

$$V = V_1 \cup V_2 \cup \cdots \cup V_r,$$

with $V_i \not\subset V_j$ for all $i \neq j$. The sets $V_1, \ldots, V_r$ are called the *irreducible components* of V; they are uniquely defined, up to order. In terms of ideals, $I(V)$ being radical, it admits a decomposition as an irredundant intersection of prime ideals $I_1, \ldots, I_r$; the irreducible algebraic sets $V(I_1), \ldots, V(I_r)$ are the irreducible components of V.

Dimension. The *dimension* of an algebraic set $V \subset \mathbb{C}^n$, denoted $\dim(V)$, can be defined as the unique integer d such that $V \cap H_1 \cap \cdots \cap H_d$ is finite, but not empty, for a generic choice of hyperplanes $H_1, \ldots, H_d$. The *codimension* of V is $n - \dim(V)$.

An algebraic set V is *equidimensional* if each of its irreducible components has the same dimension; if each component has dimension d then we say that V is d-equidimensional.

Degree. We use the definition of degree from [24]: the *degree* $\deg(V)$ of an irreducible algebraic set V is the number of intersection points between itself and $\dim(V)$ generic hy-

perplanes, and the degree of an arbitrary algebraic set is defined as the sum of the degrees of its irreducible components.

The degree of a hypersurface defined by a squarefree polynomial f is $\deg(f)$. We particularly care about algebraic sets of dimension zero; by definition, these sets are finite and their degree is equal to their cardinality.

We will often apply the Bézout bound from [24, Theorem 1], which says that $\deg(V \cap V') \leq \deg(V)\deg(V')$ holds for all algebraic sets V, V'. A last useful property is that for any linear mapping $\psi : \mathbb{C}^n \to \mathbb{C}^m$, $\deg(\overline{\psi(V)}) \leq \deg(V)$.

Noether position. Suppose that the ambient dimension n is fixed. For i in $\{1, \ldots, n\}$, let π_i denote the projection

$$\begin{aligned} \mathbb{C}^n &\to \mathbb{C}^i \\ (x_1, \ldots, x_n) &\mapsto (x_1, \ldots, x_i). \end{aligned}$$

A d-equidimensional algebraic set $V \subset \mathbb{C}^n$ is in *Noether position* for the projection π_d when the extension

$$\mathbb{C}[X_1, \ldots, X_d] \to \mathbb{C}[X_1, \ldots, X_n]/I(V)$$

is injective and integral; here, $I(V) \subset \mathbb{C}[X_1, \ldots, X_n]$ is the defining ideal of V. It is then a consequence that for any $\boldsymbol{x}$ in $\mathbb{C}^d$, the fiber $V \cap \pi_d^{-1}(\boldsymbol{x})$ has dimension zero and is thus finite and not empty.

Gradient vectors and Jacobian matrices. The gradient vector of a polynomial $f \in \mathbb{C}[X_1, \ldots, X_n]$ is written $\mathrm{grad}(f) \in \mathbb{C}[X_1, \ldots, X_n]^{1\times n}$ (so this is a row vector). Most of the time, the variables with respect to which we differentiate are clear from the context, but we may write $\mathrm{grad}_{\boldsymbol{X}}(f)$ for clarity, with $\boldsymbol{X} = X_1, \ldots, X_n$.

The Jacobian matrix of polynomials $F = f_1, \ldots, f_s$ is the $s \times n$ matrix $\mathrm{jac}(F)$, with $\partial f_i/\partial X_j$ at entry (i, j), for $1 \leq i \leq s$ and $1 \leq j \leq n$. As we do for gradients, we will write $\mathrm{jac}_{\boldsymbol{X}}(F)$ if we want to highlight what variables we differentiate with respect to.

Given $\boldsymbol{x}$ in $\mathbb{C}^n$, we then write $\mathrm{grad}(f, \boldsymbol{x})$, resp. $\mathrm{jac}(F, \boldsymbol{x})$, for the evaluation of respectively $\mathrm{grad}(f)$ and $\mathrm{jac}(F)$ at $\boldsymbol{x}$.

Tangent spaces, regular and singular points. Assume that $V \subset \mathbb{C}^n$ is a d-equidimensional algebraic set. The *Zariski-tangent space* to V at $\boldsymbol{x} \in V$ is the vector space $T_{\boldsymbol{x}}V \subset \mathbb{C}^n$ defined by the equations

$$\mathrm{grad}(g, \boldsymbol{x}) \cdot \boldsymbol{v} = 0 \text{ for all } g \in I(V), \quad \boldsymbol{v} \in \mathbb{C}^{n\times 1}.$$

Then, the point $\boldsymbol{x} \in V$ is a *regular point* (or non-singular) if $\dim(T_{\boldsymbol{x}}V) = d$; otherwise, $\boldsymbol{x}$ is a *singular point*. We let $\mathrm{reg}(V)$ and $\mathrm{sing}(V)$ respectively denote the regular and singular points of V; when the latter is empty, we say that V is *smooth*. If $I(V)$ is generated by polynomials $G = (g_1, \ldots, g_s) \in \mathbb{C}[X_1, \ldots, X_n]^s$, then at any point $\boldsymbol{x}$ of $\mathrm{reg}(V)$, the Jacobian matrix $\mathrm{jac}(G, \boldsymbol{x})$ has rank $n - d$ and the right kernel of $\mathrm{jac}(G, \boldsymbol{x})$ is $T_{\boldsymbol{x}}V$.

Changes of variables. For a matrix $\boldsymbol{A}$ in $\mathbb{C}^{n\times n}$ and a polynomial g in $\mathbb{C}[X_1,\ldots,X_n]$, we write

$$g^{\boldsymbol{A}} := g(\boldsymbol{AX}) \in \mathbb{C}[X_1,\ldots,X_n],$$

where $\boldsymbol{X}$ is the column vector with entries $X_1,\ldots,X_n$. Similarly, for a sequence of polynomials $G = (g_1,\ldots,g_s)$ in $\mathbb{C}[X_1,\ldots,X_n]^s$, we write $G^{\boldsymbol{A}} = \left(g_1^{\boldsymbol{A}},\ldots,g_s^{\boldsymbol{A}}\right)$. For an algebraic set $V \subset \mathbb{C}^n$ and a matrix $\boldsymbol{A} \in \mathrm{GL}(n)$, we define $V^{\boldsymbol{A}}$ as the image of V by the map $\phi_{\boldsymbol{A}} : \boldsymbol{x} \mapsto \boldsymbol{A}^{-1}\boldsymbol{x}$. Notice in particular that $V(G^{\boldsymbol{A}}) = \phi_{\boldsymbol{A}}(V(G)) = V(G)^{\boldsymbol{A}}$.

Locally closed sets. We will also need to work with *locally closed* sets: we say that $Y \subset \mathbb{C}^n$ is locally closed if we can write it as $Y = V - V'$, for some algebraic sets V, V'.

The notions of dimension and equidimensionality carry over to this context (they are defined through the Zariski closure of Y), as does that of tangent space: for $\boldsymbol{x}$ in Y, we set $T_{\boldsymbol{x}}Y = T_{\boldsymbol{x}}V$ (this is independent of the choice of V, V' in the definition above). If Y is equidimensional, as we did for algebraic sets, we can then define the *regular points* (or non-singular points) of Y as those points at which the tangent space has dimension d, and we say that Y is smooth if all its points are regular.

Open sets are locally closed. As another example, for any d-equidimensional algebraic set V, $\mathrm{reg}(V)$ is a smooth d-equidimensional locally closed set.

3 Describing determinantal varieties

In this section, we work with polynomials in $\mathbb{C}[Y_1,\ldots,Y_N]$, for some positive integer N. Given a matrix $\boldsymbol{A}$ in $\mathbb{C}[Y_1,\ldots,Y_N]^{q\times r}$, with $q \leq r$, together with some equations $B = (b_1,\ldots,b_s)$ in $\mathbb{C}[Y_1,\ldots,Y_N]$, we consider the locus S defined as

$$S = \{\boldsymbol{y} \in \mathbb{C}^N \;\mid\; b_1(\boldsymbol{y}) = \cdots = b_s(\boldsymbol{y}) = 0 \text{ and } \mathrm{rank}(\boldsymbol{A}(\boldsymbol{y})) < q\}.$$

One of our goals here is to give a degree bound for S; this will be used twice, in the next section for our discussion of the weak transversality lemma (in a slightly more general context where we work in an open subset of $\mathbb{C}^N$), then also to control the degrees of the systems of equations we will solve.

Consider the polynomials

$$\mathfrak{J}(\boldsymbol{A}, B) = (b_1,\ldots,b_s,M_1,\ldots,M_P),$$

where $M_1,\ldots,M_P$ are the q-minors of $\boldsymbol{A}$, with $P = \binom{r}{q}$. Since we have $V(\mathfrak{J}(\boldsymbol{A},B)) = S$, we may derive a degree bound on S using the Bézout inequality. However, even the refined form given in [25, Proposition 2.3] involves an exponential dependency in either the ambient dimension N or the number of minors P. This might be acceptable in some contexts (such as when estimating the degrees of polar varieties), but is way beyond our target bound in the context of weak transversality, for instance.

Instead, we use Lagrange systems. We let $L_1, \dots, L_q$ be new variables, thought of as Lagrange multipliers, and consider the "Lagrange polynomials" given as the r entries of $[L_1 \ \cdots \ L_q] \cdot \boldsymbol{A}$. We denote by $Z \subset \mathbb{C}^{N+q}$ the algebraic set defined by the vanishing of

$$(b_1, \dots, b_s, \ [L_1 \ \cdots \ L_q] \cdot \boldsymbol{A})$$

and by Z' the algebraic set

$$Z' := \overline{Z - \{(\boldsymbol{y}, 0, \dots, 0) \in \mathbb{C}^{N+q} \mid (\boldsymbol{y}, 0, \dots, 0) \in Z\}},$$

where the bar denotes Zariski closure (we have to remove such points, since $L_1 = \cdots = L_q = 0$ is always a trivial solution to the Lagrange equations). Finally, consider the projection

$$\begin{aligned} \mu : \ \mathbb{C}^{N+q} &\to \mathbb{C}^N \\ (\boldsymbol{y}, \boldsymbol{\ell}) \ &\mapsto \boldsymbol{y}. \end{aligned}$$

It is then possible to prove that S is the Zariski closure of $\mu(Z')$, and derive degree bounds using the equations defining Z. However, while introducing Z' is convenient, computing defining equations for it is non-trivial, as it involves saturation; besides, in several contexts, it will be advantageous to work with equations in complete intersection, which the following construction will guarantee in certain cases. For $\boldsymbol{u} = (u_1, \dots, u_q) \in \mathbb{C}^q$, consider the equations

$$\mathfrak{L}(\boldsymbol{A}, B, \boldsymbol{u}) = (b_1, \dots, b_s, \ [L_1 \ \cdots \ L_q] \cdot \boldsymbol{A}, \ u_1 L_1 + \cdots + u_q L_q - 1),$$

and let $Z_{\boldsymbol{u}} \subset \mathbb{C}^{N+q}$ be its zero-set. Using the linear equation $u_1 L_1 + \cdots + u_q L_q - 1$ allows us to discard solutions where $L_1 = \cdots = L_q = 0$, but unlucky choices of $\boldsymbol{u}$ may discard other components as well. The following proposition makes this more precise.

Proposition 3.1. *For any $\boldsymbol{u}$ in $\mathbb{C}^q$, we have the inclusion $\mu(Z_{\boldsymbol{u}}) \subset S$. There exists a non-empty open set $\mathscr{O} \subset \mathbb{C}^q$ such that for $\boldsymbol{u}$ in $\mathscr{O}$, we have the inclusion $S \subset \overline{\mu(Z_{\boldsymbol{u}})}$, and thus the equalities $S = \overline{\mu(Z_{\boldsymbol{u}})}$ and*

$$\sqrt{\langle \mathfrak{L}(\boldsymbol{A}, B, \boldsymbol{u}) \rangle \cap \mathbb{C}[Y_1, \dots, Y_N]} = \sqrt{\langle \mathfrak{J}(\boldsymbol{A}, B) \rangle}.$$

The set $\mathscr{O}$ is the complement of at most $\deg(S)$ hyperplanes.

Proof. If $(\boldsymbol{y}, \boldsymbol{\ell})$ cancels all polynomials in $\mathfrak{L}(\boldsymbol{A}, B, \boldsymbol{u})$, then $\boldsymbol{\ell}$ is non-zero, so that $\boldsymbol{A}(\boldsymbol{y})$ is rank-deficient. As a consequence, $\boldsymbol{y}$ is in S. This proves the first assertion.

For the second one, let $S_1, \dots, S_K$ be the irreducible components of S. For any given k in $\{1, \dots, K\}$, since all q-minors of $\boldsymbol{A}$ vanish on S_k, they vanish in the function field $\mathbb{C}(S_k)$, so $\boldsymbol{A}$ has rank less than q as a matrix over $\mathbb{C}(S_k)$. Thus, there exists a non-zero vector of rational functions

$$\boldsymbol{\ell}_k = (\ell_{k,1}, \dots, \ell_{k,q}) = \left(\frac{N_{k,1}}{D_k}, \dots, \frac{N_{k,q}}{D_k} \right) \in \mathbb{C}(S_k)^q,$$

such that $\boldsymbol{\ell}_k \cdot \boldsymbol{A} = 0$ in $\mathbb{C}(S_k)^r$ (here, we see $\boldsymbol{\ell}_k$ in $\mathbb{C}(S_k)^{1\times q}$). For definiteness, assume that $N_{k,\iota_k} \neq 0$. Then, in particular, $S'_k = S_k - V(D_k N_{k,\iota_k})$ is dense in S_k; for $\boldsymbol{y}$ in S'_k, $\boldsymbol{\ell}_k(\boldsymbol{y})$ is well-defined, non-zero, and still satisfies $\boldsymbol{\ell}_k(\boldsymbol{y}) \cdot \boldsymbol{A}(\boldsymbol{y}) = 0$.

Then, pick a point $\boldsymbol{y}_k$ in S'_k, so that $\boldsymbol{\ell}_k(\boldsymbol{y}_k)$ is a well-defined, non-zero vector in $\mathbb{C}^q$. This allows us to define a non-empty Zariski open set $\mathscr{O}_k \subset \mathbb{C}^q$ by the condition

$$\mathscr{O}_k := \{\boldsymbol{u} \in \mathbb{C}^q \mid \boldsymbol{\ell}_k(\boldsymbol{y}_k) \cdot \boldsymbol{u} \neq 0\},$$

where in the dot product we take $\boldsymbol{\ell}_k(\boldsymbol{y}_k)$ in $\mathbb{C}^{1\times q}$ and $\boldsymbol{u}$ in $\mathbb{C}^{q\times 1}$. Finally, we let $\mathscr{O} := \cap_{1\leq k\leq K}\mathscr{O}_k$, which is open, non-empty, and defined as the complement of $K \leq \deg(S)$ hyperplanes, as claimed. We now prove that for $\boldsymbol{u}$ in $\mathscr{O}$, the inclusion $S \subset \overline{\mu(Z_{\boldsymbol{u}})}$ holds.

For this, we take k as above, and we prove that S_k is contained in $\overline{\mu(Z_{\boldsymbol{u}})}$. Consider the rational mapping

$$\begin{aligned} S'_k &\to \mathbb{C} \\ \boldsymbol{y} &\mapsto \boldsymbol{\ell}_k(\boldsymbol{y}) \cdot \boldsymbol{u} = \frac{u_1 N_{k,1}(\boldsymbol{y}) + \cdots + u_q N_{k,q}(\boldsymbol{y})}{D_k(\boldsymbol{y})}. \end{aligned}$$

Put $S''_k = S'_k - V(u_1 N_{k,1} + \cdots + u_q N_{k,q})$; this is again an open subset of S_k, and the fact that $\boldsymbol{\ell}_k(\boldsymbol{y}_k) \cdot \boldsymbol{u}$ is non-zero, with $\boldsymbol{y}_k$ in S'_k, shows that S''_k is not empty. In particular, it is dense in S_k.

Take $\boldsymbol{y}$ in S''_k. Then, $\alpha := \boldsymbol{\ell}_k(\boldsymbol{y}) \cdot \boldsymbol{u}$ is non-zero, set we can define $\boldsymbol{\ell}' := 1/\alpha\ \boldsymbol{\ell}_k(\boldsymbol{y})$. Then, $\boldsymbol{\ell}'$ is still in the left nullspace of $\boldsymbol{A}(\boldsymbol{y})$, and by construction $\boldsymbol{\ell}' \cdot \boldsymbol{u} = 1$, so that $(\boldsymbol{y}, \boldsymbol{\ell}')$ is in $Z_{\boldsymbol{u}}$. In other words, S''_k is contained in $\mu(Z_{\boldsymbol{u}})$. Taking the Zariski closure, we obtain that S_k is contained in $\overline{\mu(Z_{\boldsymbol{u}})}$, as claimed. The equality $S = \overline{\mu(Z_{\boldsymbol{u}})}$ follows, as does the claimed equality between ideals. □

Corollary 3.2. *If all polynomials $b_1, \ldots, b_s$ have respective degrees at most $d_1, \ldots, d_s$, and all entries of $\boldsymbol{A}$ have degree at most d', then the degree of S is at most $d_1 \cdots d_s (d'+1)^r$.*

Proof. Choose $\boldsymbol{u}$ in the set $\mathscr{O}$ of the previous lemma. The algebraic set $Z_{\boldsymbol{u}}$ is defined by s equations of respective degrees at most $d_1, \ldots, d_s$, r equations of degree at most $d'+1$ and a linear equation. It follows from Bézout's Theorem [24] that $\deg(Z_{\boldsymbol{u}}) \leq d_1 \cdots d_s (d'+1)^r$. Degree does not increase through projection, so the conclusion follows from the previous lemma. □

Remark 3.3. *In the next section, we will consider the following slight variant of the problem considered here, where we are interested in the locally closed set*

$$S' = \{\boldsymbol{y} \in \Omega \ \mid\ b_1(\boldsymbol{y}) = \cdots = b_s(\boldsymbol{y}) = 0 \text{ and } \operatorname{rank}(\boldsymbol{A}(\boldsymbol{y})) < q\},$$

for some Zariski open set $\Omega \subset \mathbb{C}^N$. The Zariski closure $\overline{S'}$ is the union of certain irreducible components of the set S defined above, so the degree bound of Corollary 3.2 still holds for $\overline{S'}$.

We note that in some cases, sharper bounds are known for the degrees of determinantal varieties: for instance, when X is finite, and defined as the set of critical points on a smooth algebraic set [34], or when we want to take into account differences in the degrees of the rows and columns of $\boldsymbol{A}$ [40, 31, 23].

4 Weak transversality

Several of the generic properties of polar varieties are consequences of *weak transversality*, which is an important extension of Sard's lemma due to Thom (this observation goes back to work of Giusti, Heintz and collaborators [5, 4]). In this section, we develop a quantitative extension of Thom's weak transversality theorem, specialized to the particular case of transversality to a point. In the sequel, we will apply this result to bound the degree of particular hypersurfaces our algorithm needs to avoid to guarantee success.

4.1 Definitions and statement of the result

In its differential version, Sard's lemma states that the set of critical values of a smooth function $\mathbb{R}^n \to \mathbb{R}^m$ has measure zero; extensions exist to smooth mapping between differential manifolds. In our algebraic context, we will use the following definitions.

Consider a polynomial mapping $\Phi : Y \to \mathbb{C}^m$ from a smooth n-equidimensional locally closed set Y to $\mathbb{C}^m$, with $m \leq n$. A *critical point* of Φ is a point $\boldsymbol{y} \in Y$ for which the image of the tangent space $T_{\boldsymbol{y}}Y$ by the Jacobian matrix $\mathrm{jac}(\Phi, \boldsymbol{y})$ has dimension less than m. For instance, the case that will interest us in this section is when Y is Zariski open in $\mathbb{C}^n$, in which case we have $T_{\boldsymbol{y}}Y = \mathbb{C}^n$ for all $\boldsymbol{y}$ in Y, and the condition is equivalent to the Jacobian of Φ having rank less than m at $\boldsymbol{y}$. *Critical values* are the images by Φ of critical points; the complement of this set are the *regular values*. Notice then, a regular value is not necessarily in the image of Φ.

One can then give "algebraic" versions of Sard's lemma: for instance, [30, (3.7)] shows that for Y an irreducible algebraic set and Φ dominant, the critical values of Φ are contained in a strict algebraic subset of $\mathbb{C}^m$; below, we will rely on a straightforward generalization given in [37]. See also [11, Chapter 9] for the semi-algebraic case.

Thom's weak transversality lemma, as given for instance in [14], generalizes Sard's lemma. In this section, we consider a particular case of this result (transversality to a point), and establish a quantitative version of it.

Let n, s, and m be positive integers, with $m \leq n$ as before, let $\mathscr{O}$ be a Zariski open subset of $\mathbb{C}^n$, and denote by $\Phi : \mathscr{O} \times \mathbb{C}^s \to \mathbb{C}^m$ a mapping given by polynomials in $n + s$ indeterminates $X_1, \ldots, X_n, \Theta_1, \ldots, \Theta_s$ (the latter should be thought of as parameters). For $\boldsymbol{\vartheta}$ in $\mathbb{C}^s$, we let $\Phi_{\boldsymbol{\vartheta}} : \mathscr{O} \to \mathbb{C}^m$ be the induced mapping $\boldsymbol{x} \mapsto \Phi(\boldsymbol{x}, \boldsymbol{\vartheta})$. Thom's weak transversality lemma tells us that if 0 is a regular value of the mapping Φ, then 0 remains a regular value of the induced mapping $\Phi_{\boldsymbol{\vartheta}}$ for a generic $\boldsymbol{\vartheta}$. (Here, we are dealing with the particular case of transversality to a point, which can be rephrased entirely in terms of regular and critical values.) Our quantitative version of this result is the following.

Proposition 4.1 (Weak transversality)**.** *Let all notation be as before, and suppose that Φ is defined by m polynomials of degree at most d. If 0 is a regular value of Φ, there exists a non-zero polynomial $\Gamma \in \mathbb{C}[\Theta_1, \ldots, \Theta_s]$ of degree at most d^{m+n} such that for $\boldsymbol{\vartheta}$ in $\mathbb{C}^s$, if $\Gamma(\boldsymbol{\vartheta}) \neq 0$, then 0 is a regular value of $\Phi_{\boldsymbol{\vartheta}}$.*

Example 4.2. *Consider a squarefree polynomial f in $\mathbb{C}[X_1, X_2]$, with degree at most d, defining a smooth curve $V(f)$ in $\mathbb{C}^2$, and let the mapping $\Phi : \mathbb{C}^2 \times \mathbb{C} \to \mathbb{C}^2$ be defined by $\Phi(X_1, X_2, \Theta) = (f(X_1, X_2), X_1 - \Theta)$ (so $m = n = 2$ and $s = 1$). One checks that the Jacobian of Φ with respect to (X_1, X_2, Θ) has full rank two at any point in $\Phi^{-1}(0)$, so that 0 is a regular value of Φ and therefore the assumptions of the proposition apply.*

We then deduce that a non-zero polynomial $\Gamma \in \mathbb{C}[\Theta]$ exists, with degree at most d^4 with the property that, if ϑ in $\mathbb{C}$ does not cancel Γ then 0 is a regular value of the induced mapping Φ_ϑ. In particular, for all ϑ in $\mathbb{C}$ except at most d^4 values, the ideal $(f(X_1, X_2), X_1 - \vartheta)$ is radical in $\mathbb{C}[X_1, X_2]$; equivalently, $f(\vartheta, X_2)$ is squarefree.

In this example, we could of course obtain the same result (with a sharper degree bound) by considering the discriminant of f with respect to X_2, but the construction above will be useful later on, in a generalized form. (In this example, the bound d^4 could be sharpened by utilizing the fact that only one of the polynomials defining Φ has degree d, whereas the other one is linear.)

The rest of the section is devoted to the proof of the proposition. The proof of [37, Theorem B.3] already shows the existence of Γ; it is essentially the classical proof for smooth mappings [14, Section 3.7], written in an algebraic context. In what follows, we revisit this proof, establishing a bound on the degree of Γ.

4.2 Proof of the proposition

In what follows, we use the notation of Proposition 4.1, so that we consider m polynomials Φ that depend on variables $X_1, \ldots, X_n$ and $\Theta_1, \ldots, \Theta_s$, with $m \leq n$, and an open set $\mathscr{O} \subset \mathbb{C}^n$.

In the context of Thom's weak transversality, the "bad" parameter values show up as the critical values of a certain projection. Put $Y = \Phi^{-1}(0) \cap (\mathscr{O} \times \mathbb{C}^s)$, and let V be the Zariski closure of Y. If Y is empty, there is nothing to do, since all values $\boldsymbol{\vartheta}$ in $\mathbb{C}^s$ satisfy the conclusion of the proposition. We therefore assume that Y is not empty. Take $(\boldsymbol{x}, \boldsymbol{\vartheta})$ in Y; then by assumption, $\mathrm{jac}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta}))$ has full rank m. Since in a neighbourhood of $(\boldsymbol{x}, \boldsymbol{\vartheta})$, V coincides with $Y = \Phi^{-1}(0) \cap (\mathscr{O} \times \mathbb{C}^s)$, the Jacobian criterion [16, Corollary 16.20] implies that there is a unique irreducible component $V_{(\boldsymbol{x},\boldsymbol{\vartheta})}$ of V that contains $(\boldsymbol{x}, \boldsymbol{\vartheta})$, that $(\boldsymbol{x}, \boldsymbol{\vartheta})$ is regular on this component and that $\dim V_{(\boldsymbol{x},\boldsymbol{\vartheta})} = n + s - m$. This implies that Y is a smooth, $(n + s - m)$-equidimensional locally closed set. Now, consider the projection

$$\begin{aligned} \pi : \mathbb{C}^{n+s} &\to \mathbb{C}^s \\ (\boldsymbol{x}, \boldsymbol{\vartheta}) &\mapsto \boldsymbol{\vartheta}, \end{aligned}$$

and let Z be the set of critical points of the restriction $\pi_{|Y}$ of π to Y; that is,

$$Z := \{(\boldsymbol{x}, \boldsymbol{\vartheta}) \in Y \mid \dim(\pi(T_{\boldsymbol{x},\boldsymbol{\vartheta}} Y)) < s\}.$$

The projection $\pi(Z) \subset \mathbb{C}^s$ is thus the set of critical values of $\pi_{|Y}$.

Lemma 4.3. *The Zariski closure $\overline{\pi(Z)}$ is a strict subset of $\mathbb{C}^s$.*

Proof. The discussion above implies that $\operatorname{reg}(V)$ is a smooth, $(n+s-m)$-equidimensional locally closed set containing Y. Let then Z' be the critical points of $\pi_{|\operatorname{reg}(V)}$; by the algebraic form of Sard's lemma as given in [30, Theorem 3.7] (for irreducible V) and [37, Proposition B.2] (for general V), the Zariski closure $\overline{\pi(Z')}$ is a strict closed subset of $\mathbb{C}^s$. Now, at any point $(\boldsymbol{x}, \boldsymbol{\vartheta})$ of Y, the tangent spaces $T_{(\boldsymbol{x},\boldsymbol{\vartheta})}Y$ and $T_{(\boldsymbol{x},\boldsymbol{\vartheta})}\operatorname{reg}(V)$ coincide. As a result, Z is contained in Z', and the claim follows. □

We can now explain how $\boldsymbol{\vartheta}$ being a regular value of $\pi_{|Y}$ relates to 0 being a regular value of $\Phi_{\boldsymbol{\vartheta}}$. In what follows, we write our indeterminates as blocks of variables, with $\boldsymbol{X} = X_1, \ldots, X_n$ and $\boldsymbol{\Theta} = \Theta_1, \ldots, \Theta_s$. When not explicitly mentioned, Jacobian matrices involve derivatives with respect to both $\boldsymbol{X}$ and $\boldsymbol{\Theta}$.

Lemma 4.4. *For $(\boldsymbol{x}, \boldsymbol{\vartheta})$ in Y, $(\boldsymbol{x}, \boldsymbol{\vartheta})$ is in Z if and only if $\operatorname{jac}_{\boldsymbol{X}}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta}))$ has rank less than m.*

Proof. Let $\boldsymbol{M}$ denote the $(s+m) \times (s+n)$ Jacobian matrix of π and Φ with respect to $X_1, \ldots, X_n$ and $\Theta_1, \ldots, \Theta_s$, that is,

$$\boldsymbol{M} = \begin{bmatrix} \operatorname{jac}(\pi) \\ \operatorname{jac}(\Phi) \end{bmatrix} = \begin{bmatrix} \mathbf{0}_{s\times n} & \mathbf{I}_s \\ \operatorname{jac}(\Phi) & \end{bmatrix}.$$

Take $(\boldsymbol{x}, \boldsymbol{\vartheta})$ on Y. Then, the rank of $\boldsymbol{M}(\boldsymbol{x}, \boldsymbol{\vartheta})$ can be written as $\operatorname{rank}(\operatorname{jac}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta}))) + \operatorname{rank}([\mathbf{0}_{s\times n}\ \mathbf{I}_s] \mid \ker \operatorname{jac}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta})))$, where the latter is the rank of the restriction of $[\mathbf{0}_{s\times n}\ \mathbf{I}_s]$ to the nullspace of $\operatorname{jac}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta}))$.

Since $(\boldsymbol{x}, \boldsymbol{\vartheta})$ is in Y and since 0 is a regular value of Φ, $\operatorname{jac}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta}))$ has full rank m. On the other hand, the nullspace of that matrix is the tangent space $T_{\boldsymbol{x},\boldsymbol{\vartheta}}Y$, and $\operatorname{rank}([\mathbf{0}_{s\times n}\ \mathbf{I}_s] \mid \ker \operatorname{jac}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta})))$ is the dimension of $\pi(T_{\boldsymbol{x},\boldsymbol{\vartheta}}Y)$. In other words, the rank of $\boldsymbol{M}(\boldsymbol{x}, \boldsymbol{\vartheta})$ is equal to $m + \dim(\pi(T_{\boldsymbol{x},\boldsymbol{\vartheta}}Y))$.

This proves that for $(\boldsymbol{x}, \boldsymbol{\vartheta})$ in Y, $(\boldsymbol{x}, \boldsymbol{\vartheta})$ is in Z if and only if the matrix $\boldsymbol{M}$ has rank less than $s+m$ at $(\boldsymbol{x}, \boldsymbol{\vartheta})$. Now, notice that

$$\boldsymbol{M}(\boldsymbol{x}, \boldsymbol{\vartheta}) = \begin{bmatrix} \mathbf{0}_{s\times n} & \mathbf{I}_s \\ \operatorname{jac}_{\boldsymbol{X}}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta})) & \operatorname{jac}_{\boldsymbol{\Theta}}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta})) \end{bmatrix}.$$

This shows that the rank of $\boldsymbol{M}(\boldsymbol{x}, \boldsymbol{\vartheta})$ equals $s + \operatorname{rank}(\operatorname{jac}_{\boldsymbol{X}}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta})))$, and the lemma follows. □

As a result, suppose we take $\boldsymbol{\vartheta}$ in $\mathbb{C}^s - \pi(Z)$. Then for all $\boldsymbol{x}$ in $\Phi_{\boldsymbol{\vartheta}}^{-1}(0) \cap \mathscr{O}$, $(\boldsymbol{x}, \boldsymbol{\vartheta})$ is in Y, so it is not in Z; the previous lemma then implies that the Jacobian matrix of $\Phi_{\boldsymbol{\vartheta}}$, which is $\operatorname{jac}_{\boldsymbol{X}}(\Phi, (\boldsymbol{X}, \boldsymbol{\vartheta}))$, has full rank m at $\boldsymbol{x}$. In other words, 0 is a regular value of $\Phi_{\boldsymbol{\vartheta}}$ in the open set $\mathscr{O}$. To prove Proposition 4.1, it is thus enough to establish the existence of a non-zero polynomial of degree at most d^{m+n} that vanishes on $\overline{\pi(Z)}$. We already established that $\overline{\pi(Z)}$ is a strict subset of $\mathbb{C}^s$, so the only missing ingredient is to prove that it has degree at most d^{m+n}.

We start by bounding above the degree of $\overline{Z}$. The previous lemma shows the equality

$$Z = \{(\boldsymbol{x}, \boldsymbol{\vartheta}) \in \Omega \times \mathbb{C}^s \mid \Phi(\boldsymbol{x}, \boldsymbol{\vartheta}) = 0 \text{ and } \operatorname{rank}(\operatorname{jac}_{\boldsymbol{X}}(\Phi, (\boldsymbol{x}, \boldsymbol{\vartheta}))) < m\}.$$

Since all polynomials in Φ have degree at most d, and all entries of $\operatorname{jac}_{\boldsymbol{X}}(\Phi)$ at most $d-1$, we can apply Corollary 3.2, so as to deduce that $\deg(\overline{Z}) \leq d^{m+n}$. This implies that $\overline{\pi(\overline{Z})}$ has degree at most d^{m+n}, and the equality $\overline{\pi(\overline{Z})} = \overline{\pi(Z)}$ allows us to conclude the proof.

5 Overview of the main algorithm

Let $F = (f_1, \ldots, f_p)$ be a sequence of polynomials in $\mathbb{C}[X_1, \ldots, X_n]$. Suppose that the ideal $\langle F \rangle \subset \mathbb{C}[X_1, \ldots, X_n]$ is radical and that $V(F)$ is smooth and equidimensional of dimension $\delta = n - p$.

In this section, we give a high-level description of an algorithm from [35] that computes at least one point in each connected component of $V(F) \cap \mathbb{R}^n$. Correctness of this algorithm was established in [35] provided we are in generic coordinates: the algorithm solves a family of systems of equations that describe points on the *polar varieties* of $V(F)$, and being in generic coordinates ensures several desirable properties for these polar varieties.

After a brief review of the basic properties of polar varieties, we sketch the main algorithm and highlight what properties are needed for its correctness (the next sections will give quantitative statements regarding the genericity of these properties). In that, we mainly follow [35], but we also introduce requirements related to Lagrange systems, as introduced in Section 3, as they will be of help in further sections.

5.1 Polar varieties

Let F be as in the preamble and let $V = V(F)$. Recall that, for $i \in \{1, \ldots, n\}$, we denote by π_i the projection

$$\begin{aligned} \mathbb{C}^n &\to \mathbb{C}^i \\ (x_1, \ldots, x_n) &\mapsto (x_1, \ldots, x_i). \end{aligned}$$

For $i \leq \delta$, the i-th *polar variety* $W(i, F)$ is the set of critical points of the restriction of π_i to V, that is,

$$W(i, F) := \{\boldsymbol{x} \in V \mid \dim \pi_i(T_{\boldsymbol{x}} V) < i\}.$$

We naturally extend this definition to $i = \delta + 1$, by setting $W(\delta + 1, F) = V$.

For $1 \leq i \leq \delta + 1$, let $\operatorname{jac}(F)$, resp. $\operatorname{jac}(F, i)$, denote the Jacobian matrix of $F = (f_1, \ldots, f_p)$ with respect to $(X_1, \ldots, X_n)$, resp. to $(X_{i+1}, \ldots, X_n)$:

$$\operatorname{jac}(F) = \begin{bmatrix} \frac{\partial f_1}{\partial X_1} & \cdots & \frac{\partial f_1}{\partial X_n} \\ \vdots & & \vdots \\ \frac{\partial f_p}{\partial X_1} & \cdots & \frac{\partial f_p}{\partial X_n} \end{bmatrix}, \quad \operatorname{jac}(F, i) = \begin{bmatrix} \frac{\partial f_1}{\partial X_{i+1}} & \cdots & \frac{\partial f_1}{\partial X_n} \\ \vdots & & \vdots \\ \frac{\partial f_p}{\partial X_{i+1}} & \cdots & \frac{\partial f_p}{\partial X_n} \end{bmatrix}.$$

Since F generates a radical ideal, for any $\boldsymbol{x}$ in V, the tangent space $T_{\boldsymbol{x}}(V)$ is the kernel of $\mathrm{jac}(F, \boldsymbol{x})$; the assumption that V be δ-equidimensional and smooth implies that this kernel has dimension $\delta = n - p$ at all such $\boldsymbol{x}$. It follows that we can rephrase the definition of $W(i, F)$ as

$$W(i, F) = \{\boldsymbol{x} \in \mathbb{C}^n \mid f_1(\boldsymbol{x}) = \cdots = f_p(\boldsymbol{x}) = 0 \text{ and } \mathrm{rank}(\mathrm{jac}(F, i, \boldsymbol{x})) < p\}.$$

Let $P_i = \binom{n-i}{p}$ be the number of p-minors in $\mathrm{jac}(F, i)$, and let $M_{i,1}, \ldots, M_{i,P_i}$ be these minors (for $i = \delta + 1$, $P_{\delta+1} = 0$ since $\mathrm{jac}(F, \delta + 1)$ has size $p \times (p - 1)$). Then, as in Section 3, we deduce that $W(i, F)$ is defined by the polynomials

$$\mathfrak{J}(i, F) = \big(f_1, \ldots, f_p, M_{i,1}, \ldots, M_{i,P_i}\big). \tag{2}$$

The downside to defining polar varieties using minors of the truncated Jacobian matrix is that these equations are in general not complete intersection, due to the relations between minors of a matrix (the hypersurface case is an exception, since in this case only partial derivatives are used to define polar varieties). For both the polynomial system algorithm we will use below, and an application we will make of an effective Nullstellensatz, it will be necessary to have equations without such relations. To make this possible, we use an alternative modeling of polar varieties that uses Lagrange variables, as in Section 3. We may thus consider the zero-set of the polynomials

$$\big(F,\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F, i)\big) \in \mathbb{C}[X_1, \ldots, X_n, L_1, \ldots, L_p]^{p+n-i},$$

but as before, we will want to discard from the zero-set of these equations in $\mathbb{C}^{n+p}$ those components where all L_i's vanish identically. We pointed out that the saturation needed to remove such components is unlikely to yield convenient sets of generators, so we will again introduce a single additional equation, of the form $u_1L_1 + \cdots + u_pL_p - 1$, for a certain $\boldsymbol{u} = (u_1, \ldots, u_p)$ in $\mathbb{C}^p$. Thus, for such a vector $\boldsymbol{u}$, we define the following polynomials:

$$\mathfrak{L}(i, F, \boldsymbol{u}) = \big(F,\ [L_1\ \cdots\ L_p]\cdot\mathrm{jac}(F, i),\ u_1L_1+\cdots+u_pL_p-1\big) \in \mathbb{C}[X_1, \ldots, X_n, L_1, \ldots, L_p]^{p+n-i+1}. \tag{3}$$

Introducing the last equation discards all solutions with $L_1 = \cdots = L_p = 0$, but other components of interest may be removed as well. However, Proposition 3.1 shows that for a *generic* vector $\boldsymbol{u}$, the Zariski closure of the projection of the zero-set of these equations on the $X_1, \ldots, X_n$-space is indeed $W(i, F)$. In the algorithm, we will use random u_i's; the former proposition will allow us to quantify bad choices.

5.2 The algorithm

All notation being as before, we can now give the outline of Safey El Din and Schost's algorithm for computing at least one point in each connected component of $V(F) \cap \mathbb{R}^n$. To ensure its correctness, we will need certain genericity assumptions, which will be discussed in detail in the next sections.

After applying a randomly chosen change of variables $\boldsymbol{A}$, we further choose random $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_\delta)$ in $\mathbb{C}^\delta$, with $\delta = n - p$. Then, for $i = 1, \ldots, \delta + 1$, we compute (in the new coordinates) the points $\boldsymbol{x} = (x_1, \ldots, x_n)$ satisfying

$$x_1 = \sigma_1, \ldots, x_{i-1} = \sigma_{i-1},\ f_1(\boldsymbol{x}) = \cdots = f_p(\boldsymbol{x}) = 0,\ \operatorname{rank}(\operatorname{jac}(F, i, \boldsymbol{x})) < p. \tag{4}$$

In geometric terms, this means that we compute the intersection of $W(i, F)$ with the fiber $\pi_i^{-1}(\sigma_1, \ldots, \sigma_{i-1})$. Then, we return the union of all these sets.

Departing from [35], and following the discussion in the previous subsection, we will avoid solving the system generated by $F = (f_1, \ldots, f_p)$ and the p-minors of $\operatorname{jac}(F, i)$: to control costs, it will be beneficial to use the Lagrange system of (3) instead. Hence, some of our genericity assumptions will concern these equations. For $i = 1, \ldots, \delta + 1$, we define the following properties:

$\boldsymbol{H}_i(1)$: $W(i, F)$ is either empty or $(i-1)$-equidimensional;

$\boldsymbol{H}_i(2)$: 0 is a regular value of the $n+p-i$ polynomials $F,\ [L_1\ \cdots\ L_p] \cdot \operatorname{jac}(F, i)$ in the open set defined by $(L_1, \ldots, L_p) \neq (0, \ldots, 0)$;

$\boldsymbol{H}_i(3)$: assuming $\boldsymbol{H}_i(1)$ holds, $W(i, F)$ is either empty or in Noether position for π_{i-1}.

As we will see, these properties hold after applying a generic change of variables. Properties $\boldsymbol{H}_i(1)$ and $\boldsymbol{H}_i(3)$ ensure that Eq. (4) defines a finite set (as a consequence of the definition of Noether position), and guarantee that the output of the algorithm contains at least one point in each connected component of $V(F) \cap \mathbb{R}^n$ (this is proved in [35, Theorem 2]). The second one will be used to establish that assumption $\boldsymbol{H}_i'$ defined below holds generically.

Indeed, assuming (possibly after applying a change of variables) that F satisfies $\boldsymbol{H}_i$, we define our second genericity property:

$\boldsymbol{H}_i'$: $\boldsymbol{\sigma}$ is such that 0 is a regular value of the $n + p - 1$ polynomials

$$X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1},\ F,\ [L_1\ \cdots\ L_p] \cdot \operatorname{jac}(F, i),$$

in the open set defined by $(L_1, \ldots, L_p) \neq (0, \ldots, 0)$.

Again, we will see that this property holds for a generic choice of $\boldsymbol{\sigma}$ and that as a consequence, 0 is a regular value of the $n + p$ polynomials

$$X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1},\ F,\ [L_1\ \cdots\ L_p] \cdot \operatorname{jac}(F, i),\ u_1 L_1 + \cdots + u_p L_p - 1. \tag{5}$$

In particular, these equations admit finitely many solutions.

Suppose that for some i in $\{1, \ldots, \delta+1\}$, F satisfies $\boldsymbol{H}_i$ and $\boldsymbol{\sigma}$ satisfies $\boldsymbol{H}_i'$; then, we know that both systems (4) and (5) have finitely many solutions. In order to find the solutions of (4), we will compute those of (5) and project them on the $X_1, \ldots, X_n$-space; we choose to solve equations (5), since for this input, we can use the algorithm in [38], for which a complete bit complexity analysis is available. To guarantee success of this approach, we will rely on our last genericity property:

$\boldsymbol{H}_i''$: $\boldsymbol{u}$ is such that the projections of the solutions of (5) on the $X_1, \ldots, X_n$-space are the solutions of (1).

Applying Proposition 3.1 to the polynomials in (5) shows that this property holds for a generic choice of $\boldsymbol{u}$ (notice that since (5) has finitely many solutions, taking the Zariski closure, as done in the proposition, is not necessary in this case). If this is the case, the previous discussion shows that solving the systems (5), for $i = 1, \ldots, \delta + 1$, and projecting their solutions on the $X_1, \ldots, X_n$-space, solves our problem.

The next three sections prove the claims made above on the genericity of these properties: $\boldsymbol{H}_i(1)$ and $\boldsymbol{H}_i(2)$ in Section 6, as a first application of weak transversality; $\boldsymbol{H}_i(3)$ in Section 7, as an application of an effective Nullstellensatz; and $\boldsymbol{H}_i'$ in Section 8, as another first of weak transversality (essentially, Sard's lemma). In all cases, we gave quantitative form of these genericity statements. As we pointed out above, Proposition 3.1 is enough to prove that $\boldsymbol{H}_i''$ holds for generic $\boldsymbol{u}$, and already gives a quantitative statement.

6 Genericity of $\mathbf{H}_i(1)$ and $\mathbf{H}_i(2)$

Notation in this section is as before: we let $F = (f_1, \ldots, f_p) \in \mathbb{Z}[X_1, \ldots, X_n]^p$ be a sequence of polynomials defining a radical ideal, and where the degree of each polynomial is at most d; we also assume that the zero-set $V(F) \subset \mathbb{C}^n$ is smooth and δ-equidimensional, with $\delta = n - p$.

Consider an $n \times n$ matrix $\mathfrak{A}$ with indeterminates with entries $(\mathfrak{A}_{j,k})_{1 \le j,k \le n}$. In this section, we prove the following proposition.

Proposition 6.1. *For $i = 1, \ldots, \delta+1$, there exists a non-zero polynomial $\Delta_{i,1}$ in $\mathbb{C}[(\mathfrak{A}_{j,k})_{1 \le j,k \le n}]$ of degree at most $n(d^{5n} + 1)$ and with the following property. For $\boldsymbol{A}$ in $\mathbb{C}^{n \times n}$, if $\boldsymbol{A}$ does not cancel $\Delta_{i,1}$, then $F^{\boldsymbol{A}}$ satisfies $\mathbf{H}_i(1)$ and $\mathbf{H}_i(2)$.*

The rest of this section is devoted to the proof of the proposition; it is based on a construction introduced by Giusti, Heintz *et al.* (see for instance [3]). In all that follows, i is fixed in $1, \ldots, \delta + 1$; we then let $\mathfrak{A}_{\le i}$ denote the in indeterminates $(\mathfrak{A}_{j,k})_{1 \le j \le i, 1 \le k \le n}$. Writing $\boldsymbol{X} = X_1, \ldots, X_n$, we let $\boldsymbol{K}_i(\boldsymbol{X}, \mathfrak{A}_{\le i})$ denote the $(p + i) \times n$ matrix

$$\boldsymbol{K}_i(\boldsymbol{X}, \mathfrak{A}_{\le i}) = \begin{bmatrix} & \mathrm{jac}(F) & \\ \mathfrak{A}_{1,1} & \cdots & \mathfrak{A}_{1,n} \\ \vdots & & \vdots \\ \mathfrak{A}_{i,1} & \cdots & \mathfrak{A}_{i,n} \end{bmatrix}.$$

Consider elements $\boldsymbol{a} \in \mathbb{C}^{in}$ as vectors of length i of the form $\boldsymbol{a} = (\boldsymbol{a}_1, \ldots, \boldsymbol{a}_i)$ with $\boldsymbol{a}_i \in \mathbb{C}^n$; we say that $\boldsymbol{a}$ has rank i when $\boldsymbol{a}$ is a sequence of linearly independent vectors. Then for such an $\boldsymbol{a}$, $\boldsymbol{K}_i(\boldsymbol{X}, \boldsymbol{a})$ is naturally defined with the indeterminates $\mathfrak{A}_{\le i}$ evaluated at $\boldsymbol{a}$.

Let $\Phi : \mathbb{C}^{n+p+i} \times \mathbb{C}^{in} \to \mathbb{C}^{n+p}$ be the polynomial mapping in indeterminates $\boldsymbol{X} = X_1, \ldots, X_n$, $\boldsymbol{L} = L_1, \ldots, L_p$, $\boldsymbol{T} = T_1, \ldots, T_p$ and $\mathfrak{A}_{\le i}$ defined as

$$\Phi = (F,\ [L_1\ \cdots\ L_p\ T_1\ \cdots\ T_i] \cdot \boldsymbol{K}_i),$$

and for $\boldsymbol{a}$ in $\mathbb{C}^{ni}$, let $\Phi_{\boldsymbol{a}} : \mathbb{C}^{n+p+i} \to \mathbb{C}^{n+p}$ be the induced mapping $\Phi_{\boldsymbol{a}} = \Phi(\boldsymbol{X}, \boldsymbol{L}, \boldsymbol{T}, \boldsymbol{a})$ in variables $\boldsymbol{X}$, $\boldsymbol{L}$ and $\boldsymbol{T}$.

Let further $\mathscr{A} \subset \mathbb{C}^{n+p+i}$ be the open set defined by the condition $(L_1, \ldots, L_p) \neq (0, \ldots, 0)$. In [3, Section 3.2], it is shown that, for any $(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\vartheta}, \boldsymbol{a})$ in $\mathscr{A} \times \mathbb{C}^{in}$, the Jacobian matrix $\mathrm{jac}(\Phi)$, taken with respect to all indeterminates $\boldsymbol{X}, \boldsymbol{L}, \boldsymbol{T}, \mathfrak{A}_{\leq i}$, has full rank $n+p$ at $(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\vartheta}, \boldsymbol{a})$. In particular, this is true for $(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\vartheta}, \boldsymbol{a})$ in $\Phi^{-1}(0)$, so that 0 is a regular value of Φ on $\mathscr{A} \times \mathbb{C}^{in}$. It therefore follows by Proposition 4.1 that there exists a non-zero polynomial $\Gamma_i \in \mathbb{C}[\mathfrak{A}_{1,1}, \ldots, \mathfrak{A}_{i,n}]$ of degree at most

$$d^{(n+p+i)+(n+p)} \leq d^{5n},$$

such that if $\boldsymbol{a} \in \mathbb{C}^{in}$ does not cancel Γ_i, then 0 is a regular value of $\Phi_{\boldsymbol{a}}$ on $\mathscr{A}$. That is, for $(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\vartheta}) \in \mathscr{A} \cap \Phi_{\boldsymbol{a}}^{-1}(0)$, the Jacobian matrix $\mathrm{jac}(\Phi_{\boldsymbol{a}})$ has full rank $n+p$ at $(\boldsymbol{x}, \boldsymbol{\lambda}, \boldsymbol{\vartheta})$.

Let $\mathfrak{B} = \mathfrak{A}^{-1}$ in $\mathbb{C}((\mathfrak{A}_{j,k})_{1 \leq j,k \leq n})^{n \times n}$ and let $\mathfrak{B}_1 = [\mathfrak{B}_{1,1}, \ldots, \mathfrak{B}_{1,n}], \ldots, \mathfrak{B}_n = [\mathfrak{B}_{n,1}, \ldots, \mathfrak{B}_{n,n}]$ denote the rows of $\mathfrak{B}$. Set

$$\Delta_{i,1} := \Gamma_i(\mathfrak{B}_1, \ldots, \mathfrak{B}_i) \cdot (\det(\mathfrak{A}))^{\deg(\Gamma_i)+1}.$$

By multiplying through by $(\det(\mathfrak{A}))^{\deg(\Gamma_i)+1}$, we cancel all denominators and make $\Delta_{i,1}$ a polynomial multiple of $\det(\mathfrak{A})$.

Lemma 6.2. *The degree of $\Delta_{i,1}$ is at most $n(d^{5n}+1)$.*

Proof. Assume that

$$\mathfrak{B}_{s,t} = \mathfrak{N}_{s,t}/\det(\mathfrak{A}) \quad \text{with} \quad \mathfrak{N}_{s,t}, \det(\mathfrak{A}) \text{ in } \mathbb{C}[(\mathfrak{A}_{j,k})_{1 \leq j,k \leq n}],$$

for $1 \leq s, t \leq n$. Then, by Cramer's formulas, we have $\deg(\mathfrak{N}_{s,t}), \deg(\det(\mathfrak{A})) \leq n$, and since we have cleared all denominators by multiplying through with $(\det(\mathfrak{A}))^{\deg(\Gamma_i)+1}$, and guaranteed the presence of an extra factor $\det(\mathfrak{A})$, we therefore obtain

$$\deg(\Delta_{i,1}) \leq n \deg(\Gamma_i) + n \leq n(d^{5n}+1). \qquad \square$$

We first prove that $\Delta_{i,1}$ allows us to control when $F^{\boldsymbol{A}}$ satisfies $\mathbf{H}_i(1)$. The main ingredients in the proof of the following lemma are taken from [37], with no modification; this reference itself follows previous work such as [3].

Lemma 6.3. *For $\boldsymbol{A}$ in $\mathbb{C}^{n \times n}$, if $\boldsymbol{A}$ does not cancel $\Delta_{i,1}$, then $\boldsymbol{A}$ is invertible and the polar variety $W(i, F^{\boldsymbol{A}})$ is either empty or $(i-1)$-equidimensional.*

Proof. Consider $\boldsymbol{A} \in \mathbb{C}^{n \times n}$ that does not cancels $\Delta_{i,1}$. Since $\det(\mathfrak{A})$ divides $\Delta_{i,1}$, $\boldsymbol{A}$ is invertible, and by construction the first i rows $\boldsymbol{b}$ of $\boldsymbol{A}^{-1}$ do not cancel Γ_i. We put

$$Y := \{\boldsymbol{x} \in V(F) \mid \operatorname{rank}(\boldsymbol{K}_i(\boldsymbol{x}, \boldsymbol{b})) < p+i\}.$$

Lemma B.5 from [37] shows that all irreducible components of Y have dimension at least $i-1$; this is essentially Eagon and Northcott's result on determinantal varieties [15], and does

not depend on our choice of $\boldsymbol{b}$. On the other hand, our assumption on $\boldsymbol{b}$ allows us to apply Lemma B.11 from [37], which shows that all irreducible components of Y have dimension at most $i-1$. Therefore, Y is either empty or $(i-1)$-equidimensional. To conclude the proof, we use the equality

$$Y^{\boldsymbol{A}} = W\left(i, F^{\boldsymbol{A}}\right),$$

established in the same reference immediately before Lemma B.10. □

We conclude this section with the second property, $\mathbf{H}_i(2)$.

Lemma 6.4. *For $\boldsymbol{A}$ in $\mathbb{C}^{n\times n}$, if $\boldsymbol{A}$ does not cancel $\Delta_{i,1}$, then 0 is a regular value of the $n+p-i$ polynomials $F^{\boldsymbol{A}}$, $[L_1 \ \cdots \ L_p]\cdot \mathrm{jac}(F^{\boldsymbol{A}}, i)$ in the open set defined by $(L_1, \ldots, L_p) \neq (0, \ldots, 0)$.*

Proof. Take $\boldsymbol{A}$ in $\mathbb{C}^{n\times n}$ so that $\Delta_{i,1}(\boldsymbol{A}) \neq 0$, and let $(\boldsymbol{x}, \boldsymbol{\ell}) \in \mathbb{C}^{n+p}$ be a zero of the $n+p-i$ polynomials $F^{\boldsymbol{A}}$ and $[L_1 \ \cdots \ L_p]\cdot \mathrm{jac}(F^{\boldsymbol{A}}, i)$, with $\boldsymbol{\ell}$ non-zero. We have to show that the Jacobian matrix of these polynomials has full rank $n+p-i$ at $(\boldsymbol{x}, \boldsymbol{\ell})$.

We define a vector $\boldsymbol{\vartheta} = [\vartheta_1 \ \cdots \ \vartheta_i] \in \mathbb{C}^i$ by writing $\boldsymbol{\ell}\cdot \mathrm{jac}(F^{\boldsymbol{A}}) = [-\vartheta_1 \ \cdots \ -\vartheta_i \ 0 \ \cdots \ 0]$ (the trailing zeros result from our assumption on $\boldsymbol{x}$ and $\boldsymbol{\ell}$). It follows that $(\boldsymbol{x}, \boldsymbol{\ell}, \boldsymbol{\vartheta})$ cancels the equations

$$F^{\boldsymbol{A}}, \quad [L_1 \ \cdots \ L_p \ T_1 \ \cdots \ T_i] \begin{bmatrix} \mathrm{jac}(F^{\boldsymbol{A}}) \\ \boldsymbol{I}_i \quad \boldsymbol{0}_{i\times(n-i)} \end{bmatrix}, \tag{6}$$

where the Jacobian matrix of F is taken with respect to the variables $\boldsymbol{X} = X_1, \ldots, X_n$. We then post-multiply the right-hand matrix by $\boldsymbol{A}^{-1}$, and use the fact that $\mathrm{jac}(F^{\boldsymbol{A}}) = \mathrm{jac}(F)^{\boldsymbol{A}}\,\boldsymbol{A}$. This shows that $(\boldsymbol{x}, \boldsymbol{\ell}, \boldsymbol{\vartheta})$ also cancels the polynomials

$$F^{\boldsymbol{A}}, \quad [L_1 \ \cdots \ L_p \ T_1 \ \cdots \ T_i] \begin{bmatrix} \mathrm{jac}(F)^{\boldsymbol{A}} \\ \boldsymbol{b} \end{bmatrix}, \tag{7}$$

where again $\boldsymbol{b}$ denotes the first i rows of $\boldsymbol{A}^{-1}$. Setting $\boldsymbol{x}' = \boldsymbol{A}^{-1}\boldsymbol{x}$, we deduce that the point $(\boldsymbol{x}', \boldsymbol{\ell}, \boldsymbol{\vartheta})$ cancels

$$F, \quad [L_1 \ \cdots \ L_p \ T_1 \ \cdots \ T_i] \begin{bmatrix} \mathrm{jac}(F) \\ \boldsymbol{b} \end{bmatrix}, \tag{8}$$

that is, the polynomials $\Phi_{\boldsymbol{b}}$ defined in the preamble. The assumption on $\boldsymbol{A}$ shows that 0 is a regular value of this mapping in the open set defined by $(L_1, \ldots, L_p) \neq (0, \ldots, 0)$. Since $\boldsymbol{\ell}$ is by definition non-zero, this implies that the Jacobian matrix of the polynomials in Eq. (8) has full rank $n+p$ at $(\boldsymbol{x}', \boldsymbol{\ell}, \boldsymbol{\vartheta})$. Back in the original coordinates, we deduce that the Jacobian matrix of the polynomials in Eq. (7) has full rank $n+p$ at $(\boldsymbol{x}, \boldsymbol{\ell}, \boldsymbol{\vartheta})$. Right multiplication by $\boldsymbol{A}^{-1}$ in (8) amounts to performing a linear combination of the equations; hence, the Jacobian matrix of the polynomials in Eq. (6) has full rank $n+p$ at $(\boldsymbol{x}, \boldsymbol{\ell}, \boldsymbol{\vartheta})$ as well.

The Jacobian matrix of these polynomials taken with respect to the variables $X_1, \ldots, X_n$, $L_1, \ldots, L_p$ and $T_1, \ldots, T_i$ is equal to

$$\begin{bmatrix} \mathrm{jac}(F^{\boldsymbol{A}}) & \boldsymbol{0}_{p\times p} & \boldsymbol{0}_{p\times i} \\ \mathrm{jac}_{\boldsymbol{X},\boldsymbol{L}}\left([\boldsymbol{L}, \boldsymbol{T}]\cdot \begin{bmatrix} \mathrm{jac}(F^{\boldsymbol{A}}) \\ \boldsymbol{I}_i \quad \boldsymbol{0}_{i\times n-i} \end{bmatrix}\right) & \boldsymbol{I}_i \\ & \boldsymbol{0}_{(n-i)\times i} \end{bmatrix} = \begin{bmatrix} \mathrm{jac}(F^{\boldsymbol{A}}) & \boldsymbol{0}_{p\times p} & \boldsymbol{0}_{p\times i} \\ & *\,*\,* & \boldsymbol{I}_i \\ \mathrm{jac}_{\boldsymbol{X},\boldsymbol{L}}\left(\boldsymbol{L}\cdot \mathrm{jac}(F^{\boldsymbol{A}}, i)\right) & & \boldsymbol{0}_{(n-i)\times i} \end{bmatrix}.$$

Therefore, after removing i rows and columns, we can see that the submatrix

$$\begin{bmatrix} \mathrm{jac}(F^{\boldsymbol{A}}) & \mathbf{0}_{p\times p} \\ \mathrm{jac}_{\boldsymbol{X},\boldsymbol{L}}\left(\boldsymbol{L}\cdot\mathrm{jac}(F^{\boldsymbol{A}},i)\right) & \end{bmatrix} \tag{9}$$

has full rank $n+p-i$ at $(\boldsymbol{x},\boldsymbol{\ell})$. □

7 Genericity of $\mathbf{H}_i(3)$

Notation being as before, we now discuss the last genericity property that depends on our choice of coordinates. We already showed that in generic coordinates, the polar variety $W(i,F)$ is either empty or $(i-1)$-equidimensional. It remains to do the same for $\boldsymbol{H}_i(3)$, that is, to prove that if it is not empty, $W(i,F)$ is generically in Noether position for π_{i-1}. We will prove the following, where $\Delta_{i,1}$ is from Proposition 6.1.

Proposition 7.1. *For $i=1,\dots,\delta+1$, there exists a non-zero polynomial $\Delta_{i,2}$ in $\mathbb{C}[(\mathfrak{A}_{k,m})_{1\le k,m\le n}]$ of degree at most $4n^2(2d)^{4n}$ such that if $\boldsymbol{A}$ does not cancel $\Delta_{i,1}\Delta_{i,2}$, then $F^{\boldsymbol{A}}$ satisfies $\boldsymbol{H}_i(1)$, $\boldsymbol{H}_i(2)$ and $\boldsymbol{H}_i(3)$.*

Some results in a similar vein appear in the literature. For instance, Lemma 5 in [26] and Proposition 4.5 in [27] are quantitative Noether position statements. However, our results do not follow from these previous references, as these previous works analyze the probability that for a *fixed* algebraic set V, $V^{\boldsymbol{A}}$ be in Noether position. This does not solve our question, since $W(i,F^{\boldsymbol{A}})$, which we are interested in, is in general different from $W(i,F)^{\boldsymbol{A}}$.

Instead, we will rely on the proof given in [35] that $W(i,F^{\boldsymbol{A}})$ is in Noether position for a generic $\boldsymbol{A}$. However, we will not directly analyze the constructions used in that reference, since they involve e.g. primary decomposition in $\mathbb{C}((\mathfrak{A}_{j,k})_{1\le j,k\le n})[X_1,\dots,X_n]$, and the resulting degree bounds would be way beyond our target. We will instead combine results from [35] with an effective form of the Nullstellensatz given in [13]; as a result, we have to use Lagrange systems to describe polar varieties, since systems of minors do not satisfy assumptions needed to apply this effective Nullstellensatz.

The rest of this section is devoted to the proof of this proposition. From now on, we fix i in $0,\dots,\delta+1$.

7.1 Preliminaries

Property $\boldsymbol{H}_i(2)$ states that 0 is a regular value of the $n+p-i$ polynomials $F,\ [L_1\ \cdots\ L_p]\cdot\mathrm{jac}(F,i)$ in the open set defined by $(L_1,\dots,L_p)\neq(0,\dots,0)$; we saw that it holds in generic coordinates. We start by establishing some consequences of this fact for the polynomials $\mathfrak{L}(i,F,\boldsymbol{u})$ of Eq. (3).

Lemma 7.2. *Suppose that 0 is a regular value of the $n+p-i$ polynomials $F,\ [L_1\ \cdots\ L_p]\cdot\mathrm{jac}(F,i)$ in the open set defined by $(L_1,\dots,L_p)\neq(0,\dots,0)$. Then, for any $\boldsymbol{u}=(u_1,\dots,u_p)$ in $\mathbb{C}^p$, the $n+p-i+1$ polynomials*

$$\mathfrak{L}(i,F,\boldsymbol{u}) = F,\ [L_1\ \cdots\ L_p]\cdot\mathrm{jac}(F,i),\ u_1L_1+\cdots+u_pL_p-1$$

define a radical ideal, either trivial or $(i-1)$-equidimensional.

Proof. Take $(\boldsymbol{x}, \boldsymbol{\ell})$ in $\mathbb{C}^{n+p}$ that cancels the $n+p-i+1$ polynomials in (3). We prove that the Jacobian matrix of these equations has full rank $n+p-i+1$ at $(\boldsymbol{x}, \boldsymbol{\ell})$; the conclusion then follows from the Jacobian criterion.

Since $\boldsymbol{\ell}$ cannot be zero, our assumption implies that the Jacobian of the polynomials F and $[L_1 \ \cdots \ L_p] \cdot \mathrm{jac}(F, i)$ has full rank $n+p-i$ at $(\boldsymbol{x}, \boldsymbol{\ell})$. The conclusion therefore holds if $\mathrm{grad}(u_1 L_1 + \cdots + u_p L_p - 1) = [\mathbf{0}_{1\times n} \ u_1 \ \cdots \ u_p]$ is not in the row space of this matrix at $(\boldsymbol{x}, \boldsymbol{\ell})$. The Jacobian matrix of F and $[L_1 \ \cdots \ L_p] \cdot \mathrm{jac}(F, i)$ is equal to

$$\begin{bmatrix} \mathrm{jac}(F) & \mathbf{0}_{p\times p} \\ * * * & \mathrm{jac}(F, i)^T \end{bmatrix}.$$

Suppose that $[\mathbf{0}_{1\times n} \ u_1 \ \cdots \ u_p]$ is in the row-space of this matrix. Considering the last p columns gives us an equality $[u_1 \ \cdots \ u_p] = \boldsymbol{\mu}\, \mathrm{jac}(F, i)^T$, for some $\boldsymbol{\mu}$ in $\mathbb{C}^{1\times(n-i)}$. Right-multiplying by $\boldsymbol{\ell}^T \in \mathbb{C}^{p\times 1}$, we obtain $1 = 0$, a contradiction. □

The result carries over to our original polynomials in generic coordinates. In what follows, just as we defined $F^{\boldsymbol{A}}$ for $\boldsymbol{A}$ in $\mathbb{C}^{n\times n}$, we define $F^{\mathfrak{A}} = (f_1^{\mathfrak{A}}, \ldots, f_p^{\mathfrak{A}})$ as

$$(f_1(\mathfrak{A}\boldsymbol{X}), \ldots, f_p(\mathfrak{A}\boldsymbol{X})) \in \mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_n]^p.$$

Corollary 7.3. *For any $\boldsymbol{u} = (u_1, \ldots, u_p)$ in $\mathbb{C}^p$, the $n+p-i+1$ polynomials*

$$\mathfrak{L}(i, F^{\mathfrak{A}}, \boldsymbol{u}) = F^{\mathfrak{A}}, \ [L_1 \ \cdots \ L_p] \cdot \mathrm{jac}(F^{\mathfrak{A}}, i), \ u_1 L_1 + \cdots + u_p L_p - 1 \tag{10}$$

define a radical ideal in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_n, L_1, \ldots, L_p]$.

Proof. Proposition 6.1 and the previous lemma show that for $\boldsymbol{A}$ in a Zariski-dense subset of $\mathbb{C}^{n\times n}$, $\mathfrak{L}(i, F^{\boldsymbol{A}}, \boldsymbol{u})$ is radical in $\mathbb{C}[X_1, \ldots, X_n, L_1, \ldots, L_p]$; as a result, this must also be the case for the ideal $\mathfrak{L}(i, F^{\mathfrak{A}}, \boldsymbol{u})$ in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_n, L_1, \ldots, L_p]$. □

7.2 Degree bounds for integral dependence relationships

The results in Section 6 imply that $F^{\mathfrak{A}}$ satisfies $\boldsymbol{H}_i(1)$, so that $W(i, F^{\mathfrak{A}})$ is either empty or equidimensional of dimension $i-1$. We now point out that $F^{\mathfrak{A}}$ also satisfies $\boldsymbol{H}_i(3)$. In what follows, as in Eq. (2), we let $\mathfrak{J}(i, F^{\mathfrak{A}})$ be the polynomials consisting of $F^{\mathfrak{A}}$ and all p-minors of $\mathrm{jac}(F^{\mathfrak{A}}, i)$ in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_n]$, and we let $\mathscr{K}$ be the ideal they generate in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_n]$. In particular, the defining ideal of $W(i, F^{\mathfrak{A}})$ is $\sqrt{\mathscr{K}}$.

Our first lemma simply recalls results from [35]. In the following two lemmas, memberships statements are all considered in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_n]$; however, in the course of the proof of Lemma 7.5, we will work with the same polynomials, but seen in other polynomial rings.

Lemma 7.4. *For $j = i, \ldots, n$, there exists Q_j in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_{i-1}, X_j]$, monic in X_j and with Q_j in $\sqrt{\mathscr{K}}$. Furthermore, for any prime component $\mathfrak{P}$ of the ideal $\sqrt{\mathscr{K}}$ in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_n]$, we have $\mathfrak{P} \cap \mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \ldots, X_{i-1}] = \{0\}$.*

Proof. If $W(i, F^{\mathfrak{A}})$ is empty, then $\sqrt{\mathscr{K}}$ is the trivial ideal, so we simply take $Q_j = 1$ for all j; the second statement is vacuously true.

Otherwise, let $(\mathfrak{P}_\ell)_{1\leq \ell\leq L}$ be the prime components of $\sqrt{\mathscr{K}}$. By assumption, $L \geq 1$ and all $\mathfrak{P}_\ell$ have dimension $i-1$. By [35, Proposition 1], for all ℓ,

$$\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_{i-1}] \to \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_n]/\mathfrak{P}_\ell$$

is injective and integral. In particular, this means that $\mathfrak{P}_\ell$ contains no non-trivial polynomial in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_{i-1}]$, as claimed. Also, it proves that polynomials $q_{\ell,j} \in \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_{i-1}, X_j]$ exist, all monic in X_j, with $q_{\ell,j} \in \mathfrak{P}_\ell$ for each j in $\{i, \ldots, n\}$. Thence,

$$Q_j := \prod_{1\leq \ell\leq L} q_{\ell,j}$$

is monic in X_j and satisfies $Q_j \in \sqrt{\mathscr{K}}$, for each $j \in \{i, \ldots, n\}$. □

The former lemma does not directly give us degree bounds on the polynomials Q_j. This is the objective of the next step, where we control degree with respect to all unknowns involved, $X_1, \ldots, X_n$ as well as $\mathfrak{A}_{1,1}, \ldots, \mathfrak{A}_{n,n}$. In this respect, if P is any polynomial in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_n]$, we will let $D \in \mathbb{C}[(\mathfrak{A}_{k,m})_{1\leq k,m\leq n}]$ be the minimal common denominator of all its coefficients (defined up to a non-zero constant in $\mathbb{C}$), and we will write $\overline{P} := DP$, so that $\overline{P}$ is in $\mathbb{C}[(\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_n]$.

Lemma 7.5. *For $j = i, \ldots, n$, there exists P_j in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_{i-1}, X_j]$, monic in X_j, with P_j in $\sqrt{\mathscr{K}}$ and $\deg(\overline{P_j}) \leq (2d)^{2n}$.*

Proof. Consider the following ideals: they all have for generators the polynomials $\mathfrak{J}(i, F^{\mathfrak{A}})$, that is, $F^{\mathfrak{A}}$ and the p-minors of $\mathrm{jac}(F^{\mathfrak{A}}, i)$, but they lie in different polynomial rings.

- $\mathscr{J}$ is the ideal generated by $\mathfrak{J}(i, F^{\mathfrak{A}})$ in the polynomial ring $\mathbb{C}[(\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_n]$ in $n^2 + n$ indeterminates;
- $\mathscr{K}$, which we already saw is generated by the polynomials $\mathfrak{J}(i, F^{\mathfrak{A}})$ in the polynomial ring $\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_n]$ in n indeterminates (this is the ideal we are mainly interested in);
- $\mathscr{M}$ is the ideal defined by the same polynomials, but this time in the polynomial ring $\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1})[X_i, \ldots, X_n]$ in $n - i + 1$ indeterminates.

Step 0: Excluding a trivial case. Suppose that $W(i, F^{\mathfrak{A}})$ is empty, or equivalently that $\mathscr{K}$ is the trivial ideal. In this case, we take $P_j = 1$ for all j, and we are done. Henceforth, we assume that we are not in this situation.

Step 1: Defining the minimal polynomial P_j. The previous lemma shows that every irreducible component of the zero-set $W(i, F^{\mathfrak{A}})$ of $\mathscr{K}$ has dimension $i-1$, and that its image

by the projection π_i is onto. As a result, the extended ideal $\mathscr{M}$ has dimension zero, and the ring extension

$$\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1}) \to \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1})[X_i, \ldots, X_n]/\sqrt{\mathscr{M}}$$

is a product of finite field extensions. For $j = i, \ldots, n$, let then P_j be the minimal of X_j in this extension. Then, P_j is in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1})[X_j]$ and is monic in X_j.

Step 2: P_j is polynomial in $X_1, \ldots, X_{i-1}$. For j as above, the polynomial Q_j also belongs to $\sqrt{\mathscr{M}}$, so that P_j divides Q_j in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1})[X_j]$. We can therefore write

$$Q_j = P_j R_j, \qquad P_j, R_j \in \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1})[X_j].$$

It then follows by Gauss's lemma that we can write

$$Q_j = p_j r_j, \qquad p_j, r_j \in \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_{i-1}, X_j],$$

such that $\mu_j \in \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1})$ exists with

$$P_j = \mu_j p_j, \qquad R_j = \mu_j^{-1} r_j.$$

Since Q_j is monic in X_j, p_j and r_j must also be monic in X_j, so μ_j must be the coefficient of the highest degree term of P_j in X_j. Since P_j is monic in X_j, $\mu_j = 1$ and hence

$$P_j = 1 \cdot p_j = p_j \in \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_{i-1}, X_j].$$

Step 3: P_j is in $\sqrt{\mathscr{K}}$. It follows that P_j belongs to $\sqrt{\mathscr{M}} \cap \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_{i-1}, X_j]$. Equivalently, there exists a non-negative exponent s such that P_j^s is in the intersection $\mathscr{M} \cap \mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n})[X_1, \ldots, X_{i-1}, X_j]$. Clearing denominators in the membership equality in $\mathscr{M}$, this means that there exists D non-zero in $\mathbb{C}[(\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1}]$ such that DP_j^s is in $\mathscr{K}$, and thus in $\sqrt{\mathscr{K}}$.

By the previous lemma, no prime component of $\sqrt{\mathscr{K}}$ contains any non-zero polynomial in $\mathbb{C}[(\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1}]$. As a consequence, P_j^s is in $\sqrt{\mathscr{K}}$, and thus so is P_j itself.

Step 4: Degree of $\overline{P_j}$. For the last step of the proof, the ideal $\mathscr{J}$ is used. Let indeed Z be its zero-set in $\mathbb{C}^{n^2+n}$. Since all polynomials in $F^{\mathfrak{A}}$, resp. $\mathrm{jac}(F^{\mathfrak{A}}, i)$, have respective degrees at most $2d$ in $(\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_n$, resp. $2d-1$, Corollary 3.2 shows that Z has degree at most $(2d)^{2n}$.

Let further Z' be obtained by removing from Z all those irreducible components whose projection on the space of coordinates $(\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1}$ is not dense, and let $\mathscr{J}'$ be its defining ideal. It is a routine verification that the extension of $\mathscr{J}'$ in the polynomial ring $\mathbb{C}((\mathfrak{A}_{k,m})_{1\leq k,m\leq n}, X_1, \ldots, X_{i-1})[X_i, \ldots, X_n]$ is the radical of $\mathscr{M}$. As a result, Theorem 2 in [12] implies that the total degree of $\overline{P_j}$ is bounded above by $\deg(Z)$; this finishes the proof. □

7.3 Applying the effective Nullstellensatz

The previous lemma could allow us to give a quantitative proof that $\boldsymbol{H}_i(3)$ holds generically, if we were able to bound the degree in $(\mathfrak{A}_{k,m})_{1\le k,m\le n}$ of the corresponding membership equality in $\sqrt{\mathscr{K}}$. However, we are not aware of a suitable effective Nullstellensatz. The best suited one, due to D'Andrea, Krick and Sombra [13], requires that the number of generators in the ideal we consider be no more than the ambient dimension; this is in general not the case for the polynomials $\mathfrak{J}(i, F^{\mathfrak{A}})$.

As a result, we will use Lagrange systems instead. For $\boldsymbol{u}$ in $\mathbb{C}^p$, recall the definition of the Lagrange system $\mathfrak{L}(i, F^{\mathfrak{A}}, \boldsymbol{u})$ given in Eq. (10); let further $\mathscr{L}_{\boldsymbol{u}}$ be the ideal these polynomials generate in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n})[X_1, \dots, X_n, L_1, \dots, L_p]$. Proposition 3.1 gives the inclusion $\sqrt{\mathscr{K}} \subset \sqrt{\mathscr{L}_{\boldsymbol{u}}}$, and Corollary 7.3 shows that $\mathscr{L}_{\boldsymbol{u}}$ is a radical ideal, so that we have $\sqrt{\mathscr{K}} \subset \mathscr{L}_{\boldsymbol{u}}$. As a consequence, the polynomials P_j, and thus $\overline{P_j}$ as well, are in $\mathscr{L}_{\boldsymbol{u}}$ for any $\boldsymbol{u}$ in $\mathbb{C}^p$.

Let then $\mathfrak{u}_1, \dots, \mathfrak{u}_p$ be new indeterminates, and consider the ideal $\mathscr{L}_{\mathfrak{u}}$ generated by the polynomials $\mathfrak{L}(i, F^{\mathfrak{A}}, \mathfrak{u})$ in the ring of polynomials in $X_1, \dots, X_n, L_1, \dots, L_p$ over the field of coefficients $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n}, \mathfrak{u}_1, \dots, \mathfrak{u}_p)$; the only difference with the previous setting is that the linear form involved in these equations is now $\mathfrak{u}_1 L_1 + \cdots + \mathfrak{u}_p L_p - 1$. The previous discussion implies that all polynomials $\overline{P_j}$ belong to $\mathscr{L}_{\mathfrak{u}}$; we are now going to apply an effective Nullstellensatz to these membership equalities.

Let T be a new variable, and let $G_1, \dots, G_{n+p-i+1}$ be the $n + p - i + 1$ polynomials in the Lagrange system $\mathfrak{L}(i, F^{\mathfrak{A}}, \mathfrak{u})$. For $j = i, \dots, n$ applying the Nullstellensatz in $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n}, \mathfrak{u}_1, \dots, \mathfrak{u}_p)[X_1, \dots, X_n, L_1, \dots, L_p, T]$, and clearing denominators, we obtain the existence of A_j in $\mathbb{C}[(\mathfrak{A}_{k,m})_{1\le k,m\le n}, \mathfrak{u}_1, \dots, \mathfrak{u}_p] - \{0\}$ and of polynomial coefficients $C_{j,1}, \dots, C_{j,n+p-i+1}, B_j$ in $\mathbb{C}[(\mathfrak{A}_{k,m})_{1\le k,m\le n}, \mathfrak{u}_1, \dots, \mathfrak{u}_p)[X_1, \dots, X_n, L_1, \dots, L_p, T]$, such that

$$A_j = \sum_{\ell=1}^{n+p-i+1} C_{j,\ell} G_\ell + B_j(1 - \overline{P_j} T). \tag{11}$$

Let us then see A_j as a polynomial in $\mathfrak{u}_1, \dots, \mathfrak{u}_p$ with non-zero coefficients in $\mathbb{C}[(\mathfrak{A}_{k,m})_{1\le k,m\le n}]$, and let α_j be one of these coefficients, arbitrarily chosen. We can then define

$$\Delta_{i,2} := \alpha_i \cdots \alpha_n \in \mathbb{C}[(\mathfrak{A}_{k,m})_{1\le k,m\le n}] - \{0\}.$$

With this definition of $\Delta_{i,2}$, we prove the following lemma. It almost completes the proof of Proposition 7.1, except for the degree bound.

Lemma 7.6. *If $\boldsymbol{A} \in \mathbb{C}^{n\times n}$ does not cancel $\Delta_{i,1}\Delta_{i,2}$, then $F^{\boldsymbol{A}}$ satisfies $\boldsymbol{H}_i(1)$, $\boldsymbol{H}_i(2)$ and $\boldsymbol{H}_i(3)$.*

Proof. Let us take such a matrix $\boldsymbol{A}$. The non-vanishing of $\Delta_{i,1}(\boldsymbol{A})$ already guarantees that $F^{\boldsymbol{A}}$ satisfies $\boldsymbol{H}_i(1)$ and $\boldsymbol{H}_i(2)$. It remains to establish that $\boldsymbol{H}_i(3)$ holds, that is, that if it is not empty, $W(i, F^{\boldsymbol{A}})$ is in Noether position for π_{i-1}. In what follows, we assume that $W(i, F^{\boldsymbol{A}})$ is not empty; by $\boldsymbol{H}_i(1)$, it is $(i-1)$-equidimensional.

Fix j in $i, \ldots, n$. Because $\alpha_j(\boldsymbol{A})$ is non-zero, the polynomial $a_j := A_j(\boldsymbol{A}, \mathfrak{u}_1, \ldots, \mathfrak{u}_p)$ is non-zero in $\mathbb{C}[\mathfrak{u}_1, \ldots, \mathfrak{u}_p]$. We choose $\boldsymbol{u} = (u_1, \ldots, u_p)$ in $\mathbb{C}^p$ such that $a_j(u_1, \ldots, u_p)$ does not vanish, and such that $\boldsymbol{u}$ lies in the open set $\mathscr{O}$ associated by Proposition 3.1 to the set $W(i, F^{\boldsymbol{A}})$.

Let $g_1, \ldots, g_{n+p-i+1}$ be the polynomials in $\mathfrak{L}(i, F^{\boldsymbol{A}}, \boldsymbol{u})$. Evaluating $(\mathfrak{A}_{k,m})_{1\le k,m\le n}$ at the entries of $\boldsymbol{A}$ and $\mathfrak{u}_1, \ldots, \mathfrak{u}_p$ at $u_1, \ldots, u_p$ in (11) gives a relation of the form

$$\tilde{a}_j = \sum_{\ell=1}^{n+p-i+1} c_{j,\ell} g_\ell + b_j(1 - p_j T),$$

with $\tilde{a}_j = a_j(u_1, \ldots, u_p) \in \mathbb{C} - \{0\}$, polynomials $c_{j,\ell}$ and b_j in $\mathbb{C}[X_1, \ldots, X_n, L_1, \ldots, L_p, T]$ and p_j in $\mathbb{C}[X_1, \ldots, X_{i-1}, X_j]$, monic in X_j.

The next step is routine. Replace T by $1/p_j$ in the previous equality; after clearing denominators, this gives a membership equality of the form

$$p_j^k \in \langle \mathfrak{L}(i, F^{\boldsymbol{A}}, \boldsymbol{u}) \rangle$$

for some integer $k \ge 1$ (we cannot have $k = 0$, since we assumed that $W(i, F^{\boldsymbol{A}})$ is not empty). Using our assumption on $\boldsymbol{u}$, Proposition 3.1 then shows that p_j^k is in the ideal generated by $\mathfrak{J}(i, F^{\boldsymbol{A}})$, that is, by $F^{\boldsymbol{A}}$ and the p-minors of $\mathrm{jac}(F^{\boldsymbol{A}}, i)$. In other words, p_j is in the defining ideal of the polar variety $W(i, F^{\boldsymbol{A}})$. Repeating this for all $j = i, \ldots, n$ proves that $W(i, F^{\boldsymbol{A}})$ is in Noether position for π_{i-1}. □

To estimate the degree of $\Delta_{i,2}$, what remains is to give an upper bound on the degrees of $\alpha_i, \ldots, \alpha_n$. This will come as an application of the effective Nullstellensatz given in [13] over the function field $\mathbb{C}((\mathfrak{A}_{k,m})_{1\le k,m\le n}, \mathfrak{u}_1, \ldots, \mathfrak{u}_p)$. For this, we first need to determine degree bounds, separately in the actual indeterminates $X_1, \ldots, X_n, L_1, \ldots, L_p, T$ and in the "constants" $(\mathfrak{A}_{k,m})_{1\le k,m\le n}, \mathfrak{u}_1, \ldots, \mathfrak{u}_p$, of the polynomials in the membership relationship; we denote the former by $\deg_{\boldsymbol{X},\boldsymbol{L},T}$ and the latter by $\deg_{\mathfrak{A},\mathfrak{u}}$. Then, we have

$$\deg_{\boldsymbol{X},\boldsymbol{L},T} \{G_1, \ldots, G_{n+p-i+1}, \mathfrak{u}_1 L_1 + \cdots + \mathfrak{u}_p L_p - 1\} \le d, \quad \deg_{\boldsymbol{X},\boldsymbol{L},T}(1 - T\overline{P_j}) \le (2d)^{2n} + 1,$$

and

$$\deg_{\mathfrak{A},\mathfrak{u}} \{G_1, \ldots, G_{n+p-i+1}, \mathfrak{u}_1 L_1 + \cdots + \mathfrak{u}_p L_p - 1\} \le d \text{ and } \deg_{\mathfrak{A},\mathfrak{u}}(1 - T\overline{P_j}) \le (2d)^{2n}.$$

For each $j \in \{i, \ldots, n - p + 1\}$, since the number of equations in the ideal we consider is less than or equal to the ambient dimension $n + p + 1$, it follows from [13, Theorem 0.5] that

$$\deg(A_j) \le (2n + 2)d^{2n+1}((2d)^{2n} + 1);$$

we will use the slightly less precise bound

$$\deg(A_j) \le 4n(2d)^{4n}.$$

In particular, the same bound holds for the degree of α_j, and this gives

$$\deg(\Delta_{i,2}) \le 4n^2(2d)^{4n}.$$

This concludes the proof of Proposition 7.1.

8 Genericity of $\mathbf{H}_i^{'}$ and consequences

We still consider a sequence of polynomials $F = (f_1, \ldots, f_p) \in \mathbb{C}[X_1, \ldots, X_n]^p$ as before; in particular, recall we write $\delta = n - p$ and that d is an upper bound on the degrees of $f_1, \ldots, f_p$. Besides, we now also assume that F satisfies all assumptions $\boldsymbol{H}_i$ (for instance, because we have already applied a generic change of coordinates), and we prove the following.

Proposition 8.1. *For $i = 1, \ldots, \delta + 1$, if F satisfies $\boldsymbol{H}_i$, there exists a non-zero polynomial $\Xi_i \in \mathbb{C}[S_1, \ldots, S_{i-1}]$ of degree at most d^{4n} such that if $\boldsymbol{\sigma} = (\sigma_1, \ldots, \sigma_{i-1}) \in \mathbb{C}^{i-1}$ does not cancel Ξ_i, then $\boldsymbol{\sigma}$ satisfies assumption $\boldsymbol{H}_i'$, that is, 0 is a regular value of the $n + p - 1$ polynomials*

$$X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1},\ F,\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F, i)$$

in the open set defined by $(L_1, \ldots, L_p) \neq (0, \ldots, 0)$.

Proof. Let $\Psi : \mathbb{C}^{n+p} \times \mathbb{C}^{i-1} \to \mathbb{C}^{n+p-1}$ be the mapping defined by the $n + p - 1$ polynomials

$$X_1 - S_1, \ldots, X_{i-1} - S_{i-1},\ F,\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F, i)$$

in indeterminates $X_1, \ldots, X_n, L_1, \ldots, L_p, S_1, \ldots, S_{i-1}$. We claim that 0 is a regular value of Ψ in the open set $\Omega \times \mathbb{C}^{i-1} \subset \mathbb{C}^{n+p} \times \mathbb{C}^{i-1}$, here $\Omega \subset \mathbb{C}^{n+p}$ is defined by $(L_1, \ldots, L_p) \neq (0, \ldots, 0)$.

Consider a zero $(\boldsymbol{x}, \boldsymbol{\ell}, \boldsymbol{\sigma})$ of Ψ, with $\boldsymbol{\ell}$ non-zero. Indexing columns by

$$X_1, \ldots, X_n, L_1, \ldots, L_p, S_1, \ldots, S_{i-1},$$

the Jacobian matrix of Ψ is equal to

$$\begin{bmatrix} \boldsymbol{I}_{i-1} & \boldsymbol{0}_{(i-1)\times(n+p-i+1)} & -\boldsymbol{I}_{i-1} \\ \mathrm{jac}_{\boldsymbol{X},\boldsymbol{L}}\left(F,\ \boldsymbol{L} \cdot \mathrm{jac}(F, i)\right) & \boldsymbol{0}_{(n+p-i)\times(i-1)} \end{bmatrix}.$$

Because $\boldsymbol{\ell}$ is non-zero, $\boldsymbol{H}_i(2)$ shows that the Jacobian matrix $\mathrm{jac}_{(\boldsymbol{X},\boldsymbol{L})}\left(F,\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F, i)\right)$ has full rank $n + p - i$ at $(\boldsymbol{x}, \boldsymbol{\ell})$. Hence, the entire matrix must have full rank $n + p - 1$ at $(\boldsymbol{x}, \boldsymbol{\ell}, \boldsymbol{\sigma})$, and 0 is a regular value of Ψ.

Since all polynomials defining Ψ have degree at most d, it follows by Proposition 4.1 that there exists a non-zero polynomial Ξ_i in $\mathbb{C}[S_1, \ldots, S_{i-1}]$ of degree at most $d^{(n+p)+(n+p-1)} \leq d^{4n}$, with the property that, if $\Xi_i(\boldsymbol{\sigma}) \neq 0$ then at any root $(\boldsymbol{x}, \boldsymbol{\ell})$ of the induced mapping $\psi_{\boldsymbol{\sigma}}$ given by

$$X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1},\ F,\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F, i), \tag{12}$$

if $\boldsymbol{\ell}$ non-zero, then the Jacobian matrix of these equations has full rank $n + p - 1$ at $(\boldsymbol{x}, \boldsymbol{\ell})$. The proposition is proved. □

We will use this property through the following corollary, which we already mentioned in Subsection 5.2.

Corollary 8.2. *For $i = 1, \ldots, \delta + 1$, if F satisfies $\boldsymbol{H}_i$ and $\boldsymbol{\sigma}$ satisfies $\boldsymbol{H}'_i$, then for any $\boldsymbol{u} = (u_1, \ldots, u_p)$ in $\mathbb{C}^p$, 0 is a regular value of the $n + p$ polynomials*

$$X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1},\ F,\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F, i),\ u_1 L_1 + \cdots + u_p L_p - 1.$$

Proof. The proof is similar to that of Lemma 7.2. Suppose that $\Xi_i(\boldsymbol{\sigma})$ is non-zero, let $\boldsymbol{u} = (u_1, \ldots, u_p)$ be arbitrary in $\mathbb{C}^p$ and take $(\boldsymbol{x}, \boldsymbol{\ell})$ in $\mathbb{C}^{n+p}$ that cancels the $n+p$ polynomials

$$X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1},\ F,\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F, i),\ u_1 L_1 + \cdots + u_p L_p - 1. \tag{13}$$

Since $\boldsymbol{\ell}$ is necessarily non-zero, the previous discussion implies that the Jacobian of the polynomials in Eq. (12) has full rank $n + p - 1$ at $(\boldsymbol{x}, \boldsymbol{\ell})$. This Jacobian matrix is equal to

$$\begin{bmatrix} \boldsymbol{I}_{i-1} & \boldsymbol{0}_{(i-1)\times(n-i+1)} & \boldsymbol{0}_{(i-1)\times p} \\ & \mathrm{jac}(F) & \boldsymbol{0}_{p\times p} \\ & * * * & \mathrm{jac}(F, i)^T \end{bmatrix}.$$

As in the proof of Lemma 7.2, if we suppose that $[\boldsymbol{0}_{1\times n}\ u_1\ \cdots\ u_p]$ is in the row-space of this matrix, considering the last p columns and multiplying by $\boldsymbol{\ell}^T \in \mathbb{C}^{p\times 1}$ leads us to a contradiction. This proves that the Jacobian matrix of the equations in Eq. (13) has full rank $n + p$ at $(\boldsymbol{x}, \boldsymbol{\ell})$, as claimed. □

9 Analysis of the main algorithm

We conclude this paper by revisiting the algorithm sketched in Subsection 5.2. The probability analysis is based on the quantitative genericity results we established in the previous sections, using the DeMillo-Lipton-Schwartz-Zippel lemma. *In order to simplify some big-O estimates, we assume that the bound d on the degrees of the input polynomials satisfies $d \geq 2$, since the case of linear polynomials is trivial.*

9.1 Description of the pseudocode

The algorithm is randomized and takes as input a parameter $\epsilon \in (0, 1)$; the choices made in the algorithm guarantee that the probability of success is at least $1 - \epsilon$.

Randomness occurs in part due to the various choices we make (change of variables $\boldsymbol{A}$, parameter $\boldsymbol{\sigma}$, parameter $\boldsymbol{u}$). Besides, at Step 4, we use a minor modification of [38, Algorithm 2] to solve the system

$$X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1},\ F^{\boldsymbol{A}},\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F^{\boldsymbol{A}}, i),\ u_1 L_1 + \cdots + u_p L_p - 1.$$

of $n+p$ equations in $n+p$ unknowns $X_1, \ldots, X_n, L_1, \ldots, L_p$. This subroutine is randomized as well; in order to guarantee a higher probability of success, we repeat the calculation k times, for a well-chosen parameter k, and keep the output with the largest cardinality (we discuss this in our probability analysis below). Upon success, we have obtained a zero-dimensional

parameterization $\mathscr{Q}_i = ((q_i, v_{i,1}, \ldots, v_{i,n+p}), \lambda_i)$ of the solutions Z_i of these equations, but we are only interested in the projection Z'_i of these points on the $X_1, \ldots, X_n$-space. Recall that $\mathscr{Q}_i$ is such that

- $\lambda_i(v_{i,1}, \ldots, v_{i,n+p}) = Tq'_i \bmod q_i$, with λ_i a $\mathbb{Q}$-linear form in $X_1, \ldots, X_n, L_1, \ldots, L_p$;
- we have the equality $Z_i = \left\{ \left(\frac{v_{i,1}(\tau)}{q'(\tau)}, \ldots, \frac{v_{i,n+p}(\tau)}{q'(\tau)} \right) \mid q(\tau) = 0 \right\}$.

The only constraint on λ_i is that it take pairwise distinct values on the points of Z_i. Now, since the equations defining Z_i are linear in $L_1, \ldots, L_p$, the projection $Z_i \to Z'_i$ is one-to-one; this means that we can take λ_i depending on $X_1, \ldots, X_n$ only. This constraint can be enforced at no extra cost in the algorithm of [38]; if this is the case, then $\mathscr{Q}'_i = ((q_i, v_{i,1}, \ldots, v_{i,n}), \lambda_i)$ is a zero-dimensional parameterization of Z'_i.

The algorithm of [38] also requires that the input system be given by a straight-line program. We build it (at Step 3) in the straightforward manner already suggested in the introduction: given $F = (f_1, \ldots, f_p)$ in $\mathbb{C}[X_1, \ldots, X_n]^p$, we can build a straight-line program that evaluates each f_i in $O(d^n)$ operations, by computing all monomials of degree up to d, multiplying them by the corresponding coefficients in f_i, and adding results. To obtain a straight-line program for $f_i^{\boldsymbol{A}}$, we add $O(n^2)$ steps corresponding to the application of the change of variables $\boldsymbol{A}$. The number of operations here is thus

$$O(nd^n + n^3) = O^{\sim}(d^n);$$

note that here, we use the assumption $d \geq 2$. From this, we can compute and evaluate the required partial derivatives in the Jacobian of $F^{\boldsymbol{A}}$ in

$$O(nd^n) = O^{\sim}(d^n)$$

operations [10]. Then, the matrix vector product with the vector of Lagrange multipliers adds a cost that is polynomial in n and which we can therefore neglect in the soft-O notation. Finally, we add the linear equations $X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1}$; this gives the straight-line program Γ_i, whose length is $O^{\sim}(d^n)$.

Remark 9.1. *We can actually obtain a sharper bound for the length of the straight-line program. By computing all monomials and summing, the length of Γ_i becomes $O^{\sim}(p\binom{n+d}{n})$. However, we use the looser bound of $O(d^n)$ to simplify the analysis and not obfuscate notation.*

As we already pointed out in Subsection 5.2, if $F^{\boldsymbol{A}}$ satisfies $\boldsymbol{H}_i$, $\boldsymbol{\sigma}$ satisfies $\boldsymbol{H}'_i$ and $\boldsymbol{u}$ satisfies $\boldsymbol{H}''_i$, and if $\mathscr{Q}_i$ is a zero-dimensional parametrization of the solutions of the equations (14) at Step 3 (for all $i \in \{1, \ldots, n-p+1\}$), Theorem 2 in [35] establishes that the output returned in Step 7 will contain one point in each connected component of $V \cap \mathbb{R}^n$. (The claim made in Subsection 5.2 relied on an assertion that has since been proved, in Corollary 8.2.)

Algorithm 1: Main Algorithm

Input: $F = (f_1, \ldots, f_p) \in \mathbb{Z}[X_1, \ldots, X_n]^p$ with $\deg(f_i) \leq d$ and $\mathrm{ht}(f_i) \leq b$, and $0 < \epsilon < 1$. Assume that $d \geq 2$.

Output: $n - p + 1$ zero-dimensional parameterizations, the union of whose zeros includes at least one point in each connected component of $V(F) \cap \mathbb{R}^n$, with probability at least $1 - \epsilon$.

1 Construct

$$S := \{1, 2, \ldots, \lceil 4\epsilon^{-1} 5n^3 (2d)^{5n} \rceil\},$$
$$T := \{1, 2, \ldots, \lceil 4\epsilon^{-1} n d^{4n} \rceil\},$$
$$R := \{1, 2, \ldots, \lceil 4\epsilon^{-1} n d^{2n} \rceil\},$$

and choose $\boldsymbol{A} \in S^{n^2}$, $\boldsymbol{\sigma} \in T^{n-1}$ and $\boldsymbol{u} \in R^p$ uniformly at random;

2 **for** $i \leftarrow 1$ **to** $n - p + 1$ **do**

3 Build a straight-line program Γ_i that computes the equations

$$X_1 - \sigma_1, \ldots, X_{i-1} - \sigma_{i-1},\ F^{\boldsymbol{A}},\ [L_1\ \cdots\ L_p] \cdot \mathrm{jac}(F^{\boldsymbol{A}}, i),\ u_1 L_1 + \cdots + u_p L_p - 1; \quad (14)$$

4 Run [38, Algorithm 2] $k \geq \log_2(4n/\epsilon)$ times with input Γ_i;

5 Let $\mathscr{Q}_i = ((q_i, v_{i,1}, \ldots, v_{i,n+p}), \lambda_i)$ be the highest cardinality zero-dimensional parameterization returned in Step 4;

6 Denote by $\mathscr{Q}_i' = ((q_i, v_{i,1}, \ldots, v_{i,n}), \lambda_i)$ the parameterization of the projection of $\mathscr{Q}_i$ onto the $X_1, \ldots, X_n$-space;

7 **return** $[\mathscr{Q}_1', \ldots, \mathscr{Q}_{n-p+1}']$.

9.2 Bit operation cost

The following lists the costs for each step of Algorithm 1, assuming that the input polynomials have degree d and integer coefficients of height at most b.

(1) We defined $S := \{1, 2, \ldots, \lceil 4\epsilon^{-1} 5n^3 (2d)^{5n} \rceil\}$ and therefore the height of any $a_{i,j} \in S$ is at most

$$\log(4/\epsilon) + \log(5n^3 (2d)^{5n}) \in O^{\sim}(\log(1/\epsilon) + n \log d).$$

Since $|R|, |T| \leq |S|$, we also have that the height of any $\sigma_k \in T$ and $u_\ell \in R$ admits the same upper bound.

(3) After computing the partial derivatives in the Jacobian matrix, the height grows by at most another factor of $\log d$. Thus, all polynomials in the system considered at Step 3 have height

$$O^{\sim}(b + d(\log(1/\epsilon) + n \log(d))) = O^{\sim}(b + d \log(1/\epsilon) + dn).$$

All integer coefficients appearing in the straight-line program Γ_i satisfy the same bound.

(4) As a result, after applying [38, Algorithm 2] k times for each index i, with $k = O(\log(n) +$

$\log(1/\epsilon)$), the total boolean cost of the algorithm is

$$O^{\sim}(d^{3n+2p+1}\log(1/\epsilon)(b+\log(1/\epsilon)))$$

where the polynomials in the output have degree at most d^{n+p}, and height at most

$$O^{\sim}(d^{n+p+1}(b+\log(1/\epsilon))).$$

This proves the runtime estimate, as well as our bounds on the height of the output.

Remark 9.2. *We thank Mohab Safey El Din for pointing out to us that when $d=2$, in the quadratic case, we obtain notably better complexity. The previously stated runtime is a slight simplification of the one found in [38]. Taking into account a more precise estimate given in that reference, we see that in the quadratic case a single call to that algorithm uses*

$$O^{\sim}\left(n^5\binom{n-1}{p-1}\binom{n}{p}(b+\log(1/\epsilon)+n)2^{2p}\right)$$

bit operations, and the polynomials in the output have degree $\binom{n-1}{p-1}2^p$ and height

$$O^{\sim}\left(n\binom{n}{p}(b+\log(1/\epsilon)\right).$$

When p is fixed, all these quantities are polynomial in n. Altogether, for our main algorithm when $d=2$, the algorithm uses

$$O^{\sim}\left(n^7\binom{n-1}{p-1}\binom{n}{p}\log(1/\epsilon)(b+\log(1/\epsilon)2^p\right)$$

bit operations, and the polynomials in the output have degree $\binom{n-1}{p-1}2^p$ and height

$$O^{\sim}\left(n^2\binom{n}{p}(b+\log(1/\epsilon)2^p\right).$$

Furthermore, when in addition $p=1$ the algorithm then uses

$$O^{\sim}\left(n^8\log(1/\epsilon)(b+\log(1/\epsilon)\right)$$

bit operations, and the polynomials in the output have degree 2 and height

$$O^{\sim}\left(n^3(b+\log(1/\epsilon)\right).$$

9.3 Probability of success

Let $\Delta_{i,1}$ and $\Delta_{i,2} \in \mathbb{C}[(\mathfrak{A}_{j,k})_{1\leq j,k\leq n}]$ be the polynomials from Propositions 6.1 and 7.1. Denote by $\Delta := \prod_{i=1}^{n-p+1} \Delta_{i,1}\Delta_{i,2}$, and note that

$$\deg(\Delta) = \sum_{i=1}^{n-p+1} \deg(\Delta_{i,1}) + \deg(\Delta_{i,2}) \leq 5n^3(2d)^{5n}. \tag{15}$$

If $\boldsymbol{A} \in \mathbb{C}^{n\times n}$ does not cancel Δ, then $\boldsymbol{A}$ is invertible and $F^{\boldsymbol{A}}$ satisfies $\boldsymbol{H}_i$ for all i in $\{1, \ldots, n-p+1\}$. Now, assuming that $\boldsymbol{A}$ is such a matrix, let $\Xi_i \in \mathbb{C}[S_1, \ldots, S_{i-1}]$ be the polynomials from Proposition 8.1 applied to $F^{\boldsymbol{A}}$. Denote by $\Xi := \prod_{i=1}^{n-p+1} \Xi_i$, and note that

$$\deg(\Xi) = \sum_{i=1}^{n-p+1} \deg(\Xi_i) \leq nd^{4n}. \tag{16}$$

If $\boldsymbol{\sigma} \in \mathbb{C}^{i-1}$ does not cancel Ξ, then it satisfies $\boldsymbol{H}_i'$ for all $i \in \{1, \ldots, n-p+1\}$. Assume that this is the case.

As argued in Subsection 5.2, the last condition on our parameters is that $\boldsymbol{u}$ satisfy $\boldsymbol{H}_i''$ for all i. For a given index i, Proposition 3.1 shows the existence of a non-zero polynomial Υ_i in $\mathbb{C}[U_1, \ldots, U_p]$ such that if $\Upsilon_i(u_1, \ldots, u_p)$ is non-zero, $\boldsymbol{H}_i''$ holds; in addition, that proposition and Corollary 3.2 give an upper bound of d^{n+p} for the degree of Υ_i. We denote by $\Upsilon := \prod_{i=1}^{n-p+1} \Upsilon_i$, and note that

$$\deg(\Upsilon) = \sum_{i=1}^{n-p+1} \deg(\Upsilon_i) \leq nd^{2n}. \tag{17}$$

If $\boldsymbol{u} \in \mathbb{C}^p$ does not cancel Υ, then $\boldsymbol{u}$ satisfies $\boldsymbol{H}_i''$ for all $i \in \{1, \ldots, n-p+1\}$.

Then, the algorithm is guaranteed to succeed, as long as our calls to Algorithm 2 in [38] succeed in solving the equations at Step 3. That reference establishes that by repeating the calculation k times, and keeping the output of highest degree among those k results, we succeed with probability at least $1-(1/2)^k$. When Algorithm 2 does not succeed, it either returns a proper subset of the solutions, or FAIL. Note that Algorithm 2 is shown to succeed in a single run with probability at least $1-11/32$, and we bound the probability of success with $1-1/2$ for simplicity. Now, recall that we choose $\boldsymbol{A}$ in S^{n^2}, $\boldsymbol{\sigma}$ in T^{n-1} and $\boldsymbol{u}$ in R^p uniformly at random, with

$$\begin{aligned} S &= \{1, 2, \ldots, \lceil 4\epsilon^{-1}5n^3(2d)^{5n}\rceil\}, \\ T &= \{1, 2, \ldots, \lceil 4\epsilon^{-1}nd^{4n}\rceil\}, \\ R &= \{1, 2, \ldots, \lceil 4\epsilon^{-1}nd^{2n}\rceil\}. \end{aligned}$$

Using the DeMillo-Lipton-Schwartz-Zippel lemma, we obtain

$$\mathbb{P}[\Delta(\boldsymbol{A}) = 0] \leq \frac{\deg\Delta}{|S|} = \epsilon/4.$$

If this is the case, then

$$\mathbb{P}[\Xi(\boldsymbol{\sigma}) = 0] \leq \frac{\deg \Xi}{|T|} = \epsilon/4,$$

and if this is the case, then

$$\mathbb{P}[\Upsilon(\boldsymbol{u}) = 0] \leq \frac{\deg \Upsilon}{|R|} = \epsilon/4.$$

When all this holds, for a given index i, Step 4 succeeds with probability at least $1 - 1/2^k$, so the probability that all indices i succeed is at least $(1 - 1/2^k)^n$; our choice of the parameter k at Step 4 ensures that this probability is at least $\epsilon/4$ as well. Therefore, the overall probability of success is at least

$$(1 - \epsilon/4)^4 \geq 1 - \epsilon.$$

This finishes the proof of Theorem 1.1.

References

[1] M. Alonso, E. Becker, M.-F. Roy, and T. Wörmann. Zeroes, multiplicities and idempotents for zerodimensional systems. In *Algorithms in algebraic geometry and applications. Proceedings of MEGA'94*, volume 142 of *Progress in Mathematics*, pages 1–15. Birkhaüser, 1996.

[2] B. Bank, M. Giusti, and J. Heintz. Point searching in real singular complete intersection varieties: Algorithms of intrinsic complexity. *Mathematics of Computation*, 83:873–897, 2014.

[3] B. Bank, M. Giusti, J. Heintz, M. Safey El Din, and É. Schost. On the geometry of polar varieties. *Communication and Computing*, 21(1):33–83, 2010.

[4] B. Bank, M. Giusti, J. Heintz, L. Lehmann, and L.-M. Pardo. Algorithms of intrinsic complexity for point searching in compact real singular hypersurfaces. *Foundations of Computational Mathematics*, 12:75–122, 2012.

[5] B. Bank, M. Giusti, J. Heintz, and G. Mbakop. Polar varieties and efficient real equation solving: The hypersurface case. *Journal of Complexity*, 13(1):5–27, 1997.

[6] B. Bank, M. Giusti, J. Heintz, and G.-M. Mbakop. Polar varieties and efficient real elimination. *Mathematische Zeitschrift*, 238(1):115–144, 2001.

[7] B. Bank, M. Giusti, J. Heintz, and L.-M. Pardo. Generalized polar varieties: geometry and algorithms. *Journal of Complexity*, 21(4):377–412, 2005.

[8] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and computation in mathematics*. Springer-Verlag, 2003.

[9] S. Basu, É. Schost, M.-F. Roy, and M. Safey El Din. A baby step-giant step roadmap algorithm for general algebraic sets,. *Foundations of Computational Mathematics*, 14:1117–1172, 2014.

[10] W. Baur and V. Strassen. The complexity of partial derivatives. *Theoret. Comput. Sci.*, 22(3):317–330, 1983.

[11] J. Bochnak, M. Coste, and M.-F. Roy. *Real algebraic geometry*. Ergebnisse der Mathematik und ihrer Grenzgebite. Springer-Verlag, 1998.

[12] X. Dahan and É. Schost. Sharp estimates for triangular sets. In *ISSAC*, pages 103–110. ACM, 2004.

[13] C. D'Andrea, T. Krick, and M. Sombra. Heights of varieties in muliprojective spaces and arithmetic nullstellensatz. *Annales scientifiques de l'École Normale Supérieure*, 46(4):549–627, Aug 2013.

[14] M. Demazure. *Bifurcations and catastrophes: geometry of solutions to nonlinear problems*. Springer, 2000.

[15] J. Eagon and D. Northcott. Ideals defined by matrices and a certain complex associated with them. *Proceedings of the Royal Society of London. Series A. Mathematical and Physical Sciences*, 269(1337):188–204, 1962.

[16] D. Eisenbud. *Commutative Algebra with a View Toward Algebraic Geometry*, volume 150 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1st. edition, 1995.

[17] J. Elliott, M. Giesbrecht, and É. Schost. On the bit complexity of finding points in connected components of a smooth real hypersurface. In *ISSAC*, pages 170–177. ACM, 2020.

[18] P. Gianni and T. Mora. Algebraic solution of systems of polynomial equations using Groebner bases. In *AAECC*, volume 356 of *LNCS*, pages 247–257. Springer, 1989.

[19] M. Giusti, K. Hägele, J. Heintz, J.-E. Morais, J.-L. Montaña, and L.-M. Pardo. Lower bounds for diophantine approximation. *J. of Pure and Applied Algebra*, 117/118:277–317, 1997.

[20] M. Giusti, J. Heintz, J.-E. Morais, J. Morgenstern, and L.-M. Pardo. Straight-line programs in geometric elimination theory. *Journal of Pure and Applied Algebra*, 124:101–146, 1998.

[21] M. Giusti, J. Heintz, J.-E. Morais, and L.-M. Pardo. When polynomial equation systems can be solved fast? In *AAECC-11*, volume 948 of *LNCS*, pages 205–231. Springer, 1995.

[22] D. Grigoriev and N. Vorobjov. Solving systems of polynomial inequalities in subexponential time. *J. Symbolic Comput.*, 5:37–64, 1988.

[23] J. Hauenstein, M. Safey El Din, É. Schost, and X. T. Vu. Solving determinantal systems using homotopy techniques, 2018.

[24] J. Heintz. Definability and fast quantifier elimination in algebraically closed fields. *Theoretical Computer Science*, 24(3):239–277, May 1983.

[25] J. Heintz and C.P. Schnorr. Testing polynomials which are easy to compute. *STOC '80: Proceedings of the twelfth annual ACM symposium on Theory of computing*, 1980.

[26] G. Jeronimo and J. Sabia. Effective equidimensional decomposition of affine varieties. *Journal of Pure and Applied Algebra*, 169:229–248, 2002.

[27] T. Krick, L.-M. Pardo, and M. Sombra. Sharp estimates for the arithmetic nullstellensatz. *Duke Mathematical Journal*, 109(3):521–598, 2001.

[28] L. Kronecker. Grundzüge einer arithmetischen Theorie der algebraischen Grössen. *Journal für die reine und angewandte Mathematik*, 92:1–122, 1882.

[29] F. Macaulay. *The Algebraic Theory of Modular Systems*. Cambridge University Press, 1916.

[30] D. Mumford. *Algebraic Geometry 1 : complex algebraic varieties*. Classics in Mathematics. Springer, 1976.

[31] J. Nie and K. Ranestad. Algebraic degree of polynomial optimization. *SIAM J. on Optimization*, 20(1):485–502, 2009.

[32] R. Piene. Polar classes of singular varieties. *Annales Scientifiques de l'École Normale Supérieure*, 11(2):247–276, 1978.

[33] F. Rouillier. Solving zero-dimensional systems through the Rational Univariate Representation. *Applicable Algebra in Engineering, Communication and Computing*, 9(5):433–461, 1999.

[34] M. Safey El Din and P.-J. Spaenlehauer. Critical point computations on smooth varieties: degree and complexity bounds. In *ISSAC'16*, pages 183–190. ACM, 2016.

[35] É. Schost and M. Safey El Din. Polar varieties and computation of one point in each connected component of a smooth real algebraic set. *ISSAC*, pages 224–231, Aug 2003.

[36] É. Schost and M. Safey El Din. A baby steps/giant steps probabilistic algorithm for computing roadmaps in smooth bounded real hypersurface. *Discrete and Computational Geometry*, 5:181–220, 2011.

[37] É. Schost and M. Safey El Din. A nearly optimal algorithm for deciding connectivity queries in smooth and bounded real algebraic sets. *Journal of the ACM*, 63(6):1–48, February 2017.

[38] É. Schost and M. Safey El Din. Bit complexity for multi-homogeneous system solving application to polynomial minimization. *Journal of Symbolic Computation*, 87:176–206, May 2018.

[39] I. Shafarevich. *Basic Algebraic Geometry 1*. Springer Verlag, 1977.

[40] P.-J. Spaenlehauer. On the complexity of computing critical points with Gröbner bases. *SIAM Journal on Optimization*, 24(3):1382–1401, 2014.

[41] B. Teissier. Quelques points de l'histoire des variétés polaires, de poncelet à nos jours. In *Sém. Annales Univ. Blaise Pascal*, volume 4, 1988.

Chapter 4

Some applications of Chinese remainder theorem codes with error-correction

Some Applications of Chinese Remainder Theorem Codes with Error-Correction

Jesse Elliott[*], Éric Schost[†]

Overview

Modular algorithms based on the Chinese Remainder Theorem (CRT) control intermediate expression growth by performing computations modulo small primes. Some primes, that we call "unlucky," produce incorrect results, or no result; their number can be bounded by finding a nonzero $U \in \mathbb{Z}$ with the property that all unlucky primes divide U.

With no error correction mechanism, CRT-based methods require that all primes are lucky, resulting in large bounds on the primes we need to guarantee success with high probability $1 - \varepsilon$. Error-correction techniques introduced by Pernet and Böhm *et al.* tolerate a small number of unlucky primes, allowing us to reduce the prime size.

Given height bounds on the output and the integer U above, we perform a quantitative analysis that provides explicit sufficient conditions on the number and size of the primes to ensure a given success probability. We illustrate the approach with two examples, lexicographic Gröbner bases in two variables and the solution of zero-dimensional systems of equations, for which the bounds we need are already in the literature, or can be derived directly from previous work.

Keywords— Chinese remainder theorem codes; complexity; randomized algorithms; polynomial system solving

[*]David R. Cheriton School of Computer Science, University of Waterloo, On, Canada
[†]David R. Cheriton School of Computer Science, University of Waterloo, On, Canada

1 Introduction

When analyzing algebraic algorithms over a ground field such as $\mathbb{Q}$, algebraic complexity, which counts the number of base field operations, fails in general to predict actual runtime: even if accurate size bounds are known for the output, this is usually not the case for intermediate results (the issue of having large intermediate results is often called *intermediate expression swell*).

Modular algorithms avoid this issue by performing computations modulo one or several small primes, the ideal situation in practice being when we can use primes that fit in a single machine word, typically 64 bits, so that we can dispense with multi-precision arithmetic. The objective is to compute the requested output (typically, a matrix, or a set of polynomials) modulo a certain integer M. When M is sufficiently large, rational reconstruction can then be applied coefficientwise, recovering an output with rational coefficients.

In all this paper, we will measure the "size" of an integer, or of a rational number, using its *height*.

Definition 1. The *height* $\mathrm{ht}(r)$ of a rational number $r = u/v \in \mathbb{Q} \setminus \{0\}$ is the maximum of $\ln(|u|)$ and $\ln(v)$, for coprime integers $u \in \mathbb{Z}$ and $v \in \mathbb{N}$. □

In many situations, p-adic techniques such as Newton / Hensel iterations are available: this is the case for polynomial factorization [23, Chapter 15], or computing the non-singular solutions of polynomial equations (see [22, 12, 20] and references therein). In these cases, we work modulo a single prime p, solve the problem modulo p and lift the solution to $\mathbb{Z}/M\mathbb{Z}$, for sufficiently large $M = p^k$. The obvious alternative is to compute modulo sufficiently many "small" primes $(p_i)_{1 \le i \le \eta}$ and use the Chinese remainder theorem to obtain a solution modulo $M = p_1 \cdots p_\eta$.

For most problems of interest, there exist primes p where the procedure modulo p is ill-defined or produces results that do not match the modulo p reduction of the rational output. We call these primes *unlucky*, and otherwise *lucky*. It is in many cases not known how to test efficiently if a prime is lucky (this is of course problem-dependent). However, we can often bound the number of unlucky primes by proving the existence of a nonzero $U \in \mathbb{Z}$ with the property that all unlucky primes divide U and by bounding the height of U.

Using p-adic lifting techniques, the initial prime p must be lucky. Given an upper bound on U, we can select p from an appropriate interval to guarantee a success probability of at least $1 - \varepsilon$ for a given tolerance ε (sometimes a second prime is used, see e.g. [19] for this kind of idea over $k[t]$ instead of $\mathbb{Z}$). However, for CRT algorithms we use several primes, and CRT requires that all primes are lucky. Ensuring this condition with high probability $1 - \varepsilon$ typically results in larger primes.

In this paper, we apply error correction techniques (loosely analogous to Reed-Solomon decoding for rational number reconstruction), which tolerate some unlucky primes. We build our results on recent independent works by Böhm, Decker, Fieker, and Pfister [5, 4], in the context of algebraic geometry algorithms, and by Pernet [17], who discusses applications to linear algebra. The decoding algorithms and the sufficient conditions for success in these

works are very similar, albeit distinct: both follow an iterative reduction procedure, formulated as a variant of Euclid's GCD algorithm in Pernet's work and as a variant of Gaussian lattice reduction by Böhm *et al.*

While these techniques are not new, in Section 2, we provide explicit sufficient conditions on the number and size of the primes to ensure a given success probability. What is needed in all cases are *(i)* an upper bound on the size of the coefficients in the output and *(ii)* an upper bound on the product of unlucky primes (it would be natural to express these size bounds in terms of height, but we will use logarithms in base two, as this follows more closely Pernet's presentation). We obtain explicit, simple, values for the prime size using both Markov's and Chebyshev's bounds.

A first possible application where for such results comes from [21], where Storjohann gives a modular algorithm computing Hermite forms for matrices with entries in $\mathbb{Q}[x]$, together with the bounds on the output size and on the unlucky primes that we need. We are not aware of an alternative method that would rely on Newton iteration for this problem. Based on Storjohann's result, St-Pierre and Schost provide similar bounds for the computation of lexicographic bivariate Gröbner bases in [25]. Our work again applies directly; an alternative approach based on Newton iteration is in [24, 25], but the lifting procedure is rather intricate.

As a last example, let us mention the solution of polynomial equations (in dimension zero), by means of a data structure called *zero-dimensional parameterization* [14, 16, 22, 11, 2, 9, 10, 12, 18]. When complex solutions have multiplicity one, a simple form of Newton iteration applies, going back to [22]. Without this assumption, however, lifting techniques still exist but become substantially more complex [15], so considering CRT-based approaches is of interest. In this context, we can use the arithmetic Bézout theorem to bound the output size, but describing the unlucky primes is less straightforward. The work in reference [8] quantifies primes p where the number of solutions of a zero-dimensional polynomial system changes modulo p; in Section 3, we show how this analysis can also allow us to bound primes of bad reduction for our target output.

Finally, in Section 4, we illustrate the bounds derived in Section 2 on non-trivial instance sizes for the last two problems we mentioned, lexicographic Gröbner bases in two variables, and zero-dimensional systems. We compare the expressions we obtained using Markov's and Chebyshev's bounds to the size of primes that would guarantee *all* primes to be lucky with high probability (for situations where we do not want to use error correction). As we will see, for many instances, word-size primes are sufficient to guarantee a high probability of success.

Remark 1. We focus our discussion on reconstructing a single rational number. However, in the contexts we consider, algorithms return several numbers (such as coefficients of a polynomial, or entries in a matrix). In many cases, and in particular in the three applications we described above, one proves at once the existence of an integer U (and one can bound its size), such that if p does not divide U, *all* coefficients we are interested in reduce well modulo p. Hence, we can choose a unique set of primes and analyze the success probability for all coefficients at once.

One can go further along this line: typically, the rational numbers we get as output are known to admit a small common denominator. In error-free contexts, *vector rational*

reconstruction [6] allows us to recover all outputs by means of lattice reduction techniques using less modular information (asymptotically, using a modulus M of up to half the height we would use in the scalar case). Accounting for errors naturally leads one to error-tolerant vector rational reconstruction [1]: given models on the distribution of errors, the authors analyze the probability of there being a unique solution to the reconstruction problem, even though the number of errors would be too large to guarantee it *a priori*. We did not consider this direction in this work. □

2 CRT for Rational Reconstruction with Errors

2.1 The setting

Let us first review Chinese remaindering for rational reconstruction in the presence of errors. Our objective is to reconstruct a rational number r, which admits the expression $r = f/g$, with $g > 0$ and f, g coprime. Here is our setup:

- We assume that we are given an upper bound H on the base-2 representation of r, that is, $\log_2(|f|),\ \log_2(g) \leq H$.
- To each prime $p \in \mathbb{N}$, we suppose we can associate a value $r(p) \in \{0, \ldots, p-1\} \cup \{\infty\}$, where ∞ is a new symbol that indicates no integer value is available. We say that p is *lucky* when $r(p) = r \bmod p$ (in particular, p cannot divide g for such primes), and *unlucky* otherwise.
- We assume that there are finitely many unlucky primes and we let U be their product. In addition to the output size bound H, we assume that we are given a bound C such that $\log_2(U) \leq C$.

Both H and C are problem-dependent.

In this section, we conduct a quantitative analysis of an error-correcting CRT algorithm due to Böhm *et al.* [5] and Pernet [17], which results in explicit sufficient conditions on the number of primes and their size to guarantee an arbitrary probability of success. To our knowledge, our derivation, while relatively straightforward, is new.

Take positive integers σ, η and set $\Sigma = \{\sigma, \ldots, 2\sigma\} \subset \mathbb{N}$. Suppose we are given η primes $p_1, \ldots, p_\eta$ in Σ, and define

$$N = \sum_{1 \leq i \leq \eta} \log_2(p_i) \quad \text{and} \quad L = \sum_{\substack{1 \leq i \leq \eta \\ p_i \text{ unlucky}}} \log_2(p_i).$$

Pernet proved in [17, Lemma 2.5.4] that if $L < (N - 2H + 1)/2$, given $[p_1, \ldots, p_\eta]$ and the vector $[r(p_1), \ldots, r(p_\eta)]$, one can reconstruct r. Böhm *et al.* proved similar results. Note that here we will not be concerned with the cost of the reconstruction algorithm, but it is easily seen to be polynomial, and can be made quasi-linear, in H.

We will enforce the following relationship between the number η of primes we choose and the interval $\Sigma = \{\sigma, \ldots, 2\sigma\}$ where we choose them: throughout, we will set

$$\eta = \left\lceil \frac{(2+\gamma)H}{\log_2(\sigma)} \right\rceil, \tag{1}$$

for a fixed parameter $\gamma > 0$. Since all primes we choose are in Σ, in an error-free context, we would be able to perform CRT for rational reconstruction as soon as $\eta \log_2(\sigma) > 2H$. In the presence of errors, the factor 2 is too tight, but a factor $2+\gamma$ is sufficient. Using a small value for γ means that the amount of modular information we compute is close to what it would be in error-free situations.

Remark 2. In the context of a modular algorithm, the most important component in the cost analysis is usually the total time spent solving the problem modulo the primes p_i: CRT / rational reconstruction is essentially linear time, and finding primes is in practice not a bottleneck. In rough approximation, one can assume that execution mod p_i takes T operations modulo p_i, where T is independent of i. It follows that the total boolean cost is softly-linear in $T \sum_{1 \le i \le \eta} \log_2(p_i) \simeq T\eta \log(\sigma)$. In our construction, we have $\eta \log_2(\sigma) \le (2+\gamma)H + \log_2(\sigma)$. In other words, the boolean cost involves both the output size H, which is as expected, together with $\log_2(\sigma)$ (which will be negligible except for astronomical values of σ). □

2.2 Prime selection

Our first task is to be able to sample η distinct primes in the interval $\Sigma = \{\sigma, \ldots, 2\sigma\} \subset \mathbb{N}$. Our solution is simple: sample a first prime p_1 (by repeatedly picking a random element in Σ until we find a prime), then p_2 until we find a prime distinct from p_1, etc.

While this is not the main focus of this paper, we should mention the expected runtime of this procedure: there are $\Theta(\sigma/\log(\sigma))$ primes in Σ, so we expect to do $\Theta(\log(\sigma))$ attempts before finding a prime; each primality test can be done in polynomial time (in a boolean RAM model). Assuming that there are at least 2η primes in Σ, at any stage it is enough to pick an expected $O(1)$ primes before finding one that was not previously selected. We can store our primes in a self-balancing binary tree, with a boolean cost of $O(\log(\eta)\log(\sigma))$ for search and insert. Altogether, this results in an expected total of $O(\log(\eta)\log(\sigma)^c)$ bit operations, for some constant c.

Since there are at least $\sigma/2\log_2(\sigma)$ primes in Σ [23, Ex. 18.18], to satisfy the condition we gave above, it suffices that $2\eta \le \sigma/2\log_2(\sigma)$. Since our definition of η implies

$$\eta \le \frac{2(2+\gamma)H}{\log_2(\sigma)},$$

it is sufficient to take σ such that

$$\sigma \ge 8(2+\gamma)H.$$

2.3 Quantitative bounds

For $i = 1, \ldots, \eta$, let X_i be the indicator variable defined as

$$X_i = \begin{cases} 1 & \text{if } p_i \text{ is unlucky} \\ 0 & \text{otherwise.} \end{cases}$$

Note that for all i, $\mathbb{P}(p_i = p)$ is independent of the choice of the prime p in Σ. On the other hand, since we require the primes to be pairwise distinct, the X_i's are not pairwise independent. However, we can say the following about their covariance.

Lemma 1. For all $i < j$, the covariance $\mathrm{Cov}(X_i, X_j)$ is negative.

Proof. Let Σ_P be the primes in Σ, let M be their number, and let M_L, resp. M_U, be the number of lucky, resp. unlucky ones. In particular, $\mathbb{P}(X_i = 1) = \mathbb{P}(X_j = 1) = M_U/M$. We now have to estimate $\mathbb{P}(X_i = 1 \text{ and } X_j = 1)$.

There are $M_U(M_U - 1)$ choices of (p_i, p_j) that have $p_i \neq p_j$ and p_i, p_j both unlucky, and $M(M-1)$ choices of (p_i, p_j) that satisfy $p_i \neq p_j$, so the probability above is $M_U(M_U - 1)/M(M-1)$. It follows that the covariance of X_i, X_j is given by

$$\mathrm{Cov}(X_i, X_j) = \frac{M_U(M_U-1)}{M(M-1)} - \frac{M_U^2}{M^2} = \frac{M_U(M_U - M)}{M^2(M-1)} < 0.$$

□

Define further

$$X = \sum_{i=1}^{\eta} X_i,$$

that is, the number of unlucky primes we chose. For any choice of η distinct primes $p_1, \ldots, p_\eta$ in Σ, the quantities N and L above satisfy $N \geq \eta \log_2(\sigma)$ and $L \leq \log_2(2\sigma)X$. It follows that if we set

$$\Delta = (\eta \log_2(\sigma) - 2H + 1)/2,$$

if

$$X < \frac{\Delta}{\log_2(2\sigma)} = \frac{\eta \log_2(\sigma) - 2H + 1}{2 \log_2(2\sigma),}$$

the inequality $L < (N - 2H + 1)/2$ holds, and we successfully recover our rational number.

Recall that U is the product of all unlucky primes, and that C is such that $\log_2(U) \leq C$.

Lemma 2. Let r, H, C be as above. For a prime p chosen uniformly at random in $\Sigma = \{\sigma, \ldots, 2\sigma\}$, the probability that p be unlucky is at most $2C/\sigma$.

Proof. Let $\mathscr{D}$ be the set of primes in Σ that divide U, and notice that the product $\prod_{p \in \mathscr{D}} p$ also divides U, so that in particular $\prod_{p \in \mathscr{D}} p \leq U$. Each prime in Σ is at least equal to σ, so that

$\#\mathscr{D} \leq C/\log_2(\sigma)$. On the other hand, the number of primes in Σ is at least $\sigma/(2\log_2(\sigma))$, so that for a prime p chosen at random in Σ, the probability that it divides U is at most

$$\frac{(C/\log_2(\sigma))}{(\sigma/(2\log_2(\sigma)))} = \frac{2C}{\sigma}.$$

□

It follows that for all i, $\mathbb{E}[X_i] = \mathbb{P}(X_i = 1) \leq P := 2C/\sigma$, and thus

$$\mathbb{E}[X] \leq \frac{2\eta C}{\sigma}.$$

In the following proposition, we use Markov's inequality to give an estimate on σ to guarantee an arbitrary small probability of failure.

Proposition 1. Let r, H, C be as above. For $\varepsilon > 0$ and $\gamma > 0$, let σ be an integer such that

$$\sigma \geq \max\left(8(2+\gamma)H,\ 8\left(\frac{2+\gamma}{\gamma}\right)\frac{C}{\varepsilon}\right),$$

and take η as in (1).

Select pairwise distinct primes $p_1, \ldots, p_\eta$ independently and uniformly at random from the set $\Sigma = \{\sigma, \ldots, 2\sigma\} \subset \mathbb{N}$. Then, with probability at least $1 - \varepsilon$, given given $[p_1, \ldots, p_\eta]$ and the vector $[r(p_1), \ldots, r(p_\eta)]$, one can reconstruct r.

Proof. First, note that the first condition in the inequality for σ implies that there are indeed at least 2η primes in the interval $\Sigma = \{\sigma, \ldots, 2\sigma\}$.

Then, Markov's inequality states that for $a > 0$, $\mathbb{P}(X \geq a) \leq \mathbb{E}[X]/a$. As per our previous discussion, we take

$$a = \frac{\eta\log_2(\sigma) - 2H + 1}{2\log_2(2\sigma)}.$$

For our choice of η in (1), we get $\eta \geq (2+\gamma)H/\log_2(\sigma)$, so a is indeed positive. We deduce

$$\begin{aligned}\mathbb{P}(\text{fail}) &\leq \left(\frac{2\eta C}{\sigma}\right)\left(\frac{2\log_2(2\sigma)}{\eta\log_2(\sigma) - 2H + 1}\right) \\ &\leq \frac{8C}{\sigma}\frac{\eta\log_2(\sigma)}{\eta\log_2(\sigma) - 2H} = \frac{8C}{\sigma}\frac{1}{1 - \frac{2H}{\eta\log_2(\sigma)}}.\end{aligned}$$

Our inequality on η also implies that the right-most factor in the inequality above is at most $(2+\gamma)/\gamma$, so that

$$\mathbb{P}(\text{fail}) \leq \left(\frac{8C}{\sigma}\right)\left(\frac{2+\gamma}{\gamma}\right).$$

This can be made less than ε as soon as $\sigma \geq \left(\frac{8C}{\varepsilon}\right)\left(\frac{2+\gamma}{\gamma}\right)$. □

Next, we derive bounds on σ using Chebyshev's inequality.

Proposition 2. Let r, H, C be as above. For $\varepsilon > 0$ and $\gamma > 0$, let σ be an integer such that

$$\sigma \geq \max\left(8(2+\gamma)H,\ 16^{(\gamma+2)/\gamma},\ 8C\left(\frac{2+\gamma}{\gamma}\right),\ 2A\log_2(A)\right),$$

with $A := 128\frac{(2+\gamma)}{\gamma^2}\frac{C}{\varepsilon H}$, and take η as in (1). Then, the same conclusion as in Proposition 1 holds.

Proof. As above, our first assumption on σ implies that there are at least 2η primes in the interval $\{\sigma, \ldots, 2\sigma\}$.

Next, let P_0 denote the probability $\mathbb{P}(X_i = 1)$, so that $P_0 \leq P = 2C/\sigma$. Since we saw that $\mathrm{Cov}(X_i, X_j) < 0$ for all $i < j$, it follows that the variance of X is bounded above by the sum of the variances of the X_i's, that is, $\eta P_0(1 - P_0)$. This is at most $\eta P(1 - P)$, since our assumption $\sigma \geq 8C(2+\gamma)/\gamma$ implies $P \geq 1/2$.

We want the probability $\mathbb{P}\left(X \geq \Delta/\log_2(2\sigma)\right)$ to be at most ε. Note that using $\mathbb{E}[X] \leq 2\eta C/\sigma$, Chebyshev's inequality gives, for $k > 0$,

$$\begin{aligned}
&\mathbb{P}\left(X - \frac{2\eta C}{\sigma} \geq k\sqrt{\mathrm{Var}(X)}\right) \\
&\leq \mathbb{P}\left(X - \mathbb{E}[X] \geq k\sqrt{\mathrm{Var}(X)}\right) \\
&\leq \mathbb{P}\left(|X - \mathbb{E}[X]| \geq k\sqrt{\mathrm{Var}(X)}\right) \leq \frac{1}{k^2}.
\end{aligned}$$

Therefore put

$$k := \left(\frac{\Delta}{\log_2(2\sigma)} - \frac{2\eta C}{\sigma}\right) \cdot \frac{1}{\sqrt{\mathrm{Var}(X)}},$$

so that the left-hand term in the previous inequalities is precisely $\mathbb{P}\left(X \geq \Delta/\log_2(2\sigma)\right)$. Rewrite k as

$$\begin{aligned}
k &= \left(\frac{\eta\log_2(\sigma) - 2H + 1}{2\log_2(2\sigma)} - \frac{2\eta C}{\sigma}\right) \cdot \frac{1}{\sqrt{\mathrm{Var}(X)}} \\
&= \left(\frac{(\eta\log_2(\sigma) - 2H + 1)\sigma - 4\eta C\log_2(2\sigma)}{2\sigma\log_2(2\sigma)}\right) \cdot \frac{1}{\sqrt{\mathrm{Var}(X)}}.
\end{aligned}$$

Set

$$\Gamma := \frac{(\eta\log_2(\sigma) - 2H + 1)\sigma - 4\eta C\log_2(2\sigma)}{\eta\sigma\log_2(2\sigma)}.$$

We now give a lower bound on Γ, which will prove in particular that $\Gamma > 0$, and thus $k > 0$. Indeed, we have

$$\begin{aligned}
\Gamma &> \frac{\log_2(\sigma)}{1 + \log_2(\sigma)} - \frac{2H}{\eta\log_2(2\sigma)} - \frac{4C}{\sigma} \\
&= \left(1 - \frac{2H}{\eta\log_2(2\sigma)}\right) - \frac{4C}{\sigma} + \left(\frac{\log_2(\sigma)}{1 + \log_2(\sigma)} - 1\right).
\end{aligned}$$

Recall from the prior analysis with Markov's inequality that

$$1 - \frac{2H}{\eta \log_2(2\sigma)} \geq \frac{\gamma}{2+\gamma}.$$

Besides, for $\sigma \geq 8C(2+\gamma)/\gamma$, we get

$$\frac{4C}{\sigma} \leq \frac{1}{2}\frac{\gamma}{2+\gamma}$$

Finally, for $\sigma > 16^{(\gamma+2)/\gamma}$, we get

$$1 - \frac{\log_2(\sigma)}{1+\log_2(\sigma)} = \frac{1}{1+\log_2(\sigma)} \leq \frac{1}{4}\frac{\gamma}{2+\gamma}$$

and thus

$$\Gamma > \frac{1}{4}\frac{\gamma}{2+\gamma} > 0.$$

Then, we can bound the probability of error by

$$\frac{1}{k^2} < \frac{4}{\Gamma^2}\frac{\eta p(1-p)}{\eta^2} \leq 128\frac{(2+\gamma)^2}{\gamma^2}\frac{C}{\eta\sigma}.$$

Since $\eta \geq (2+\gamma)H/\log_2(\sigma)$, this is less than

$$128\frac{(2+\gamma)}{\gamma^2}\frac{C}{H}\frac{\log_2(\sigma)}{\sigma}$$

We make this less than ε by taking σ such that

$$\frac{\sigma}{\log_2(\sigma)} \geq A(\gamma, C, H, \varepsilon) := 128\frac{(2+\gamma)}{\gamma^2}\frac{C}{\varepsilon H},$$

which is satisfied for $\sigma \geq 2A\log_2(A)$. □

3 Example: Solutions of Polynomial Systems in Dimension Zero

In this section, we are interested in the solutions of polynomial equations with integer coefficients. As input, we are given polynomials $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{Z}[X_1, \ldots, X_n]^s$, and we make the assumption that $V(\boldsymbol{f})$ is finite; we recall height bounds on the description of $V(\boldsymbol{f})$, and give bounds on the number of "unlucky primes", in a sense that will be made precise. In all the following discussion, the height $\mathrm{ht}(f)$ of a polynomial f with rational coefficients is defined as the maximum of the heights of the minimal common denominator $v \in \mathbb{N}$ of all non-zero coefficients of f, and of the absolute values of the nonzero coefficients of vf.

Starting from $\boldsymbol{f}$, our output should be defined without ambiguity, but few quantitative results are known on, say, the height of coefficients in a Gröbner basis. Instead, we will target *zero-dimensional parameterizations* as our output, with the caveat that such objects are uniquely defined only after we choose a linear form.

Definition 2. Let $\mathbb{K}$ be a field, let $V \subset \overline{\mathbb{K}}^n$ be a finite set, and let λ be a linear form in the variables $X_1, \ldots, X_n$. We say that λ *separates* V if it takes pairwise distinct values on the points in V.

Suppose further that V is algebraic and defined over $\mathbb{K}$. If λ separates V, the *zero-dimensional parameterization* of V associated to λ is the unique family $\mathscr{Q} = ((q, v_1, \ldots, v_n), \lambda)$, with q monic and squarefree in $\mathbb{K}[T]$ and v_i of degree less than $\deg(q)$ in $\mathbb{K}[T]$ for all i, such that:

1. $\lambda(v_1, \ldots, v_n) = Tq' \bmod q$,
2. $V = \left\{ \left(\frac{v_1(\tau)}{q'(\tau)}, \ldots, \frac{v_n(\tau)}{q'(\tau)} \right) \mid q(\tau) = 0 \right\}$.

□

The first condition expresses the fact that the roots of q are the values that λ takes on V. We use rational functions to parameterize the points of V, with q' as a denominator, since when $\mathbb{K} = \mathbb{Q}$, this is known to result in v_i of height similar to that of q (we discuss this below).

In all that follows, we will assume that such a λ is part of the input. If one wanted to find λ, it is possible to explicitly build a set of size at most $n \deg(V)^2/2$ that is known to contain a separating λ [18, Lemma 2.1], with λ of height $O(n \log(d))$. One can also choose λ at random and guarantee that it separates V with arbitrarily high probability, by picking its entries in a set of sufficient cardinality [20, Lemma 11].

Definition 3. Let $\boldsymbol{f}$ be polynomials in $\mathbb{Z}[X_1, \ldots, X_n]$ with $V = V(\boldsymbol{f}) \subset \mathbb{C}^n$ of dimension zero, and let λ be a separating linear form for V. We say a prime p is lucky for $(\boldsymbol{f}, \lambda)$ if:

1. p does not divide any denominator in the polynomials of $\mathscr{Q}$,
2. $V_p := V(\boldsymbol{f} \bmod p) \subset \overline{\mathbb{F}}_p^n$ has dimension zero,
3. $\lambda_p := \lambda \bmod p$ is a separating element for V_p,
4. $\mathscr{Q} \bmod p$ is the zero-dimensional parameterization of V_p associated to λ_p.

Theorem 1. Let $\boldsymbol{f} = (f_1, \ldots, f_s)$ be in $\mathbb{Z}[X_1, \ldots, X_n]^s$, with $\deg(f_i) \leq d$ and $\mathrm{ht}(f_i) \leq h$ for all i, and let $V = V(\boldsymbol{f}) \subset \mathbb{C}^n$. Let further λ be a linear form that separates V, with integer coefficients of height at most ℓ. Then, all polynomials in the zero-dimensional parameterization $\mathscr{Q}$ of V associated to λ have height at most

$$H = d^n(3nh + \ell + \ln(n+1) + 1),$$

and the product of unlucky primes is an integer of height at most

$$C = (11n+4)d^{3n+1}h + (55n+99)\log((2n+5)s)d^{3n+2} + 8d^{2n}(nh+\ell)$$

All ingredients in the proof are in the literature. We start with the definition of *Chow forms* of zero-dimensional algebraic sets.

Definition 4. Let $\mathbb{K}$ be a field let $V \subset \overline{\mathbb{K}}^n$ be a zero dimensional algebraic set. A *Chow form* of V is a homogeneous polynomial of the form

$$C_{V,c} = c \prod_{\boldsymbol{\alpha} \in V} (T_0 + y_1 T_1 + \cdots + y_n T_n),$$

with $c \in \overline{\mathbb{K}} - \{0\}$. We call c the *leading coefficient* of $C_{V,c}$. □

If V is defined over $\mathbb{K}$, it admits Chow forms with coefficients in $\mathbb{K}$. In particular, if $\mathbb{K} = \mathbb{Q}$, V then admits two Chow forms with integer coefficients and contents 1; they differ by sign. The unique one with positive leading coefficient is called the *primitive Chow form* C_V of V. We denote by $c_V \in \mathbb{N}$ the corresponding leading coefficient.

When V is given as $V = V(\boldsymbol{f})$ as in the theorem, the bound on unlucky primes will be obtained as an immediate consequence of results due to D'Andrea, Ostafe, Shparlinski and Sombra [8], which will relate the primitive Chow form of $V = V(\boldsymbol{f})$ and the Chow forms of $V_p = V(\boldsymbol{f} \bmod p)$.

Indeed, D'Andrea *et al.* prove the existence of a positive integer $\mathfrak{A}$ (notation of the proof of their Theorem 2.1) such that if p does not divide $\mathfrak{A}$,

- the leading coefficient c_V does not vanish modulo p
- C_V remains squarefree modulo p
- $C_V \bmod p$ is a Chow form for a zero-dimensional algebraic set called $\mathcal{V}_p \subset \overline{\mathbb{F}}_p^n$, with $\mathcal{V}_p \subset V_p$ (we do not need to give the definition of this set, see *loc. cit.*)
- V_p has dimension zero and cardinality $\deg(V)$.

While this is not explicitly stated in that reference, it then follows directly that in this case, $\mathcal{V}_p = V_p$ and thus $C_V \bmod p$ is a Chow form of V_p (since $\mathcal{V}_p \subset V_p$ and both sets have cardinality $\deg(V)$).

Given any Chow form $C_{V,c}$ of V, it is well known (see [16], or [3] in the computer algebra literature) that a linear form $\lambda = \lambda_1 X_1 + \cdots + \lambda_n X_n$ separates V if and only if $C_{V,c}(T, \lambda_1, \ldots, \lambda_n)$ is squarefree (for any c), and if it is the case, we can form the zero-dimensional parameterization $\mathscr{Q} = ((q, v_1, \ldots, v_n), \lambda)$ by taking

$$\begin{aligned} q(T) &= \frac{1}{c} C_{V,c}(T, \lambda_1, \ldots, \lambda_n) \\ v_i &= -\frac{1}{c} \frac{\partial C_{V,c}}{\partial T_i}(T, \lambda_1, \ldots, \lambda_n), i = 1, \ldots, n. \end{aligned}$$

It follows that if p does not divide $\mathfrak{A}$ and if the discriminant of $C_V(T, \lambda_1, \ldots, \lambda_n) \bmod p$ is nonzero, $C_V \bmod p$ remains squarefree upon evaluation at $\lambda_1, \ldots, \lambda_n$, so $\lambda \bmod p$ remains a separating element for V_p, and the corresponding zero-dimensional parameterization of V_p is obtained by reducing the rational zero-dimensional parameterization modulo p. In other words, p is a lucky prime for $(\boldsymbol{f}, \lambda)$.

In order to complete the proof of the theorem, we first we find the height of the primitive Chow form C_V of V: from [8, Lemma 2.10] (which is based on an arithmetic Bézout theorem of [13, 7]) it follows that $\mathrm{ht}(C_V) \leq 3nd^n h$. If we assume that λ is separating for V and has height at most ℓ, then by [13, Lemma 1.2(1)(c)], the height of $Q = C_V(T, \lambda_1, \ldots, \lambda_n)$ must be at most

$$\mathrm{ht}(C_V) + d^n(\ell + \ln(n+1)) \leq d^n(3nh + \ell + \ln(n+1)).$$

The height of Q' is at most

$$\mathrm{ht}(Q) + n\ln(d) \leq d^n(3nh + \ell + \ln(n+1) + 1),$$

and the same holds for $\partial C_V/\partial T_i(T, \lambda_1, \ldots, \lambda_n)$. In particular, since dividing by c will not increase their height, this gives the claimed bound on the height of $\mathscr{Q}$.

Besides, we deduce from [13, end of section 1.1.1] that the discriminant of Q is a nonzero integer of height at most

$$2d^n(\mathrm{ht}(Q) + 3n\ln(d)) \leq 2d^{2n}(3nh + \ell + \ln(n+1) + 3).$$

Theorem 2.1 in [8] gives an upper bound on the height of $\mathfrak{A}$; taking this into account, we obtain the result claimed in the theorem, after a straightforward manipulation.

4 Comparative Analysis

Finally, we illustrate the bounds from Section 2 on the prime size needed to achieve a given probability of success for some sample problem sizes from either the polynomial system solving question in the previous section, or the computation of lexicographic Gröbner bases in two variables mentioned in the introduction.

We also include a comparison of the prime size when we do not allow for errors. Without error correction, in order to be able to reconstruct r, we need *all* primes to be lucky. The probability that a single prime is unlucky is at most $2C/\sigma$, so when choosing η primes, the probability that at least one of them is unlucky is at most

$$1 - \left(1 - \left(\frac{2C}{\sigma}\right)\right)^{\eta} \leq \frac{2\eta C}{\sigma}.$$

Choosing η as above, let us derive an estimate on σ that ensures $2\eta C/\sigma < \varepsilon$. We proceed informally and take

$$\eta \approx \frac{2H}{\log_2(\sigma)},$$

so our inequality is satisfied when $\sigma \log_2(\sigma) > 4CH/\varepsilon$. In particular, this is the case if $\sigma \geq 2B/\log(2, B)$, with $B(\gamma, C, H, \varepsilon) := 4CH/\varepsilon$.

For a fixed γ, the bounds in Proposition 1 and 2 are asymptotically stronger: a rough estimate shows that the primes we select with error correction are smaller by a factor of about $H/\log_2(C)$.

4.1 Solutions of polynomials systems in dimension zero

Reducing prime size with error-correction. Consider the following example where we solve a zero-dimensional polynomial system. Consider a system of 10 equations in 10 unknowns, where the polynomials have degree $d = 2$ and height $h = 50$, defining a zero-dimensional algebraic set. Our goal is to compute a zero-dimensional parameterization of the solutions. Assume the parameterization is defined with a separating linear form of height at most 50. Taking $\varepsilon = 0.0001$ and $\gamma = 0.25$, Proposition 1 provides estimates of

- $H \leq 1,590,680$, and
- $C \leq 34,449,366,690,000$.

For a success probability of $1-\varepsilon = 0.9999$, using the conventional approach (without allowing for errors) requires a prime size of at least 76 bits. When applying Proposition 1, the required prime size is reduced to 65 bits. In addition, Proposition 2 (Chebyshev) reduces the prime size even further to 57 bits. This last reduction is particularly significant, as a 57-bit prime falls within the range of machine primes. These results are summarized in the middle row of Table 2.

Next, consider the probability of success when using the reduced prime sizes estimated by Propositions 1 and 2, while maintaining the standard CRT approach without errors (that is, when we require all primes to be lucky). Using the prime size derived from Proposition 1, we compute:

- $P = \frac{2C}{\sigma} = 2.778 \times 10^{-6}$
- $\eta = \left\lceil \frac{2H}{\log_2(\sigma)} \right\rceil = 49,380$
- $(1-P)^{\eta} = 0.872$.

A success probability of 0.872 is insufficient for many practical applications.

Similarly, using the prime size estimated from Proposition 2 (Chebyshev), we obtain:

- $P = \frac{2C}{\sigma} = 6.928 \times 10^{-4}$
- $\eta = \left\lceil \frac{2H}{\log_2(\sigma)} \right\rceil = 56,343$
- $(1-P)^{\eta} = 1.100 \times 10^{-17}$.

The success probability of 1.100×10^{-17} is effectively zero. These results are summarized in Table 1.

Method	$\lceil \log_2(\sigma) \rceil$	P	η	$(1-P)^\eta$
Big prime	76	1.569e−9	42296	9.999e−1
Markov	65	2.778e−6	49380	8.720e−1
Chebyshev	57	6.928e−4	56343	1.100e−17

Table 1: Success probabilities $(1-P)^\eta$ for different prime sizes in solving a zero-dimensional polynomial system, where $P = \frac{2C}{\sigma}$ represents the probability that a chosen prime is unlucky, and $\eta = \left\lceil \frac{2H}{\log_2(\sigma)} \right\rceil$ is the number of selected primes. The table demonstrates how the probability of success in CRT without errors diminishes when using the same prime size as in the error-allowed cases.

Impact of probability of failure on prime size. Table 2 compares the prime sizes required for different methods when the probability of failure is set to $\varepsilon = 0.5$ versus $\varepsilon = 2^{-80}$, and also $\varepsilon = 0.0001$. When applying Propositions 1 (Markov) and 2 (Chebyshev), the results clearly show that reducing the probability of failure significantly increases the required prime size. Specifically, when ε decreases from 0.5 to 2^{-80}, the bit length of the prime more than doubles.

For instance, not allowing for errors, the required prime size increases from approximately 10^{19} (64 bits) to 10^{42} (142 bits). Similarly, the Markov and Chebyshev methods show similar growth, with their required prime sizes increasing from 53 to 132 bits and from 52 to 125 bits, respectively.

ε	Method	$\lceil \log_2(\sigma) \rceil$
0.5	CRT without errors	64
	Markov	53
	Chebyshev	52
0.0001	CRT without errors	76
	Markov	65
	Chebyshev	57
2^{-80}	CRT without errors	142
	Markov	132
	Chebyshev	125

Table 2: Comparison of required prime sizes $\lceil \log_2(\sigma) \rceil$ for different values of ε in solving a zero-dimensional polynomial system, using CRT without errors and Propositions 1 and 2. As ϵ decreases, the required prime size increases, with Markov and Chebyshev consistently estimating smaller primes than CRT without errors.

Remark 3. Let us briefly justify why one would consider such a large value as 0.5 for ε. If one is able to quickly verify whether the output is correct, $\varepsilon = 0.5$ is a suitable choice, since

it suffices to run the whole procedure an expected $O(1)$ time. Verifying that, for instance, we have all the solutions to a polynomial system is of course not straightforward, but one can perform a random verification as follows.

Suppose, in a general setting, that we have obtained a candidate r_0 for the rational number r (or for the polynomials, or matrix, ...) we want to reconstruct. Pick a single new prime p, large enough that p is lucky with overwhelming probability, say at least $1 - 2^{-80}$. As in Section 2, if we choose p at random in an interval $\sigma, \ldots, 2\sigma$, we should take $\sigma \geq 2^{81}C$: this is large, but we only deal with a single prime of this size. Then, compare the residues $r(p)$ (which is equal to r mod p if p is lucky) and r_0 mod p. If $r_0 = r$, this test always returns true. If $r_0 \neq r$, the test mistakenly returns true only if p divides the nonzero integer $uv_0 - u_0v$, if we write $r = u/v$ and $r_0 = u_0/v_0$. The base-2 logarithm of this integer is at most $2H + 1$, so if we also make sure that $\sigma \geq 2^{81}(2H + 1)$, this happens with probability at most 2^{-80}. □

Impact of γ. Table 3 presents the prime sizes $\lceil \log_2(\sigma) \rceil$ for different values of γ, comparing estimates obtained from Proposition 1 (Markov) and 2 (Chebyshev). As γ decreases, the prime size increases for both methods. For moderate size γ, Chebyshev predicts smaller primes by just under an order of magnitude (8-9 bits). However, for sufficiently small γ, Markov predicts much smaller primes compared to Chebyshev. Indeed, as γ becomes very small, the Chebyshev estimate grows significantly, predicting much larger prime sizes than Markov.

γ	Method	$\lceil \log_2(\sigma) \rceil$
1	Markov	62
	Chebyshev	53
0.25	Markov	65
	Chebyshev	57
0.1	Markov	66
	Chebyshev	84
0.05	Markov	67
	Chebyshev	164

Table 3: Required prime sizes $\lceil \log_2(\sigma) \rceil$ for different values of γ in solving a zero-dimensional polynomial system, varying γ while using Proposition 1 (Markov) and Proposition 2 (Chebyshev) to estimate the prime size.

4.2 Bivariate Gröbner bases

In [25], St-Pierre and Schost provide similar bounds H and C for the computation of bivariate Gröbner bases. In this case, an alternative approach based on Newton iteration exists, but with high complexity. Our work applies directly.

Consider the following example of computing a Gröbner basis for a bivariate polynomial system. Let the system consist of 10 equations in 2 unknowns, where the polynomials have degree $d = 10$ and height $h = 50$. Taking $\varepsilon = 0.0001$ and $\gamma = 0.25$, [25, Proposition 3.3] shows that we can take

$$H = C \leq 5725.$$

Table 4 presents the required prime sizes $\lceil \log_2(\sigma) \rceil$ for computing a bivariate Gröbner basis, comparing estimates obtained using the standard Chinese Remainder Theorem (CRT) approach without errors and those based on our bounds from Section 2, proposition 1 and 2. The Markov bound consistently reduces the required prime size compared to the CRT approach without errors. However, the Chebyshev bound is less effective with this example.

As with solving zero-dimensional polynomial systems, the results show as the probability of failure ε decreases, the required prime size increases significantly.

ε	Method	$\lceil \log_2(\sigma) \rceil$
0.5	CRT without errors	24
	Markov	20
	Chebyshev	36
0.0001	CRT without errors	36
	Markov	32
	Chebyshev	36
2^{-80}	CRT without errors	101
	Markov	99
	Chebyshev	100

Table 4: Comparison of required prime sizes $\lceil \log_2(\sigma) \rceil$ for different values of ε in the computation of a bivariate Gröbner basis. The table compares estimates from the standard Chinese Remainder Theorem (CRT) approach without errors and those obtained using Propositions 1 (Markov) and 2 (Chebyshev). As ε decreases, the required prime size increases, with Markov consistently estimating smaller primes than CRT without errors.

References

[1] Matteo Abbondati, Eleonora Guerrini, and Romain Lebreton. Decoding simultaneous rational evaluation codes. In *ISSAC '24*, page 153–161. ACM, 2024.

[2] M. Alonso, E. Becker, M.-F. Roy, and T. Wörmann. Zeroes, multiplicities and idempotents for zerodimensional systems. In *Algorithms in algebraic geometry and applications. Proceedings of MEGA'94*, volume 142 of *Progress in Mathematics*, pages 1–15. Birkhaüser, 1996.

[3] M. E. Alonso, E. Becker, M.-F. Roy, and T. Wörmann. *Zeros, multiplicities, and idempotents for zero-dimensional systems*, page 1–15. Birkhauser Verlag, CHE, 1996.

[4] J. Böhm, W. Decker, C. Fieker, S. Laplagne, and G. Pfister. Bad primes in computational algebraic geometry. In *Mathematical Software – ICMS 2016*, pages 93–101. Springer, 2016.

[5] J. Böhm, W. Decker, C. Fieker, and G. Pfister. The use of bad primes in rational reconstruction. *Math. Comput.*, 84:3013–3027, 2012.

[6] Curtis Bright and Arne Storjohann. Vector rational number reconstruction. In *ISSAC '11*, page 51–58. ACM, 2011.

[7] C. D'Andrea, T. Krick, and M. Sombra. Heights of varieties in muliprojective spaces and arithmetic nullstellensatz. *Annales scientifiques de l'École Normale Supérieure*, 46(4):549–627, Aug 2013.

[8] C. D'Andrea, A. Ostafe, I. Shparlinski, and M. Sombra. Reductions modulo primes of systems of polynomial equations and algebraic dynamical systems. *Trans. Amer. Math. Soc.*, 371(2):1169–1198, 2019.

[9] M. Giusti, K. Hägele, J. Heintz, J.-E. Morais, J.-L. Montaña, and L.-M. Pardo. Lower bounds for diophantine approximation. *J. of Pure and Applied Algebra*, 117/118:277–317, 1997.

[10] M. Giusti, J. Heintz, J.-E. Morais, J. Morgenstern, and L.-M. Pardo. Straight-line programs in geometric elimination theory. *Journal of Pure and Applied Algebra*, 124:101–146, 1998.

[11] M. Giusti, J. Heintz, J.-E. Morais, and L.-M. Pardo. When polynomial equation systems can be solved fast? In *AAECC-11*, volume 948 of *LNCS*, pages 205–231. Springer, 1995.

[12] M. Giusti, G. Lecerf, and B. Salvy. A Gröbner-free alternative for polynomial system solving. *Journal of Complexity*, 17(1):154–211, 2001.

[13] T. Krick, L.-M. Pardo, and M. Sombra. Sharp estimates for the arithmetic nullstellensatz. *Duke Mathematical Journal*, 109(3):521–598, 2001.

[14] L. Kronecker. Grundzüge einer arithmetischen Theorie der algebraischen Grössen. *Journal für die reine und angewandte Mathematik*, 92:1–122, 1882.

[15] G Lecerf. Quadratic newton iteration for systems with multiplicity. *Foundations of Computational Mathematics*, 2(3):247–293, 2002.

[16] F. Macaulay. *The Algebraic Theory of Modular Systems*. Cambridge University Press, 1916.

[17] C. Pernet. *High Performance and Reliable Algebraic Computing*. HDR, Université Joseph Fourier, Grenoble 1, November 2014.

[18] F. Rouillier. Solving zero-dimensional systems through the Rational Univariate Representation. *Applicable Algebra in Engineering, Communication and Computing*, 9(5):433–461, 1999.

[19] É. Schost. Computing parametric geometric resolutions. *Applicable Algebra in Engineering, Communication and Computing*, 13(5):349–393, 2003.

[20] É. Schost and M. Safey El Din. Bit complexity for multi-homogeneous system solving application to polynomial minimization. *Journal of Symbolic Computation*, 87:176–206, May 2018.

[21] Storjohann, A. Computation of hermite and smith normal forms of matrices. Master's thesis, University of Waterloo, 1994.

[22] W. Trinks. On improving approximate results of buchberger's algorithm by newton's method. *SIGSAM Bull.*, 18(3):7–11, 1984.

[23] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, third edition, 2013.

[24] Éric Schost and Catherine St-Pierre. Newton iteration for lexicographic gröbner bases in two variables. *Journal of Algebra*, 653:325–377, 2024.

[25] Éric Schost and Catherine St-Pierre. An m-adic algorithm for bivariate gröbner bases. *Journal of Symbolic Computation*, 128:102389, 2025.

Chapter 5

Bit-size estimates for computing roadmaps in smooth bounded real hypersurfaces

5.1 Introduction

5.1.1 Problem statement and motivation

A *roadmap* for an algebraic set $X \subset \mathbb{C}^n$ is an algebraic curve $\mathscr{R} \subset X$ with non-empty and connected intersection with each connected components of $X \cap \mathbb{R}^n$. Roadmaps efficiently decide connectivity queries by reducing higher dimensional computations to one-dimensional computations. In this way, a roadmap is like a skeleton which tracks connectivity. Deciding connectivity in real algebraic sets has applications that include robot motion planning (by deciding if paths exist between points) [9]. In fact, John Canny introduced the roamdap in his PhD thesis "On the Complexity of Robot Motion Planning" [8]. Roadmap computation is also used in more general algorithms in semi-algebraic geometry, including the decomposition of a semi-algebraic set into its semi-algebraically connected components [22].

Our starting point is the algorithm introduced by Safey El Din and Schost in [42] (*A baby steps/giant steps probabilistic algorithm for computing roadmaps in smooth bounded real hypersurface*). Computational costs were given in an algebraic complexity model, and bit complexity questions were left for future work. The algorithm also applies random changes of coordinates to establish certain required genericity properties. The main result roughly states that, for a degree d, square-free polynomial f in $\mathbb{Q}[X_1, \ldots, X_n]$ where

1. $V(f) \subset \mathbb{C}^n$ is smooth, and
2. $V(f) \cap \mathbb{R}^n$ is bounded,

there exists a randomized algorithm that, upon success, if all genericity properties are satisfied, computes a roadmap of $V(f)$ after performing $(nd)^{O(n^{1.5})}$ arithmetic operations. In this Chapter, assuming bounds on:

1. the height of f, and
2. the height of entries in the changes of coordinates we apply to f,

we give bit-size estimates showing that the height of the output follows the same order of complexity.

Remark 5.1.1. *In [6], bit complexity is stated for another algorithm for roadmap computation. The algorithm that we analyze here is one we expect to run efficiently in practice.*

5.1.2 Roadmaps and control points

In [42], the following definition of a roadmap of higher dimension is used. Roadmaps of higher dimension are defined to account for the recursive nature of the algorithm.

Definition. *An algebraic set $\mathscr{R} \subset \mathbb{C}^n$ is a roadmap of a variety V if:*

RM₁ *Each semi-algebraically connected component of $V \cap \mathbb{R}^n$ has non-empty and connected intersection with $\mathscr{R} \cap \mathbb{R}^n$.*

RM₂ *$\mathscr{R}$ is contained in V.*

In addition, $\mathscr{R}$ is said to be an i-roadmap of V when the following also holds:

RM₃ *The set $\mathscr{R}$ is either i-equidimensional or empty.*

The algorithm uses a finite set of *control points* $\mathscr{P}$ included in the input along with V. These points are used to test if the points of $\mathscr{P}$ are connected on $V \cap \mathbb{R}^n$.

Definition. *$\mathscr{R}$ is a roadmap (resp. i-roadmap) of $(V, \mathscr{P})$ if in addition:*

RM₄ *The set $\mathscr{R}$ contains $\mathscr{P} \cap V \cap \mathbb{R}^n$.*

5.1.3 Overview of the algorithm

Let f be a degree d square-free polynomial in $\mathbb{Q}[X_1, \ldots, X_n]$, and let V denote the hypersurface $V(f)$. Assume

1. $V \subset \mathbb{C}^n$ is smooth, and
2. $V \cap \mathbb{R}^n$ is bounded.

The algorithm uses a *baby steps / giant steps* design paradigm with the following two subroutines:

- CannyRoadmap for performing baby steps, based on the original roadmap algorithm from John Canny [8],
- MainRoadmap for performing giant steps.

The algorithm is based on the critical point method and the computation of polar varieties. See Chapter 2.2 for definitions and notation on polar varieties. In particular, the algorithm computes polar curves of polar points.

Canny roadmap. The original algorithm from John Canny [8] computes the polar curve $W(\pi_2, V)$. From the assumption that $V \cap \mathbb{R}^n$ is bounded, it follows that $W(\pi_2, V)$ intersects each connected component of $V \cap \mathbb{R}^n$. However, the intersections are not necessarily connected. The algorithm repairs connectivity by computing additional points. Consider

- $\mathscr{C} = K(\pi_1, W(\pi_2, V))$,
- $\mathscr{C}' = \pi_1(\mathscr{C})$, and
- $\mathscr{C}'' = V \cap \pi_1^{-1}(C')$.

Then, Canny proved

$$W(\pi_2, V) \cup \mathscr{C}''$$

is a $(n-2)$-roadmap of $V(f)$ of degree $d^{O(n)}$.

The algorithm is recursive where an additional variable is fixed at each recursive step until the output roadmap has dimension one. The recursive calls are geometrically equivalent to performing a recursive call with the input

$$f(x_i, X_2, \ldots, X_n), \quad \forall x_i \in \mathscr{C}'.$$

At each recursive call the number of control points computed is multiplied by $d^{O(n)}$ while the dimension of the input decreases by one. The depth of the recursion is n, and therefore the roadmap obtained from Canny's algorithm has degree $d^{O(n^2)}$.

Main roadmap. To obtain the improved complexity of $(nd)^{O(n^{1.5})}$, Safey El Din and Schost show how to reduce the number of recursive calls from n to $\sqrt{n}$.

Now let

- $\mathscr{C} = K(\pi_1, W(\pi_i, V))$,

- $\mathscr{C}' = \pi_{i-1}, (\mathscr{C})$, and
- $\mathscr{C}'' = V \cap \pi_{i-1}^{-1}(\mathscr{C}')$.

Safey El Din and Schost show that, for $i \approx \sqrt{n}$, under some genericity assumptions,

$$W(\pi_i, V) \cup \mathscr{C}''$$

is a roadmap of V of dimension $\max(i-1, n-i)$. Since the depth of the recursion decreases from n to $\sqrt{n}$, the degree of the output decreases to

$$(d^{O(n)})^{\sqrt{n}} = d^{O(n^{1.5})}.$$

5.1.4 Data structures

Zero-dimensional parameterizations. The algorithm processes zero- and one-dimensional algebraic sets represented through parameterizations.

Consider a zero-dimensional algebraic set $S \subset \mathbb{C}^n$ defined over $\mathbb{Q}$. A *zero-dimensional parametrization*

$$\mathscr{Q} = ((q, v_1, \ldots, v_n), \lambda),$$

of S is a data structure with

- univariate polynomials $q \in \mathbb{Q}[T]$, monic and squarefree, and $v_i \in \mathbb{Q}[T]$ with $\deg(v_i) < \deg(q)$;
- a $\mathbb{Q}$-linear form λ in the variables $X_1, \ldots, X_n$,

such that:

1. $\lambda(v_1, \ldots, v_n) = Tq' \bmod q$,
2. $S = \left\{ \left(\frac{v_1(\tau)}{q'(\tau)}, \ldots, \frac{v_n(\tau)}{q'(\tau)} \right) \mid q(\tau) = 0 \right\}$.

The constraint on λ ensures that the roots of q are the values taken by λ on S; we call these λ *separating*.

Given a zero-dimensional parameterization $\mathscr{Q}$, the defining algebraic set V is written $\mathsf{Z}(\mathscr{Q})$.

One-dimensional parameterizations. The output of the main algorithm is a data structure representing an algebraic curve called a one-dimensional parameterization.

Let $C \subset \mathbb{C}^n$ be a one-dimensional algebraic set defined over $\mathbb{Q}$ (a curve). A *one-dimensional parametrization* is a pair

$$\mathscr{Q} = ((q, v_1, \ldots, v_n), (\lambda, \lambda')),$$

with

- bi-variate polynomials $(q, v_1, \ldots, v_n) \in \mathbb{Q}[T, U]$, q square-free and monic in T and U, with $\mathrm{GCD}(q, v_1) = 1$ and $\deg(v_i) < \deg(q)$;
- $\lambda = \lambda_1 X_1 + \ldots + \lambda_n X_n, \lambda' = \lambda'_1 X_1 + \ldots + \lambda'_n X_n$ linear forms in $X_1, \ldots, X_n$, with
  - $$\lambda(v_1, \ldots, v_n) = T\frac{\partial q}{\partial T} \mod q;$$
  - $$\lambda'(v_1, \ldots, v_n) = U\frac{\partial q}{\partial T} \mod q.$$

Then C is the Zariski closure

$$\mathsf{Z}(\mathscr{Q}) := \overline{\left\{ \left(\frac{v_1(\tau,\rho)}{\partial q/\partial T(\tau,\rho)}, \ldots, \frac{v_n(\tau,\rho)}{\partial q/\partial T(\tau,\rho)} \right) \in \mathbb{Q}^n \mid q(\tau,\rho) = 0, \frac{\partial q}{\partial T}(\tau,\rho) \neq 0 \right\}}.$$

Remark 5.1.2. *The proof of the main result is based on geometric analysis without actually referring to data structures (until bounding the height of the data structure in the output). The proof could be written without reference to zero-dimensional parameterizations.*

5.1.5 Main result

The following are definitions of height for rational numbers and polynomials with rational coefficients.

- The height of a rational number $r = u/v$ in $\mathbb{Q} \setminus \{0\}$, denoted $\mathrm{ht}(r)$, is the maximum of $\log|u|$ and $\log v$, where $u \in \mathbb{Z}$ and $v \in \mathbb{N}$ are coprime.
- The height of a polynomial with rational coefficients $f \in \mathbb{Q}[X_1, \ldots, X_n]$, denoted $\mathrm{ht}(f)$, where $v \in \mathbb{N}$ is the smallest common denominator of its nonzero coefficients, is the maximum of $\log v$ and the logarithms of the absolute values of the integer coefficients of vf.

By using the Lagrangian modeling of polar varieties we obtain equations that are a regular sequence, and we are then able to apply techniques using intersection theory and the *arithmetic Chow ring* from [16] to obtain our height bounds. For a polynomial system $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{Q}[X_1, \ldots, X_n]$, the Lagrangian formulation involves a generically chosen vector of integers $\boldsymbol{u}$ in $\mathbb{Z}^s$, which defines the linear form

$$\sum_{i=1}^{s} u_i L_i - 1$$

where $(L_1, \ldots, L_s)$ are the corresponding Lagrange multipliers. We let $h_{\boldsymbol{u}}$ bound the height of coefficients in this linear form.

Our main result is the following.

Theorem 5.1.3. *(Main result) Consider a square-free polynomial f in $\mathbb{Q}[X_1, \ldots, X_n]$ of degree d and height h, such that*

1. *$V(f) \subset \mathbb{C}^n$ is smooth, and*
2. *$V(f) \cap \mathbb{R}^n$ is bounded.*

Also let $\mathscr{P}$ be a set of control points of degree $\mathfrak{d}$ and height $\mathfrak{h}$. Then, upon success, if all genericity properties are satisfied, the height of the one-dimensional parameterization of the 1-roadmap of $(V(f), \mathscr{P})$ in the output of [42, MainRoadmap] is

$$O^{\sim}\left(\mathfrak{h} d^{24n^{3/2}} + \mathfrak{d}(h + h_{\boldsymbol{A}} + h_{\boldsymbol{u}} + h_{\boldsymbol{A}^{-1}}) d^{30n^{3/2}+1}\right)$$

where

- *$h_{\boldsymbol{A}}$ bounds the height of entries in the change of coordinates matrix $\boldsymbol{A}$;*
- *$h_{\boldsymbol{u}}$ bounds the height of the coefficients in the linear forms of the Lagrange systems;*
- *$h_{\boldsymbol{A}^{-1}}$ bounds the height of entries in the inverse change of coordinates matrix $\boldsymbol{A}^{-1}$.*

Remark 5.1.4. *The asymptotic notation in the main result uses soft-O notation, $O^{\sim}(\cdot)$, which ignores polylogarithmic factors.*

Remark 5.1.5. *The height is as expected with exponent $O(n^{1.5})$.*

5.2 Preliminaries

5.2.1 Projective geometry

The following preliminaries on projective geometry are based on [45, Chapter 1].

Projective space. Let k be a field. *Projective n-space over k*, denoted $\mathbb{P}^n$ or $\mathbb{P}^n(\bar{k})$ when the field needs to be specified, is defined as the sets of all lines in affine space through the origin. Consider the set of all $(n+1)$-tuples

$$(x_0, \ldots, x_n) \in \mathbb{A}^{n+1}(k)$$

with the requirement that not all coordinates are zero. Consider these tuples modulo an equivalence relation defined as

$$(x_0, \ldots, x_n) \sim (y_0, \ldots, y_n)$$

when there exists $\lambda \in \bar{k}^*$ and

$$(x_0, \ldots, x_n) = \lambda(y_0, \ldots, y_n).$$

Let $[x_0, \ldots, x_n]$ be the equivalence class

$$\{(\lambda x_0, \ldots, \lambda x_n) : \lambda \in \bar{k}^*\}.$$

The variables $X = (X_0, \ldots, X_n)$ are called *homogeneous coordinates*.

Projective varieties. Consider a degree d polynomials

$$f \in \bar{k}[X] = \bar{k}[X_0, \ldots, X_n]$$

with the property that, for all $\lambda \in \bar{k}$,

$$f(\lambda X_0, \ldots, \lambda X_n) = \lambda^d f(X_0, \ldots, X_n).$$

We call these polynomial *homogeneous*, and we call an ideal generated by homogeneous polynomials a *homogeneous ideal*.

Given a non-homogeneous degree d polynomial

$$f = \sum_{\alpha} c_\alpha X_1^{\alpha_1} X_2^{\alpha_2} \ldots X_n^{\alpha_n} \in k[X_1, \ldots, X_n],$$

the *homogenization* of f, denoted $f^h \in k[X_0, \ldots, X_n]$, is a homogeneous polynomial where each monomial of f:

$$c_\alpha X_1^{\alpha_1} X_2^{\alpha_2} \ldots X_n^{\alpha_n},$$

is replaced with

$$c_\alpha X_0^{d-|\alpha|} X_1^{\alpha_1} X_2^{\alpha_2} \ldots x_n^{\alpha_n},$$

where

$$|\alpha| = \sum_{i=1}^{n} \alpha_i$$

is the total degree of the monomial.

Now let f be a homogeneous polynomial and let P be a point in $\mathbb{P}^n$. A *projective algebraic set* is an algebraic set of the form

$$\{P \in \mathbb{P}^n : f(P) = 0 \text{ for all homogeneous } f \in I\}$$

for a homogeneous ideal I.

Consider an affine algebraic set $V \subset \mathbb{A}^n(k)$ with ideal $I(V)$, and consider V as a subset of $\mathbb{P}^n$. Then, the *projective closure* of V, denoted by $\bar{V}$, is the projective algebraic set with homogeneous ideal $I(\bar{V})$ generated by

$$\{f^h(X) : f \in I(V)\}.$$

Ruled joins. For $i \in \{1, 2\}$, consider irreducible algebraic sets $A_i \subset \mathbb{P}^{n_i}_{\mathbb{C}}$ for integers n_i, and let $I(A_i) \subset \mathbb{C}[X_i]$ denote the ideal of A_i. Then, the ruled join $A_1 \# A_2$ is the irreducible subvariety of $\mathbb{P}^{n_1+n_2+1}_{\mathbb{C}}$ defined by the homogeneous ideal which is generated by

$$I(A_1) \cup I(A_2) \subset \mathbb{C}[X_1, X_2].$$

5.2.2 The arithmetic Chow ring

We use the arithmetic Chow ring to provide our bit-size estimates, introduced by D'Andrea, Krick, and Sombra in [16]. Consider the ring of truncated power series

$$\mathbb{Z}[\theta]/\langle\theta^{n+1}\rangle;$$

this is the *Chow ring* of the projective space $\mathbb{P}^n(\overline{k})$, a tool developed for the study of intersection theory. Our height bounds are based on an extended, arithmetic version, the *arithmetic Chow ring of* $\mathbb{Q}$:

$$\mathbb{R}[\eta, \theta]/\langle\eta^2, \theta^{n+1}\rangle.$$

5.2.3 Canonical height

Consider a 0-dimensional algebraic set $V \subset \mathbb{P}^n$ defined over $\mathbb{Q}$. The class of V :

$$[V]_{\mathbb{Z}} \in \mathbb{R}[\eta, \theta]/\langle\eta^2, \theta^{n+1}\rangle$$

is associated to a homogeneous polynomial of degree n :

$$[V]_{\mathbb{Z}} = \hat{h}(V)\eta\theta^{n-1} + \deg(V)\theta^n.$$

The degree, $\deg(V)$, is its degree as an algebraic set. The definition of *canonical height*, $\hat{h}(V)$, is given in [16]. For simplicity, we do not define canonical height explicitly. Instead we only state the properties we use, which are the following.

1. $\hat{h}(V) \geq 0$; and thus $[V]_{\mathbb{Z}} \geq 0$, where inequalities in arithmetic Chow rings should be interpreted coefficientwise.

2. For a square-free and primitive polynomial

$$f \in \mathbb{Z}[X_1, \ldots, X_n],$$

it follows from [16, Proposition 2.53] that

$$[V(f)]_{\mathbb{Z}} = m(f)\eta + \deg(f)\theta$$

where

$$m(f) = \int_{S_1^{n+1}} \log(|f|) d\mu^{n+1}$$

is the *Mahler measure* of f with respect to the Haar measure μ of mass 1 on the complex unit circle S_1.

3. Consider a polynomial

$$f \in \mathbb{Z}[X_1, \ldots, X_n].$$

From Corollary 2.61 and Lemma 2.30 in [16],

$$[V \cap V(f^h)]_{\mathbb{Z}} \leq [V]_{\mathbb{Z}} \cdot \left((\mathrm{ht}(f^h) + \log(n+1)\deg(f^h))\eta + \deg(f^h)\theta\right).$$

5.3 The algorithm

5.3.1 Genericity assumptions

The algorithm applies randomly chosen changes of coordinates $\boldsymbol{A} \in \mathbb{Z}^{n\times n}$ to the input polynomials which generically ensure certain desirable geometric properties.

Let $\boldsymbol{f} = (f_1, \ldots, f_s)$ in $\mathbb{Q}[X_1, \ldots, X_n]^s$ be a sequence of degree d polynomials and $V = V(\boldsymbol{f}) \subset \mathbb{C}^n$ their corresponding algebraic set. Consider the following genericity properties originally defined in [42].

$\boldsymbol{H}(1)$: The ideal $\langle \boldsymbol{f} \rangle$ is squarefree;

$\boldsymbol{H}(2)$: $V = V(\boldsymbol{f})$ is equidimensional of positive dimension $n - s$;

$\boldsymbol{H}(3)$: $\mathrm{sing}(V)$ is finite;

$\boldsymbol{H}(4)$: $V \cap \mathbb{R}^n$ is bounded.

For $i \in \{2, \ldots, n-s\}$:

$\boldsymbol{H}_i'(1)$: V is in Noether position for π_{n-s};

$\boldsymbol{H}_i'(2)$: either $W(\pi_i, V)$ is empty or $(i-1)$-equidimensional and in Noether position for π_{i-1};

$\boldsymbol{H}_i'(3)$: $K(\pi_1, V)$ is finite;

$\boldsymbol{H}_i'(4)$: $K(\pi_1, W(\pi_i, V))$ is finite;

$\boldsymbol{H}_i'(5)$: for $\boldsymbol{x} \in W(\pi_i, V) - \mathrm{sing}(V), \mathrm{jac}_{\boldsymbol{x}}([\boldsymbol{f}, \Delta], [X_1, \ldots, X_n])$ has rank $n-(i-1)$,

$$\Delta = [\partial f/\partial X_i \mid i \in [e+i+1, \ldots, n]],$$

where $e \in \mathbb{N}$ is the depth of the recursion.

Safey El Din and Schost also prove the following Lemmas specifying the genericity assumptions of the algorithm.

- [42, Lemma 31:] Suppose that $n-s-e > 1$ and that $[f, Q]$ satisfies $\boldsymbol{H}$. After a generic change of variables in $\mathrm{GL}(n, e)$, the system $[f, Q]$ satisfies $\boldsymbol{H}_2'$ as well.
- [42, Lemma 34.] Suppose that $n - e > i$ and that $[f, Q]$ satisfies $\boldsymbol{H}$. After a generic change of variables in $\mathrm{GL}(n, e)$,
  (a) $[f, Q]$ satisfies $\boldsymbol{H}$ and $\boldsymbol{H}_i'$;
  (b) $[f, Q]$ satisfies $\boldsymbol{H}$.

5.3.2 Fibers of projections

Let $\boldsymbol{f} = (f_1, \ldots, f_s) \in \mathbb{Q}[X_1, \ldots, X_n]^s$ be a sequence of polynomials and $Q = Q(X_1, \ldots, X_e)$ a zero-dimensional parameterization. The algorithms are recursive, processing pairs of the form $[\boldsymbol{f}, Q]$ where

$$V([\boldsymbol{f}, Q]) = V \cap \pi_e^{-1}(\mathsf{Z}(Q)),$$

which restricts $X_1, \ldots, X_e$ to a finite number of values described by Q. Also let $P = P(X_1, \ldots, X_n)$ be a zero-dimensional parameterization for the set of control points $\mathscr{P}$.

To analyze fibers of projections throughout the recursion, following the original work in [42], we introduce the following notation. For $\boldsymbol{x} = (x_1, \ldots, x_e) \in \mathbb{C}^e$ and $1 \leq j \leq s$, define

$$f_{j,\boldsymbol{x}} = f_j(x_1, \ldots, x_e, X_{e+1}, \ldots, X_n)$$

and

$$\boldsymbol{f}_x = (f_{1,\boldsymbol{x}}, \ldots, f_{s,\boldsymbol{x}}).$$

The pair $[\boldsymbol{f}, Q]$ is said to satisfy $\boldsymbol{H}$ if, for all $\boldsymbol{x} \in \mathsf{Z}(Q)$, the system $\boldsymbol{f}_x$ satisfies $\boldsymbol{H}$ in $\mathbb{C}[X_{e+1}, \ldots, X_n]$. In particular, the variety $V([\boldsymbol{f}, Q])$ is equidimensional of dimension $n - e - s > 0$. Assuming $n - s \geq 2$ and fixing an integer $i \in \{2, \ldots, n-s\}$, $[\boldsymbol{f}, Q]$ satisfies $\boldsymbol{H}'_i$ if, for all $\boldsymbol{x} \in \mathsf{Z}(Q)$, the system $\boldsymbol{f}_x$ satisfies $\boldsymbol{H}'_i$ in $\mathbb{C}[X_{e+1}, \ldots, X_n]$.

Recall that we use the Lagrangian modeling of polar varieties when obtaining our height estimates.

Let $\pi_{e,i}$ denote the projection

$$\begin{array}{ccc} \mathbb{C}^n & \to & \mathbb{C}^i \\ (x_1, \ldots, x_n) & \mapsto & (x_e, \ldots, x_i). \end{array}$$

Remark 5.3.1. *Notice* $\pi_i = \pi_{0,i}$.

Based on the approach from [42] (and using notation introduced in Chapter 2), we define

- $w_{i,Q} := w(\pi_{e,i}, V[\boldsymbol{f}, Q])$,
- $W_{i,Q} := \overline{w_{i,Q}}$, and
- $\mathscr{W}_{\boldsymbol{u},i,Q} := \mathscr{W}_{\boldsymbol{u}}(\pi_{e,i}, V([\boldsymbol{f}, Q]))$

and define

- $\mathscr{C}_Q = K(\pi_{e,1}, V([\boldsymbol{f}, Q]))$,
- $\mathscr{C}'_Q = \pi_{e+i-1}(\mathscr{C}_Q)$
- $\mathscr{C}''_Q = V \cap \pi^{-1}_{e+i-1}(\mathscr{C}'_{Q'})$.

5.3.3 Subroutines

Based directly on the notation and definitions from [42], we define the following subroutines.

Unions of parameterizations. Consider zero-dimensional (resp. one-dimensional) parameterizations Q and Q' of zero-dimensional (resp. one-dimensional) algebraic sets. Then, $\mathsf{Union}(Q, Q')$ computes a parameterization of $\mathsf{Z}(Q) \cup \mathsf{Z}(Q')$.

Polynomial system solving. Let $\mathsf{Solve}(\boldsymbol{f}, Q, j)$ be the subroutine that return a parametrization of the j-dimensional component of $V([\boldsymbol{f}, Q])$, with $j \in \{0, 1\}$.

Projections. For $1 \leq r < k \leq n$, let $\mathsf{Projection}(Q, [X_r, \ldots, X_k])$ denote the subroutine that computes a parameterization $R(X_r, \ldots, X_k)$ with $\mathsf{Z}(R) = \pi_{r,k}(\mathsf{Z}(Q))$.

Changes of variables. Consider polynomials $\boldsymbol{f} = (f_1, \ldots, f_s) \in \mathbb{C}[X_1, \ldots, X_n]^s$ and a matrix $\boldsymbol{A} \in \mathbb{Z}^{n \times n}$. We let $\mathsf{ChangeVariables}(\boldsymbol{A}, \boldsymbol{f})$ denote the subroutine that computes $\boldsymbol{f}^{\boldsymbol{A}}$.

Fibers. Consider zero-dimensional parameterizations $Q = Q(X_1, \ldots, X_e)$ and $P = P(X_1, \ldots, X_n)$. The subroutine $\mathsf{Lift}(P, Q)$ returns a zero-dimensional parameterization of the set

$$\mathsf{Z}(P) \cap \pi_e^{-1}(\mathsf{Z}(Q)).$$

Consider polynomials $\boldsymbol{f} = (f_1, \ldots, f_s) \in \mathbb{Q}[X_1, \ldots, X_n]^s$ and a zero-dimensional parameterization $Q = Q(X_1, \ldots, X_e)$ satisfying $\boldsymbol{H}$ and $\boldsymbol{H}'_i$. Also consider a zero-dimensional parameterization $Q'(X_1, \ldots, X_{e+i-1})$ of $\mathscr{C}'_Q$. The subroutine $\mathsf{LiftW}(\boldsymbol{f}, Q')$ returns a zero-dimensional parameterization R with

$$W_{i,Q} \cap \pi_{e+i-1}^{-1}(\mathsf{Z}(Q')) \subset \mathsf{Z}(R).$$

Critical set computations. The roadmap subroutines compute a variety of critical sets. Again let $\boldsymbol{f} = (f_1, \ldots, f_s) \in \mathbb{Q}[X_1, \ldots, X_n]^s$ be a sequence of polynomials and $Q = Q(X_1, \ldots, X_e)$ be a zero-dimensional parameterization. The subroutine $\mathsf{CriticalCurve}(\boldsymbol{f}, Q)$ returns a one-dimensional parameterization of $W_{2,Q}$, assuming $\boldsymbol{f}$ and Q satisfy $\boldsymbol{H}$ and $\boldsymbol{H}'_2$. The subroutine $\mathsf{requiredCriticalPoints}(\boldsymbol{f}, Q, i)$ computes a zero-dimensional parameterization of the set

$$K(\pi_{e,1}, W_{i,Q})$$

Remark 5.3.2. *In the baby steps / giant steps paper by Safey El Din and Schost [42],* $\mathsf{requiredCriticalPoints}(\boldsymbol{f}, Q)$ *returns a parameterization of the set*

$$K(\pi_{e,1}, V([\boldsymbol{f}, Q])) \cup K(\pi_{e,1}, W_{i,Q}).$$

However, it was shown in [36] that computing $K(\pi_{e,1}, W_{i,Q})$ *alone is sufficient.*

Roadmap subroutines. Algorithm 1 presents pseudocode of Canny's algorithm as given in [42], and Algorithm 2 presents pseudocode of the main roadmap subroutine in [42]. While letting the pseudocode be a guide, we analyze both subroutines, bounding the height of the output.

Algorithm 1: Canny Roadmap

Input: a sequence of polynomials $\boldsymbol{f} = (f_1, \ldots, f_s) \in \mathbb{Q}[X_1, \ldots, X_n]^s$, and zero-dimensional parameterizations $Q = Q(X_1, \ldots, X_e)$ and $P = P(X_1, \ldots, X_n)$

Output: a one-dimensional parameterization of a roadmap $\mathscr{R}$ of $(V([\boldsymbol{f}, Q]), \mathsf{Z}(P))$

if $n - s - e = 1$ **then**
  return $\mathsf{Solve}(\boldsymbol{f}, Q, 1)$
Let $\boldsymbol{A}$ be a random matrix in $\mathrm{GL}(n, e)$
Let $\boldsymbol{f} \leftarrow \mathsf{ChangeVariables}(\boldsymbol{A}, \boldsymbol{f})$
Let $P \leftarrow \mathsf{ChangeVariables}(\boldsymbol{A}, P)$
$P \leftarrow \mathsf{Lift}(P, Q)$
$C \leftarrow \mathsf{Union}(\mathsf{requiredCriticalPoints}(\boldsymbol{f}, Q, 2), P)$
$Q' \leftarrow \mathsf{Projection}(C, [X_1, \ldots, X_{e+1}])$
$P' \leftarrow \mathsf{Union}(\mathsf{LiftW}(\boldsymbol{f}, Q'), P)$
if $C \neq -1$ **then**
  $\mathscr{R} \leftarrow \mathsf{CannyRoadmap}(\boldsymbol{f}, Q', P')$
else
  $\mathscr{R} \leftarrow (-1)$
$\mathscr{R}' \leftarrow \mathsf{CriticalCurve}(\boldsymbol{f}, Q)$
$\mathscr{R}'' \leftarrow \mathsf{Union}(\mathscr{R}, \mathscr{R}')$
return $\mathsf{ChangeVariables}(\boldsymbol{A}^{-1}, \mathscr{R}'')$

5.4 Proof of Theorem 5.1.3

To bound the height of the output of the roadmap subroutines, a straightforward application of the *arithmetic Bézout theorem* (see [30]) is insufficient. The problem arises from solving polynomial equations with a special shape. The variables are divided into two separate blocks, each subject to different constraints:

- $X_1, \ldots, X_e$ with high degree and bit-size, and
- $X_{e+1}, \ldots, X_n$ with low degree and bit-size,

where $e \in \mathbb{N}$ is the depth of the recursion. We instead use techniques introduced by D'Andrea, Krick and Sombra [16] that involve the arithmetic Chow ring and which precisely allow the two blocks to be handled separately. To apply these techniques, we use the Lagrangian modeling of polar varieties, as their equations form a regular sequence.

Remark 5.4.1. *In what follows,* log *refers to the natural logarithm.*

Algorithm 2: Main Roadmap

Input: a square-free polynomial $f \in \mathbb{Q}[X_1, \ldots, X_n]$, zero-dimensional parameterizations $Q = Q(X_1, \ldots, X_e)$ and $P = P(X_1, \ldots, X_n)$

Output: a one-dimensional parameterization of a roadmap $\mathscr{R}$ of $(V([f, Q]), \mathsf{Z}(P))$

if $n - e \leq \lceil \sqrt{n} \rceil$ **then**
  Return CannyRoadmap(f, Q)
Let $\boldsymbol{A}$ be a random matrix in $\mathrm{GL}(n, e)$
Let $f \leftarrow$ChangeVariables$(\boldsymbol{A}, f)$
Let $P \leftarrow$ChangeVariables$(\boldsymbol{A}, P)$
$P \leftarrow$Lift(P, Q)
$C \leftarrow$Union(requiredCriticalPoints$(f, Q, \lceil \sqrt{n} \rceil), P)$
$Q' \leftarrow$Projection$(Q, [X_1, \ldots, X_{e+1}])$
$P' \leftarrow$Union(LiftW$(f, Q'), P)$
if $C \neq -1$ **then**
  $R \leftarrow$ MainRoadmap$(f, Q', P', \lceil \sqrt{n} \rceil)$
else
  $R \leftarrow (-1)$
$\Delta \leftarrow \left[\frac{\partial f}{\partial x_i} \mid i \in [e + \lceil \sqrt{n} \rceil + 1, \ldots, n]\right], \boldsymbol{f} \leftarrow (f, \Delta)$
$R' \leftarrow$CannyRoadmap$(\boldsymbol{f}, Q, P')$
$\mathscr{R}'' \leftarrow$Union$(\mathscr{R}, \mathscr{R}')$
Return ChangeVariables$(\boldsymbol{A}^{-1}, \mathscr{R}'')$

5.4.1 Nested Lagrange systems

Recall from Chapter 2, for $\boldsymbol{u} \in \mathbb{Z}^s$ and $i \in \{1, \ldots, n - s + 1\}$, we let $\mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f})$ denote the polynomials

$$\left(\boldsymbol{f}, \boldsymbol{Lagrange}(\boldsymbol{f}, i, (L_1, \ldots, L_s)), \sum_{i=1}^{s} u_i L_i - 1\right) \in \mathbb{C}[X_1, \ldots, X_n, L_1, \ldots, L_s]^{s+n-i+1},$$

and we let $\mathscr{W}_{\boldsymbol{u}}(\pi_i, V(\boldsymbol{f}))$ denote the Lagrange polar variety

$$\mathscr{W}_{\boldsymbol{u}}(\pi_i, V(\boldsymbol{f})) = V(\mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f})) \subset \mathbb{C}^{n+s}.$$

Recall again, from Proposition 3.1 in Chapter 3, and the discussion above the statement of Proposition 3.1, for a *generic* choice of $\boldsymbol{u} \in \mathbb{Z}^s$, the Zariski closure of the projection of the zero-set of $\mathscr{W}_{\boldsymbol{u}}(\pi_i, V(\boldsymbol{f}))$ on the $X_1, \ldots, X_n$-space is the polar variety $W(\pi_i, V(\boldsymbol{f}))$.

Remark 5.4.2. *Note that $\mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f})$ is a polynomial system with*

- $s + n - i + 1$ *equations, and*

- $s+n$ *unknowns.*

Now, for $\boldsymbol{v} \in \mathbb{Z}^{s+n-i+1}$, consider the polar variety

$$\mathscr{W}_{\boldsymbol{v}}(\pi_1, \mathscr{W}_{\boldsymbol{u}}(\pi_i, V(\boldsymbol{f}))) \subset \mathbb{C}^{2s+2n-i+1},$$

defied by polynomials

$$\mathscr{I}_{\boldsymbol{v}}(1, \mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f})) = \left(\mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f}), \boldsymbol{Lagrange}(\mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f}), 1, [L'_1 \cdots L'_{s+n-i+1}]), \sum_{i=1}^{s+n-i+1} v_i L'_i - 1 \right)$$

in $\mathbb{C}[X_1, \ldots, X_n, L_1, \ldots, L_s, L_1, \ldots, L_{s+n-i+1}]$. By the discussion given previously regarding the Lagrangian modelling of polar varieties, but where this time we apply Proposition 3.1 from Chapter 3 twice, for generic choices of both $\boldsymbol{u}$ and $\boldsymbol{v}$, the polar variety $W(\pi_1, W(\pi_i, V(\boldsymbol{f})))$ is equal to the Zariski closure of the projection on the $X_1, \ldots, X_n$ variables of the algebraic set defined by the vanishing of equations $\mathscr{I}_{\boldsymbol{v}}(1, \mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f}))$.

Remark 5.4.3. *Note that $\mathscr{I}_{\boldsymbol{v}}(1, \mathscr{I}_{\boldsymbol{u}}(i, \boldsymbol{f}))$ is a polynomial system with*

- $2(s+n)-i+1$ *equations, and*
- $2(s+n)-i+1$ *unknowns.*

After fixing e variables. Once e variables are fixed,

$$\boldsymbol{Lagrange}(\boldsymbol{f}, i, (L_1, \ldots, L_s)) = [L_1, \ldots, L_s] \cdot \begin{bmatrix} \frac{\partial f_1}{\partial X_{i+1}} & \cdots & \frac{\partial f_1}{\partial X_n} \\ \vdots & & \vdots \\ \frac{\partial f_s}{\partial X_{i+1}} & \cdots & \frac{\partial f_s}{\partial X_n} \end{bmatrix}$$

is further truncated to

$$\boldsymbol{Lagrange}([\boldsymbol{f}, Q], i, (L_1, \ldots, L_s)) := [L_1, \ldots, L_s] \cdot \begin{bmatrix} \frac{\partial f_1}{\partial X_{e+i+1}} & \cdots & \frac{\partial f_1}{\partial X_n} \\ \vdots & & \vdots \\ \frac{\partial f_s}{\partial X_{e+i+1}} & \cdots & \frac{\partial f_s}{\partial X_n} \end{bmatrix}$$

so that $\mathscr{W}_{\boldsymbol{u},i,Q}$ is defined by the equations

$$\mathscr{I}_{\boldsymbol{u}}(i, [\boldsymbol{f}, Q]) := \left(\boldsymbol{f}, \boldsymbol{Lagrange}([\boldsymbol{f}, Q], i, (L_1, \ldots, L_s)), \sum_{i=1}^{s} u_i L_i - 1 \right),$$

in

$$\mathbb{C}[X_{e+1}, \ldots, X_n, L_1, \ldots, L_s]^{s+n-e-i+1}.$$

Remark 5.4.4. *Note that $\mathscr{I}_{\boldsymbol{u}}(i, [\boldsymbol{f}, Q])$ is a polynomial system with*

- $s+n-e-i+1$ *equations, and*
- $s+n-e$ *unknowns.*

Now, for $\boldsymbol{v} \in \mathbb{Q}^{s+n-e-i+1}$, consider the polar variety

$$\mathscr{W}_{\boldsymbol{v}}(\pi_{e,1}, \mathscr{W}_{\boldsymbol{u},i,Q}) \subset \mathbb{C}^{2s+2n-i+1},$$

defied by polynomials

$$\begin{aligned}&\mathscr{I}_{\boldsymbol{v}}(1, \mathscr{I}_{\boldsymbol{u}}(i, [\boldsymbol{f}, Q])) \\ &= \left(\mathscr{I}_{\boldsymbol{u}}(i, [\boldsymbol{f}, Q]), \boldsymbol{Lagrange}(\mathscr{I}_{\boldsymbol{u}}(i, [\boldsymbol{f}, Q]), 1, [L'_1 \cdots L'_{s+n-e-i+1}]), \sum_{i=1}^{s+n-e-i+1} v_i L'_i - 1\right)\end{aligned}$$

in $\mathbb{C}[X_{e+1}, \ldots, X_n, L_1, \ldots, L_s, L_1, \ldots, L_{s+n-e-i+1}]$.

Again, by the discussion given previously regarding the Lagrangian modelling of polar varieties, but where this time we apply Proposition 3.1 from Chapter 3 twice, if $\boldsymbol{u}$ and $\boldsymbol{v}$ are chosen generically, $W(\pi_{e,1}, W(\pi_{e,i}, V([\boldsymbol{f}, Q])))$ is equal to the Zariski closure of the projection on the $X_{e+1}, \ldots, X_n$ variables of the algebraic set defined by the vanishing of equations $\mathscr{I}_{\boldsymbol{v}}(1, \mathscr{I}_{\boldsymbol{u}}(i, [\boldsymbol{f}, Q]))$.

Remark 5.4.5. *Note that $\mathscr{I}_{\boldsymbol{v}}(1, \mathscr{I}_{\boldsymbol{u}}(i, [\boldsymbol{f}, Q]))$ is a polynomial system with*

- $2(s+n-e)-i+1$ *equations, and*
- $2(s+n-e)-i+1$ *unknowns.*

5.4.2 Canny roadmap: degree and canonical height

Please refer to Algorithm 1 throughout this section.

Let $\boldsymbol{f} = (f_1, \ldots, f_s) \in \mathbb{Q}[X_1, \ldots, X_n]^s$ be a sequence of polynomials, with $\deg f_i \leq d, \mathrm{ht}(f_i) \leq h$, and let $Q = Q(X_1, \ldots, X_e)$ and $P = P(X_1, \ldots, X_n)$ be zero-dimensional parameterizations where

- $d_{Q,i}, h_{Q,i}, \hat{h}_{Q,i}$, and
- $d_{P,i}, h_{P,i}, \hat{h}_{P,i}$

are the degrees, heights, and canonical heights, respectively, of $\mathsf{Z}(Q)$ and $\mathsf{Z}(P)$, and where i is the current level of recursion, running from 0 to $n-e-s$.

Here we give degree and canonical height bounds for the roadmap computed in the subroutine $\mathsf{CannyRoadmap}(\boldsymbol{f}, Q)$. In what follows we go through the steps of Algorithm 1, bounding degree and canonical height at each step.

Canny analysis.

- Let $\boldsymbol{A} \in \mathrm{GL}(n, e)$ be a random matrix with entries of height at most $h_{\boldsymbol{A}}$, and assume the inverse of $\boldsymbol{A}$ has entries with height at most $h_{\boldsymbol{A}^{-1}}$.
- Apply the change of coordinates $\boldsymbol{f} = \mathsf{ChangeVariables}(\boldsymbol{A}, \boldsymbol{f})$, and let

$$H = h + nh_{\boldsymbol{A}}.$$

  There are at most n levels of recursion, and therefore multiplying by n gives a bound on the accumulative cost of applying the change of coordinates at each level.

- Apply the change of coordinates $P = \mathsf{ChangeVariables}(\boldsymbol{A}, P)$. The degree will not change, but canonical height will increase.

  Proposition 5.4.6. *Canonical height increases by:*

$$\hat{h}_{P,i} \leftarrow \hat{h}_{P,i} + n(12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i}.$$

  Proof. First, given at most n levels of recursion, multiplying by n is sufficient for the accumulated cost of applying the change of coordinates. Second, consider the following three steps.

  1. By [16, Proposition 2.39(5)],

$$\mathrm{ht}(P) \leq \hat{h}(P) + \frac{7}{2}\log(n+1)d_{P,i}.$$

  2. By [30, Proposition 2.4],

$$\mathrm{ht}(P^{\boldsymbol{A}}) \leq h_{P,i} + (h_{\boldsymbol{A}} + 8\log(2n+1))d_{P,i}.$$

  3. Therefore, by [16, Proposition 2.39(5)],

$$\begin{aligned}\hat{h}(P^{\boldsymbol{A}}) &\leq \frac{7}{2}\log(n+1)d_{P,i} + \mathrm{ht}(P^{\boldsymbol{A}}) \\ &\leq \frac{7}{2}\log(n+1)d_{P,i} + (h_{P,i} + (h_{\boldsymbol{A}} + 8\log(2n+1))d_{P,i}) \\ &\leq \hat{h}_{P,i} + (12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i}.\end{aligned}$$

  □

- Now, $P = \mathsf{Lift}(P, Q)$. Both degree and height will not change:
  - $d_{P,i} \leftarrow d_{P,i}$
  - $\hat{h}_{P,i} \leftarrow \hat{h}_{P,i}$.

Degree will not increase; intersection can only reduce a finite set. From [16, Proposition 2.64] we see canonical height will also not increase.

- Next, $C = \mathsf{Union}(\mathsf{requiredCriticalPoints}(\boldsymbol{f}, Q(X_1, \ldots, X_e), 2), P)$.

1. Let C first be the zero-dimensional parameterization returned by

$$C = \mathsf{requiredCriticalPoints}(\boldsymbol{f}, Q(X_1, \ldots, X_e), 2).$$

C is a zero-dimensional parameterization of

$$W(\pi_{e,1}, W_{2,Q}) \cup \mathrm{sing}(W_{2,Q}).$$

We therefore bound the degree and height of both

- $W(\pi_{e,1}, W_{2,Q})$ and
- $\mathrm{sing}(W_{2,Q})$;

then, the sum of their respective degrees and heights gives the degree and height of C.

(a) First we bound the degree and height of $W(\pi_{e,1}, W_{2,Q})$; we use the Lagrangian formulation of Polar varieties. Consider the Lagrangian polar variety

$$\mathscr{W}_{\boldsymbol{v}}(\pi_{e,1}, \mathscr{W}_{\boldsymbol{u},2,Q}) \subset \mathbb{C}^{2(s+n-e)-1}. \tag{5.1}$$

(Recall the Lagrangian modeling was introduced in Section 2, and reviewed in Section 5.4.1.) After computing the partial derivatives in the Jacobian matrix, the height grows by at most another factor of $\log(d)$. In addition, we make generic choices for $\boldsymbol{u}$ and $\boldsymbol{v}$ and assume their heights are both bound by $h_{\boldsymbol{u}}$. Put

$$H := h + n(h_{\boldsymbol{A}} + h_{\boldsymbol{u}} + \log(d)).$$

The degree of equations defining (5.1) are at most d and height at most H. Recall from Section 5.4.1 that $\mathscr{W}_{\boldsymbol{v}}(\pi_{e,1}, \mathscr{W}_{\boldsymbol{u},2,Q})$ is defined by a polynomial system with

- $2(s+n-e)-1$ equations, and
- $2(s+n-e)-1$ unknowns.

Let

$$N := 2n + 2s - 2e - 1 \leq 4n - 2e - 1 \leq 4n.$$

Remark 5.4.7. *We assume $n \geq 2$.*

By Property (2) from Section 5.2.3, in System (5.1), each equation defines a hypersurface whose class is, entrywise, at most

$$(\log(N+1)d + H)\eta + d\theta.$$

Let $S \subset \mathbb{P}^e_{\mathbb{C}}$ denote the projective closure of $Q(X_1, \ldots, X_e)$ so that $|S| \leq d_{Q,i}$ and canonical height of S is at most $\hat{h}_{Q,i}$. We obtain a cylinder containing the points $Q(X_1, \ldots, X_e)$ by taking the ruled join of S and $\mathbb{P}^{N-e-1}_{\mathbb{C}}$, introducing additional variables to describe the cylinder. The ruled join in the bigger projective space is defined by the same equations defining S but for a larger polynomial ring; it consists in the same equations defining S in addition to new variables defining a cylinder. Consider the class of S :

$$[S]_{\mathbb{Z}} = \hat{h}_{Q,i}\eta\theta^{e-1} + d_{Q,i}\theta^e \in \mathbb{R}[\eta, \theta]/(\eta^2, \theta^{e+1}).$$

The ruled join of S and $\mathbb{P}^{N-e-1}$ is the projectivve algebraic set

$$S\#\mathbb{P}^{N-e-1}_{\mathbb{C}} \subset \mathbb{P}^{e+(N-e-1)+1}_{\mathbb{C}} = \mathbb{P}^N_{\mathbb{C}},$$

which has degree

$$\deg(S\#\mathbb{P}^{N-e-1}_{\mathbb{C}}) = \deg(S) \cdot \deg(\mathbb{P}^{N-e-1}_{\mathbb{C}}) = d_{Q,i} \cdot 1 = d_{Q,i}.$$

Note that $[\mathbb{P}^{N-e-1}_{\mathbb{C}}]_{\mathbb{Z}} = 1$ (see [16, Proposition 1.10 (2)]). Therefore, the polynomial representing

$$[S\#\mathbb{P}^{N-e-1}_{\mathbb{C}}]_{\mathbb{Z}}$$

is equal to the polynomial representing $[S]_{\mathbb{Z}}$, and our cylinder containing $Q(X_1, \ldots, X_e)$ is represented by $[S]_{\mathbb{Z}}$.

Remark 5.4.8. *To apply the intersection theory tools from 5.2.3, we need to homogenize the equations first. Doing so will not increase degree or height.*

Now by Property (3) from Section 5.2.3, the intersection of Q with the equations defining (5.1) give entriwise degree and canonical height bounds as follows:

$$\begin{aligned}
&[S]_{\mathbb{Z}} \cdot ((\log(N+1)d + H)\eta + d\theta)^N \\
&\equiv (\hat{h}_{Q,i}\eta\theta^{e-1} + d_{Q,i}\theta^e) \cdot \left(Nd^{N-1}((\log(N+1)d + H))\eta\theta^{N-1} + d^N\theta^N\right) \\
&\quad \bmod (\eta^2, \theta^{N+1})) \\
&\equiv \left(\hat{h}_{Q,i}d^N + d_{Q,i}Nd^{N-1}((\log(N+1)d + H))\right)\eta\theta^{(N+e)-1} + d_{Q,i}d^N\theta^{(N+e)} \\
&\quad \bmod (\eta^2, \theta^{(N+e)+1}),
\end{aligned}$$

so that

– $d_{C,i} \leq d_{Q,i}d^N$

– $\hat{h}_{C,i} \leq (\hat{h}_{Q,i} d^N + d_{Q,i} N d^N((\log(N+1)d + H))$.

(b) We also bound the height of the singular points $\mathrm{sing}(W_{2,Q})$. First note

$$\mathrm{sing}(W_{2,Q}) \subset \mathrm{sing}(V([\boldsymbol{f}, Q])),$$

and therefore we need only compute $\mathrm{sing}(V([\boldsymbol{f}, Q]))$; they have the following Lagrangian modeling:

$$\boldsymbol{f}, [\mathfrak{L}_1, \ldots, \mathfrak{L}_s] \cdot \mathrm{jac}(\boldsymbol{f}, e), \sum_{i=1}^{s} u_i \mathfrak{L}_i - 1 \tag{5.2}$$

for a randomly chosen vector of integers $\boldsymbol{u} \in \mathbb{Z}^s$ and new Lagrangian multipliers $[\mathfrak{L}_1, \ldots, \mathfrak{L}_s]$. This is an overdetermined system with

– $n - e + s - 1$ equations

– $n - e + s$ unknowns.

To obtain the degree bound, we can apply [26, Proposition 2.3] where E_1 is taken to be the cylinder above Q obtained by

$$Q \times \mathbb{A}^{N-e} \subset \mathbb{A}^N$$

and $E_2, \ldots, E_{N+1}$ correspond to the equations defining (5.2); the degree is bounded by

$$d_{Q,i} d^N.$$

Next, to obtain a bound on the height of the singular points, we apply [30, Corollary 2.10]. To set up the application of this Corollary, let

– Let $V \subset \mathbb{A}^n$ be the affine part of

$$S \# \mathbb{P}^{n-e-1} \subset \mathbb{P}^n,$$

with degree equal to the degree of $S \# \mathbb{P}^{n-e-1}$ which is $d_{Q,i}$, and height (by Proposition 2.39(5) [16]):

$$h_{Q,i} \leq \hat{h}_{Q,i} + \frac{7}{2} \log(n+1) d_{Q,i}.$$

– Set $V' = V \times \mathbb{A}^s \subset \mathbb{A}^{n+s}$. The degree and height of V' remain the same as the degree and height of V.

It then follows by [30, Corollary 2.10] that

$$\begin{aligned}
&\mathrm{ht}(V' \cap V(\boldsymbol{f}, [\mathfrak{L}_1, \ldots, \mathfrak{L}_s] \cdot \mathrm{jac}(\boldsymbol{f}), \sum_{i=1}^{s} u_i \mathfrak{L}_i - 1) \\
\leq & d^N \left((\hat{h}_{Q,i} + \frac{7}{2} \log(n+1) d_{Q,i}) + N H d_{Q,i} + N \log(N+1) d_{Q,i} \right) \\
\leq & (\hat{h}_{Q,i} d^N + d_{Q,i} N d^N((\log(N+1)d + H)).
\end{aligned}$$

Therefore after adding both respective degrees and heights we obtain:

- $d_{C,i} \leq 2d_{Q,i}d^N$
- $\hat{h}_{C,i} \leq 2(\hat{h}_{Q,i}d^N + d_{Q,i}Nd^{N-1}((\log(N+1)d + H))$.

2. Now we take the union $C = \mathsf{Union}(C, P)$:

- $d_{C,i} \leq 2d_{Q,i}d^N + d_{P,i}$
- $\hat{h}_{C,i} \leq 2\left(\hat{h}_{Q,i}d^N + d_{Q,i}Nd^{N-1}(\log(N+1)d + H)\right)$
  $+ (\hat{h}_{P,i} + n(12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i})$.

- We obtain $Q' = \mathsf{Projection}(C, [X_1, \ldots, X_{e+1}])$:
  - $d_{Q',i} \leq d_{C,i} \leq 2d_{Q,i}d^N + d_{P,i}$
  - $\hat{h}_{Q',i} \leq \hat{h}_{C,i}$.

- We next obtain $P' = \mathsf{Union}(\mathsf{LiftW}(\boldsymbol{f}, Q'), P)$ where

1. First, $L = \mathsf{LiftW}(\boldsymbol{f}, Q')$ is a parameterization containing

$$W_{2,Q} \cap \pi^{-1}_{e+2-1}(\mathsf{Z}(Q')) \subset \mathsf{Z}(R).$$

Again, we use the Lagrangian modeling to obtain degree and height bounds. Consider the Lagrangian polar variety

$$\mathscr{W}_{\boldsymbol{u},2,Q} \subset \mathbb{C}^{n-e+s}. \tag{5.3}$$

Recall there are

$$M := n + s - e + 1 \leq 2n$$

equations defining the Lagrange polar variety (5.3).
Let S' denote the projective closure of Q'. We obtain a cylinder containing the points Q' by taking the ruled join of S' and $\mathbb{P}^{M-e-1}_{\mathbb{C}}$. Consider the class of S' :

$$[S']_{\mathbb{Z}} = \hat{h}_{Q',i}\eta\theta^{e-1} + d_{Q',i}\theta^e \in \mathbb{R}[\eta, \theta]/(\eta^2, \theta^{e+1}).$$

The ruled join of S' and $\mathbb{P}^{M-e-1}$ is the projective variety

$$S' \# \mathbb{P}^{M-e-1}_{\mathbb{C}} \subset \mathbb{P}^{e+(M-e-1)+1}_{\mathbb{C}} = \mathbb{P}^M_{\mathbb{C}},$$

which has degree

$$\deg(S' \# \mathbb{P}^{M-e-1}_{\mathbb{C}}) = \deg(S') \cdot \deg(\mathbb{P}^{M-e-1}_{\mathbb{C}}) = d_{Q',i} \cdot 1 = d_{Q',i}.$$

By Property (2) from 5.2.3, in System (5.3), each equation defines a hypersurface whose class is, entrywise, at most

$$(\log(M+1)d + H)\eta + d\theta.$$

By Property (3) from 5.2.3, the intersection of Q' with the equations defining (5.3) obtains entriwise height bounds as follows.

$$\begin{aligned}
&[S']_{\mathbb{Z}} \cdot ((\log(M+1)d + H)\eta + d\theta)^M \\
&\equiv (\hat{h}_{Q',i}\eta\theta^{e-1} + d_{Q',i}\theta^e) \cdot \left(Md^{M-1}((\log(M+1)d+H))\eta\theta^{M-1} + d^M\theta^M\right) \\
&\quad \bmod (\eta^2, \theta^{M+1})) \\
&\equiv \left(\hat{h}_{Q',i}d^M + d_{Q',i}Md^{M-1}(\log(M+1)d+H)\right)\eta\theta^{(M+e)-1} + d_{Q',i}d^M\theta^{(M+e)} \\
&\quad \bmod (\eta^2, \theta^{(M+e)+1}),
\end{aligned}$$

and therefore

- $d_{L,i} \leq d_{Q',i}d^M$
- $\hat{h}_{L,i} \leq \left(\hat{h}_{Q',i}d^M + d_{Q',i}Md^{M-1}(\log(M+1)d+H)\right)$.

2. Now $P' = \mathsf{Union}(\mathsf{LiftW}(\boldsymbol{f}^{\boldsymbol{A}}, Q'), P)$:
   - $d_{P',i} \leq d_{L,i} + d_{P,i} \leq d_{Q',i}d^M + d_{P,i}$
   - $\hat{h}_{P',i} \leq \hat{h}_{L,i} + (\hat{h}_{P,i} + (12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i}) \leq$

     $\left(\hat{h}_{Q',i}d^M + d_{Q',i}Md^{M-1}(\log(M+1)d+H)\right) + (\hat{h}_{P,i} + n(12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i})$.

- We now enter the recursive step with $R = \mathsf{CannyRoadmap}(\boldsymbol{f}, Q', P')$ and obtain the following recurrences:
  - $d_{Q,i+1} = d_{Q',i} \leq 2d_{Q,i}d^{4n} + d_{P,i}$
  - $\hat{h}_{Q,i+1} = \hat{h}_{Q',i} \leq \hat{h}_{C,i} \leq$

    $2(\hat{h}_{Q,i}d^{4n} + d_{Q,i}(4n)d^{4n-1}(\log(4n+1)d) + H)) + (\hat{h}_{P,i} + n(12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i})$
  - $d_{P,i+1} = d_{P',i} \leq d_{Q',i}d^{2n} + d_{P,i} \leq (2d_{Q,i}d^{4n} + d_{P,i})d^{2n} + d_{P,i}$

    $\leq 2d_{Q,i}d^{4n+2n} + d_{P,i}d^{2n} + d_{P,i}$
  - $\hat{h}_{P,i+1} = \hat{h}_{P',i} \leq \hat{h}_{L,i} + (\hat{h}_{P,i} + n(12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i})$

    $\leq \left(\hat{h}_{Q',i}d^{2n} + d_{Q',i}(2n)d^{2n-1}(\log(2n+1)d+H)\right) + (\hat{h}_{P,i} + n(12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i})$.

To solve the recurrence for the degree, assume $d_i \geq \max\{d_{Q,i}, d_{P,i}\}$, and then notice

$$\begin{aligned}
d_{Q,i+1} &\leq d_{P,i+1} \\
&\leq 2d_{Q,i}d^{6n} + d_{P,i}d^{2n} + d_{P,i} \\
&\leq d_i(2d^{6n} + d^{2n} + 1) \\
&= d_iK
\end{aligned}$$

where we define $K = (2d^{6n} + d^{2n} + 1)$. Now, setting

$$d_0 = \max\{d_{Q,0}, d_{P,0}\}$$

and

$$d_{i+1} = d_i K,$$

it follows that

- $d_{Q,i+1} \leq d_{i+1}$
- $d_{P,i+1} \leq d_{i+1}$,

and therefore

$$d_i = d_0 K^i.$$

Now, to solve the recurrence for the height, assume $\hat{h}_i \geq \max\{\hat{h}_{Q,i}, \hat{h}_{P,i}\}$, define

- $K_1 = (\log(4n+1)d + H)(4n)$
- $K_2 = n(12\log(2n+1) + h_{\boldsymbol{A}})$
- $K_3 = 2(2K_1 + K_2)$,

and notice

- $\hat{h}_{Q,i+1} = \hat{h}_{Q',i} \leq \hat{h}_{C,i} \leq$
  $2(\hat{h}_{Q,i}d^{4n} + d_{Q,i}(4n)d^{4n-1}(\log(4n+1)d + H)) + (\hat{h}_{P,i} + n(12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i})$
  $\leq 2\hat{h}_{Q,i}d^{4n} + d_0K^i(2K_1 + K_2)d^{4n} + \hat{h}_{P,i}$
- $\hat{h}_{P,i+1} = \hat{h}_{P',i} \leq \hat{h}_{L,i} + (\hat{h}_{P,i} + (12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i})$
  $\leq \left(\hat{h}_{Q',i}d^{2n} + d_{Q',i}(2n)d^{2n}(\log(2n+1)d + H)\right) + (\hat{h}_{P,i} + n(12\log(2n+1) + h_{\boldsymbol{A}})d_{P,i})$
  $\leq \hat{h}_{Q',i}d^{2n} + d_{Q',i}d^{2n}K_1 + \hat{h}_{P,i} + K_2 d_{P,i}$
  $\leq (2\hat{h}_{Q,i}d^{4n} + d_0K^i(2K_1 + K_2)d^{4n} + \hat{h}_{P,i})d^{2n} + (d_0K^{i+1})d^{2n}K_1 + \hat{h}_{P,i} + K_2 d_{P,i}$
  $\leq 2\hat{h}_{Q,i}d^{6n} + d_0K^i(2K_1 + K_2)d^{6n} + \hat{h}_{P,i}d^{2n} + (d_0K^{i+1})d^{2n}K_1 + \hat{h}_{P,i} + K_2(d_0K^i)$
  $\leq (2\hat{h}_{Q,i}d^{6n} + \hat{h}_{P,i}d^{2n} + \hat{h}_{P,i})$
  $+ d_0K^i(2K_1 + K_2)d^{6n} + (d_0K^{i+1})d^{2n}K_1 + K_2(d_0K^i)$
  $\leq (2\hat{h}_{Q,i}d^{6n} + \hat{h}_{P,i}d^{2n} + \hat{h}_{P,i})$
  $+ d^{6n}(d_0K^i(2K_1 + K_2) + (d_0K^{i+1})K_1 + K_2(d_0K^i))$
  $\leq (2\hat{h}_{Q,i}d^{6n} + \hat{h}_{P,i}d^{2n} + \hat{h}_{P,i})$
  $+ d^{6n}(d_0K^{i+1}(2K_1 + K_2) + (d_0K^{i+1})K_1 + K_2(d_0K^{i+1}))$
  $\leq \hat{h}_i K + d^{6n}(d_0K^{i+1})(4K_1 + 2K_2)$
  $\leq \hat{h}_i K + (d_0K^{i+2})(4K_1 + 2K_2)$
  $= \hat{h}_i K + (d_0K^{i+2})K_3.$

Setting

$$\hat{h}_0 = \max\{\hat{h}_{Q,0}, \hat{h}_{P,0}\}$$

and

$$\hat{h}_{i+1} := \hat{h}_i K + d_0 K^{i+2} K_3 = K(\hat{h}_i + d_0 K^{i+1} K_3),$$

it follows that

- $\hat{h}_{Q,i+1} \leq \hat{h}_{i+1}$
- $\hat{h}_{P,i+1} \leq \hat{h}_{i+1}$

and

$$\hat{h}_i = K^{i+1} d_0 K_3 i + h_0 K^i.$$

- Now we compute $R' = \mathsf{CriticalCurve}(\boldsymbol{f}, Q)$. Then, by Property (3) in Section 5.2.3, the intersection of Q with the equations defining (5.3) obtain entriwise degree and canonical height bounds as follows.

$$\begin{aligned}
&[S]_{\mathbb{Z}} \cdot ((\log(M+1)d + H)\eta + d\theta)^M \\
&\equiv \left(\hat{h}_{Q,i} d^M + d_{Q,i} M d^{M-1}(\log(M+1)d + H)\right) \eta\theta^{(M+e)-1} + d_{Q,i} d^M \theta^{(M+i)} \\
&\quad \bmod (\eta^2, \theta^{(M+e)+1}),
\end{aligned}$$

and therefore

- $d_{R',i} \leq d_{Q,i} d^M \leq d_i d^{2n} = d_0 K^i d^{2n} \leq d_0 K^{i+1}$
- $\hat{h}_{R',i} \leq$

$$\begin{aligned}
&\hat{h}_{Q,i} d^M + d_{Q,i} M d^{M-1}(\log(M+1)d + H) \\
\leq &\hat{h}_i d^M + d_i d^{M-1} K_1 \\
\leq &(K^{i+1} d_0 K_3 i + \hat{h}_0 K^i) d^{2n} + (d_0 K^i) K_1 d^{2n} \\
\leq &K^{i+2} d_0 K_3 i + \hat{h}_0 K^{i+1} + d_0 K^{i+1} K_1 d^{2n} \\
= &K^{i+2} d_0 K_3 i + K^{i+1}(\hat{h}_0 + d_0 K_1 d^{2n}).
\end{aligned}$$

- Now $R'' = \mathsf{Union}(R, R')$, and the degree and height of R'' increase by at most a factor of the depth i:

- $d_{R'',i} \leq i d_{R',i} \leq$

$$i d_0 K^{i+1}$$

- $\hat{h}_{R'',i} \leq i \hat{h}_{R',i} \leq$

$$i\left(K^{i+2} d_0 K_3 i + K^{i+1}(\hat{h}_0 + d_0 K_1 d^{2n})\right).$$

- In the last step we invert the change of coordinates: ChangeVariables($\boldsymbol{A}^{-1}, R''$). Degree does not change; height increases:
  - $d_{R'',i} \leq d_{R'',i}$
  - $\hat{h}_{R'',i} \leftarrow \hat{h}_{R'',i} + (15\log(2n+1) + h_{\boldsymbol{A}^{-1}})d_{R'',i}$.

  The proof for the bound on the height follows the same logic as Proposition 5.4.6. We define
  $$K_4 = (15\log(2n+1) + h_{\boldsymbol{A}^{-1}}).$$

Noting that at step 1 of CannyRoadmap (Algorithm 1), the algorithm halts when $n - s - e = 1$, we have proven the following proposition.

Proposition 5.4.9. *Given a sequence of polynomials $\boldsymbol{f} = (f_1, \ldots, f_s) \in \mathbb{Q}[X_1, \ldots, X_n]^s$ of degree d and height h, and zero-dimensional parameterizations $Q = Q(X_1, \ldots, X_e)$ and $P = P(X_1, \ldots, X_n)$ with degree and canonical height at most d_0 and $\hat{h}_0$, the output of* CannyRoadmap *on input $(\boldsymbol{f}, Q, P)$ is a roadmap R with degree and canonical height at most*

- $d_R \leq nd_0K^{(n-s-e)+1}$
- $\hat{h}_R \leq n(K^{n-s-e+2}d_0K_3n + K^{n-s-e+1}(\hat{h}_0 + d_0K_1d^{2n}) + K_4d_0K^{n-s-e+1})$

where

- $K = 2d^{6n} + d^{2n} + 1$
- $K_1 = (\log(4n+1)d + H)(4n)$
- $K_2 = n(12\log(2n+1) + h_{\boldsymbol{A}})$
- $K_3 = 2(2K_1 + K_2)$
- $K_4 = (15\log(2n+1) + h_{\boldsymbol{A}^{-1}})$
- $H = h + n(h_{\boldsymbol{A}} + h_{\boldsymbol{v}} + \log(d))$.

Remark 5.4.10. *When* CannyRoadmap *is called by* MainRoadmap *then the depth is equal to*
$$n - s - e = n - (n - e - \sqrt{n}) - e = \sqrt{n}.$$

5.4.3 Main roadmap: degree and canonical height

Please refer to Algorithm 2 throughout this section.

Let $f \in \mathbb{Q}[X_1, \ldots, X_n]$ be a squarefree polynomial with $\deg f \leq d, \mathrm{ht}(f) \leq h$, and let $Q = Q(X_1, \ldots, X_e)$ and $P = P(X_1, \ldots, X_n)$ be zero-dimensional parameterizations where

- $\mathfrak{d}_{Q,j}, \mathfrak{h}_{Q,j}, \hat{\mathfrak{h}}_{Q,j}$, and
- $\mathfrak{d}_{P,j}, \mathfrak{h}_{P,j}, \hat{\mathfrak{h}}_{P,j}$

are the degrees, heights, and canonical heights, respectively, of $\mathsf{Z}(Q)$ and $\mathsf{Z}(P)$, and where j is the current level of recursion, running from 0 to $\sqrt{n}$. As j increases by 1 the number of steps increases by $\lceil\sqrt{n}\rceil$. Therefore after roughly $\sqrt{n}$ steps, $\sqrt{n} \cdot \sqrt{n} = n$ is achieved.

Here we give degree and canonical height bounds for the roadmap computed in the subroutine $\mathsf{MainRoadmap}(f, Q)$. In what follows we go through the steps of Algorithm 2, bounding degree and canonical height at each step.

The analysis is very similar to the one for Canny. To avoid unnecessary repetition, some of the claims will be stated without justification.

Main analysis.

- Let $\boldsymbol{A} \in \mathrm{GL}(n, e)$ be a random matrix with the same height bounds $h_{\boldsymbol{A}}$, and $h_{\boldsymbol{A}^{-1}}$.
- $f = \mathsf{ChangeVariables}(\boldsymbol{A}, f)$, $H = h + nh_{\boldsymbol{A}}$.
- $P = \mathsf{ChangeVariables}(\boldsymbol{A}, P)$:
  - $\mathfrak{d}_{P,j} \leftarrow \mathfrak{d}_{P,j}$
  - $\hat{\mathfrak{h}}_{P,j} \leftarrow \hat{\mathfrak{h}}_{P,j} + (12\log(2n+1) + h_{\boldsymbol{A}})\mathfrak{d}_{P,j}$.
- $P = \mathsf{Lift}(P, Q)$:
  - $\mathfrak{d}_{P,j} \leftarrow \mathfrak{d}_{P,j}$
  - $\hat{\mathfrak{h}}_{P,j} \leftarrow \hat{\mathfrak{h}}_{P,j}$.
- $C = \mathsf{Union}(\mathsf{requiredCriticalPoints}(f, Q(X_1, \ldots, X_e), \lceil\sqrt{n}\rceil), P)$.

1. First, $C = \mathsf{requiredCriticalPoints}(f, Q(X_1, \ldots, X_e), \lceil\sqrt{n}\rceil)$, where C is a zero-dimensional parameterization of

$$W(\pi_{e,1}, W_{\lceil\sqrt{n}\rceil,Q}) \cup \mathrm{sing}(W_{\lceil\sqrt{n}\rceil,Q}).$$

We therefore bound the degree and height of both

– $W(\pi_{e,1}, W_{\lceil\sqrt{n}\rceil,Q})$ and
– $\mathrm{sing}(W_{\lceil\sqrt{n}\rceil,Q})$;

then, the sum of their respective degrees and heights gives the degree and height of C.

(a) First we bound the degree and height of $W(\pi_{e,1}, W_{\lceil\sqrt{n}\rceil,Q})$. We again use the Lagrangian formulation.

$$\mathscr{W}_{\boldsymbol{v}}(\pi_{e,1}, \mathscr{W}_{\boldsymbol{u},\lceil\sqrt{n}\rceil,Q}) \subset \mathbb{C}^{2(1+n-e)-\lceil\sqrt{n}\rceil+1}. \tag{5.4}$$

As with Canny, take

$$H = h + n(h_{\boldsymbol{A}} + h_{\boldsymbol{u}} + \log(d)).$$

– The degree of equations defining (5.4) are at most d; and
– the height of equations defining (5.4) are at most H.

Recall from Section 5.4.1 that $\mathscr{W}_{\boldsymbol{v}}(\pi_{e,1}, \mathscr{W}_{\boldsymbol{u},\lceil\sqrt{n}\rceil,Q})$ is defined by a polynomial system with

– $2(1+n-e) - \lceil\sqrt{n}\rceil + 1$ equations, and
– $2(1+n-e) - \lceil\sqrt{n}\rceil + 1$ unknowns.

Let

$$\mathscr{N} := 2n + 2 - 2e - \lceil\sqrt{n}\rceil + 1 \leq 2n + 2 - 2e \leq 4n$$

In System (5.4), by Property (2) in Section 5.2.3, each equation defines a hypersurface whose class is, entrywise, at most

$$(\log(\mathscr{N}+1)d + H)\eta + d\theta.$$

As with Canny, we obtain a cylinder containing the points $Q(X_1, \ldots, X_e)$ by taking the ruled join of S and $\mathbb{P}_{\mathbb{C}}^{\mathscr{N}-e-1}$, introducing additional variables to describe the cylinder. The ruled join of $[S]_{\mathbb{Z}}$ and $[\mathbb{P}^{\mathscr{N}-e-1}]_{\mathbb{Z}}$ is the projective variety

$$S\#\mathbb{P}_{\mathbb{C}}^{\mathscr{N}-e-1} \subset \mathbb{P}_{\mathbb{C}}^{e+(\mathscr{N}-e-1)+1} = \mathbb{P}_{\mathbb{C}}^{\mathscr{N}},$$

which has degree

$$\deg(S\#\mathbb{P}_{\mathbb{C}}^{\mathscr{N}-e-1}) = \deg(S)\cdot\deg(\mathbb{P}_{\mathbb{C}}^{n-e-1}) = \mathfrak{d}_{Q,j}\cdot 1 = \mathfrak{d}_{Q,j}.$$

Consider the class of S :

$$[S]_{\mathbb{Z}} = \hat{\mathfrak{h}}_{Q,j}\eta\theta^{e-1} + \mathfrak{d}_{Q,j}\theta^e \in \mathbb{R}[\eta,\theta]/(\eta^2, \theta^{e+1}).$$

The polynomial representing

$$[S\#\mathbb{P}_{\mathbb{C}}^{\mathscr{N}-e-1}]_{\mathbb{Z}}$$

is equal to the polynomial representing $[S]_{\mathbb{Z}}$, so that our cylinder containing $Q(X_1, \ldots, X_e)$ is represented by $[S]_{\mathbb{Z}}$.
Now by Property (3) from Section 5.2.3, the intersection of Q with the equations defining (5.4) obtain

$$\begin{aligned}
&[S]_{\mathbb{Z}} \cdot ((\log(\mathscr{N}+1)d + H)\eta + d\theta)^{\mathscr{N}} \\
&\equiv (\hat{\mathfrak{h}}_{Q,j}\eta\theta^{e-1} + \mathfrak{d}_{Q,j}\theta^e) \cdot \left(\mathscr{N} d^{\mathscr{N}-1}((\log(\mathscr{N}+1)d + H))\eta\theta^{\mathscr{N}-1} + d^{\mathscr{N}}\theta^{\mathscr{N}}\right) \\
&\quad \bmod (\eta^2, \theta^{\mathscr{N}+1})) \\
&\equiv \left(\hat{\mathfrak{h}}_{Q,j} d^{\mathscr{N}} + \mathfrak{d}_{Q,j}\mathscr{N} d^{\mathscr{N}-1}((\log(\mathscr{N}+1)d + H))\right)\eta\theta^{(\mathscr{N}+e)-1} + \mathfrak{d}_{Q,j} d^{\mathscr{N}}\theta^{(\mathscr{N}+e)} \\
&\quad \bmod (\eta^2, \theta^{(\mathscr{N}+e)+1}).
\end{aligned}$$

(b) The singular points have the following Lagrangian formulation:

$$f, \mathfrak{L} \cdot \nabla_{(X_{e+1},\ldots,X_n)} f, u\mathfrak{L} - 1 \tag{5.5}$$

with $\mathfrak{L}$ a Lagrange variable and u a non-zero integer.
The degree and canonical height of the singular points are obtained in the same way as was done in the Canny analysis. After adding with the degree and canonical height of the critical points we obtain:

- $\mathfrak{d}_{C,j} \leq 2\mathfrak{d}_{Q,j} d^{\mathscr{N}}$
- $\hat{\mathfrak{h}}_{C,j} \leq 2(\hat{\mathfrak{h}}_{Q,j} d^{\mathscr{N}} + \mathfrak{d}_{Q,j}\mathscr{N} d^{\mathscr{N}-1}((\log(\mathscr{N}+1)d + H))$.

2. Now, $C = \mathsf{Union}(C, P)$:
   - $\mathfrak{d}_{C,j} \leq 2\mathfrak{d}_{Q,j} d^{\mathscr{N}} + \mathfrak{d}_{P,j}$
   - $\hat{\mathfrak{h}}_{C,j} \leq 2\left(\hat{\mathfrak{h}}_{Q,j} d^{\mathscr{N}} + \mathfrak{d}_{Q,j}\mathscr{N} d^{\mathscr{N}-1}(\log(\mathscr{N}+1)d + H)\right)$
     $+ (\hat{\mathfrak{h}}_{P,j} + n(12\log(2n+1) + h_{\boldsymbol{A}})\mathfrak{d}_{P,j})$

- $Q' = \mathsf{Projection}(C, [X_1, \ldots, X_{e+\lceil\sqrt{n}\rceil+1}])$:
  - $\mathfrak{d}_{Q',j} \leq \mathfrak{d}_{C,j} \leq 2\mathfrak{d}_{Q,j} d^{\mathscr{N}} + \mathfrak{d}_{P,j}$
  - $\hat{\mathfrak{h}}_{Q',j} \leq \hat{\mathfrak{h}}_{C,j}$.

- $P' = \mathsf{Union}(\mathsf{LiftW}(f^{\boldsymbol{A}}, Q'), P)$:

1. First, $L = \mathsf{LiftW}(f^{\boldsymbol{A}}, Q')$ is a parameterization containing

$$W_{\lceil\sqrt{n}\rceil,Q} \cap \pi^{-1}_{e+\lceil\sqrt{n}\rceil-1}(Z(Q')) \subset \mathsf{Z}(R).$$

Recall that there are

$$\mathscr{M} := n + 1 - e + 1 \leq 2n$$

equations defining

$$\mathscr{W}_{\boldsymbol{u},\lceil\sqrt{n}\rceil,Q} \subset \mathbb{C}^{n-e+1}. \tag{5.6}$$

Let S' denote the projective closure of Q'. We obtain a cylinder containing the points Q' by taking the ruled join of S' and $\mathbb{P}_{\mathbb{C}}^{\mathscr{M}-e-1}$. Consider the class of S' :

$$[S']_{\mathbb{Z}} = \hat{\mathfrak{h}}_{Q',e}\eta\theta^{e-1} + \mathfrak{d}_{Q',e}\theta^{e} \in \mathbb{R}[\eta,\theta]/(\eta^2,\theta^{e+1}).$$

The ruled join of $[S']_{\mathbb{Z}}$ and $[\mathbb{P}^{\mathscr{M}-e-1}]_{\mathbb{Z}}$ is the projective variety

$$S'\#\mathbb{P}_{\mathbb{C}}^{\mathscr{M}-e-1} \subset \mathbb{P}_{\mathbb{C}}^{e+(\mathscr{M}-e-1)+1} = \mathbb{P}_{\mathbb{C}}^{\mathscr{M}},$$

which has degree

$$\deg(S'\#\mathbb{P}_{\mathbb{C}}^{\mathscr{M}-e-1}) = \deg(S')\cdot\deg(\mathbb{P}_{\mathbb{C}}^{\mathscr{M}-e-1}) = \mathfrak{d}_{Q',e}\cdot 1 = \mathfrak{d}_{Q',e}.$$

By Property (2), Section 5.2.3, the equations in System (5.6) each define a hypersurface whose class is, entrywise, at most

$$((\log(\mathscr{M}+1)d+H)\eta + d\theta.$$

By Property (3), Section 5.2.3, the intersection of Q' with the equations in (5.6) obtains

$$\begin{aligned}
&[S']_{\mathbb{Z}}\cdot((\log(\mathscr{M}+1)d+H)\eta+d\theta)^{\mathscr{M}}\\
&\equiv(\hat{\mathfrak{h}}_{Q',j}\eta\theta^{e-1}+\mathfrak{d}_{Q',j}\theta^{e})\cdot\left(\mathscr{M}d^{\mathscr{M}-1}((\log(\mathscr{M}+1)d+H))\eta\theta^{\mathscr{M}-1}+d^{\mathscr{M}}\theta^{\mathscr{M}}\right)\\
&\quad \bmod (\eta^2,\theta^{\mathscr{M}+1}))\\
&\equiv\left(\hat{\mathfrak{h}}_{Q',j}d^{\mathscr{M}}+\mathfrak{d}_{Q',j}\mathscr{M}d^{\mathscr{M}-1}(\log(\mathscr{M}+1)d+H)\right)\eta\theta^{(\mathscr{M}+e)-1}+\mathfrak{d}_{Q',j}d^{\mathscr{M}}\theta^{(\mathscr{M}+e)}\\
&\quad \bmod (\eta^2,\theta^{(\mathscr{M}+e)+1}).
\end{aligned}$$

- $\mathfrak{d}_{L,j} \le \mathfrak{d}_{Q',j}d^{\mathscr{M}}$
- $\hat{\mathfrak{h}}_{L,j} \le \left(\hat{\mathfrak{h}}_{Q',j}d^{\mathscr{M}}+\mathfrak{d}_{Q',j}\mathscr{M}d^{\mathscr{M}-1}(\log(\mathscr{M}+1)d+H)\right)$

2. Now $P' = \mathsf{Union}(L,P)$:
   - $\mathfrak{d}_{P',j} \le \mathfrak{d}_{Q',j}d^{\mathscr{M}} + \mathfrak{d}_{P,j}$
   - $\hat{\mathfrak{h}}_{P',j} \le \left(\hat{\mathfrak{h}}_{Q',j}d^{\mathscr{M}}+\mathfrak{d}_{Q',j}\mathscr{M}d^{\mathscr{M}-1}(\log(\mathscr{M}+1)d+H)\right)+(\hat{\mathfrak{h}}_{P,j}+n(12\log(2n+1)+h_{\boldsymbol{A}})\mathfrak{d}_{P,j})$.

- We obtain the same recurrences as we did with the Canny analysis.

$$R = \mathsf{MainRoadmap}(f,Q',P',\lceil\sqrt{n}\rceil)\ :$$

- $\mathfrak{d}_{Q,j+1} = \mathfrak{d}_{Q',j} \le 2\mathfrak{d}_{Q,j}d^{4n} + \mathfrak{d}_{P,j}$
- $\hat{\mathfrak{h}}_{Q,j+1} = \hat{\mathfrak{h}}_{Q',j} \le \hat{\mathfrak{h}}_{C,j}$

$$2(\hat{\mathfrak{h}}_{Q,j}d^{4n}+\mathfrak{d}_{Q,j}(4n)d^{4n-1}(\log(4n+1)d+H))+(\hat{\mathfrak{h}}_{P,j}+(12\log(2n+1)+h_{\boldsymbol{A}})\mathfrak{d}_{P,j})$$

- $\mathfrak{d}_{P,j+1} = \mathfrak{d}_{P',j} \leq \mathfrak{d}_{Q',j} d^{2n} + \mathfrak{d}_{P,j} \leq (2\mathfrak{d}_{Q,j} d^{4n} + \mathfrak{d}_{P,j}) d^{2n} + \mathfrak{d}_{P,j}$
$\leq 2\mathfrak{d}_{Q,j} d^{4n+2n} + \mathfrak{d}_{P,j} d^{2n} + \mathfrak{d}_{P,j}.$
- $\hat{\mathfrak{h}}_{P,j+1} = \hat{\mathfrak{h}}_{P',j} \leq \hat{\mathfrak{h}}_{L,j} + (\hat{\mathfrak{h}}_{P,j} + (12\log(2n+1) + h_{\boldsymbol{A}})\mathfrak{d}_{P,j})$
$\leq 2\left(\hat{\mathfrak{h}}_{Q',j} d^{2n} + \mathfrak{d}_{Q',j}(2n) d^{2n-1}(\log(2n+1)d + H)\right) + (\hat{\mathfrak{h}}_{P,j} + (12\log(2n+1) + h_{\boldsymbol{A}})\mathfrak{d}_{P,j}).$

Let

- $\mathfrak{K} = (2d^{6n} + d^{2n} + 1)$
- $\mathfrak{K}_1 = (\log(4n+1)d + H)(4n)$
- $\mathfrak{K}_2 = n(12\log(2n+1) + h_{\boldsymbol{A}})$
- $\mathfrak{K}_3 = 2(2\mathfrak{K}_1 + \mathfrak{K}_2)$.

Remark 5.4.11. *The values of $\mathfrak{K}, \mathfrak{K}_1, \mathfrak{K}_2, \mathfrak{K}_3$ are the same as K, K_1, K_2, K_3 from the Canny analysis. However, to separate the two analyses, we choose to use separate notation.*

Then, with

- $\mathfrak{d}_0 = \max\{\mathfrak{d}_{Q,0}, \mathfrak{d}_{P,0}, 1\}$
- $\hat{\mathfrak{h}}_0 = \max\{\hat{\mathfrak{h}}_{Q,0}, \hat{\mathfrak{h}}_{P,0}, 1\}$,

we obtain

- $\mathfrak{d}_j = \mathfrak{d}_0 \mathfrak{K}^j$
- $\hat{\mathfrak{h}}_j = \mathfrak{K}^{j+1}\mathfrak{d}_0\mathfrak{K}_3 j + \mathfrak{h}_0\mathfrak{K}^j$.

- Next, $R' = \mathsf{CannyRoadmap}(\boldsymbol{f}, Q, P')$, where
  1. $\boldsymbol{f} = [f, \Delta]$
  2. $\Delta = [\partial f/\partial X_i \mid i \in [e+i+1, \ldots, n]]$.

  Entries in $\boldsymbol{f}$ have
  - degree at most d
  - height at most H.

  We call Proposition 5.4.9 with
  - $s = n - e - \lceil\sqrt{n}\rceil$
  - $d_0 = \max\{\mathfrak{d}_{Q,j}\mathfrak{d}_{P',j}, 1\} \leq \mathfrak{d}_0\mathfrak{K}^{j+1}$

– $\hat{h}_0 = \max\{\hat{\mathfrak{h}}_{Q,j}\hat{\mathfrak{h}}_{P',j}, 1\} \leq \mathfrak{K}^{j+2}\mathfrak{K}_4(j+1) + \hat{\mathfrak{h}}_0\mathfrak{K}^{j+1}$.

We obtain

1. $\mathfrak{d}_{R'} \in$
$$O(n\mathfrak{d}_0 d^{24n^{3/2}})$$
2. $\hat{\mathfrak{h}}_{R'} \in$
$$O\left((\mathfrak{d}_0\mathfrak{K}_4 + \hat{\mathfrak{h}}_0)nd^{24n^{3/2}} + (\mathfrak{K}_4\sqrt{n} + \mathfrak{d}_0(\mathfrak{K}_1 + n\mathfrak{K}_3))nd^{30n^{3/2}}\right)$$
   – $\mathfrak{K} = 2d^{6n} + d^{2n} + 1$
   – $\mathfrak{K}_1 = (\log(4n+1)d + H)(4n)$
   – $\mathfrak{K}_2 = n(12\log(2n+1) + h_{\boldsymbol{A}})$
   – $\mathfrak{K}_3 = 2(2\mathfrak{K}_1 + \mathfrak{K}_2)$
   – $\mathfrak{K}_4 = (15\log(2n+1) + h_{\boldsymbol{A}^{-1}})$.
   – $H = h + n(h_{\boldsymbol{A}} + h_{\boldsymbol{u}} + h_{\boldsymbol{v}} + \log(d))$.

Proof. 1. Immediately from Proposition 5.4.9 we obtain
$$\mathfrak{d}_{R'} \leq n\mathfrak{d}_0\mathfrak{K}^{j+1}\mathfrak{K}^{n-s-e+1}.$$
Now our index j is at most $\sqrt{n}$:
$$\mathfrak{d}_{R'} \leq n\mathfrak{d}_0\mathfrak{K}^{\sqrt{n}+1}\mathfrak{K}^{n-s-e+1}.$$
Substitute $\mathfrak{K} = (2d^{6n} + d^{2n} + 1) \leq 4d^{6n}$ gives
$$\mathfrak{d}_{R'} \leq n\mathfrak{d}_0(4d^{6n})^{\sqrt{n}+1}(4d^{6n})^{n-s-e+1}.$$
Given that $s = n - e - \sqrt{n}$:
$$\mathfrak{d}_{R'} \in O(n\mathfrak{d}_0(d^{6n})^{\sqrt{n}+1}(d^{6n})^{\sqrt{n}+1}).$$
Therefore
$$\mathfrak{d}_{R'} \in O(n\mathfrak{d}_0 d^{12n^{3/2}+12n}) \in O(n\mathfrak{d}_0 d^{24n^{3/2}}).$$

2. Immediately from Proposition 5.4.9, using
$$n - s - e = n - (n - e - \sqrt{n}) - e = \sqrt{n},$$
we obtain
$$n(\mathfrak{K}^{\sqrt{n}+2}d_0\mathfrak{K}_3 n + \mathfrak{K}^{\sqrt{n}+1}(\hat{h}_0 + d_0\mathfrak{K}_1 d^{2n}) + \mathfrak{K}_4 d_0\mathfrak{K}^{\sqrt{n}+1})$$
Substitute $d_0 \leq \mathfrak{d}_0\mathfrak{K}^{j+1}$:
$$n(\mathfrak{K}^{\sqrt{n}+2}(\mathfrak{d}_0\mathfrak{K}^{j+1})\mathfrak{K}_3 n + \mathfrak{K}^{\sqrt{n}+1}(\hat{h}_0 + \mathfrak{d}_0\mathfrak{K}^{j+1}\mathfrak{K}_1 d^{2n}) + \mathfrak{K}_4\mathfrak{d}_0\mathfrak{K}^{j+1}\mathfrak{K}^{\sqrt{n}+1})$$

substitute $\hat{h}_0 \leq \mathfrak{K}^{j+2}\mathfrak{K}_4(j+1) + \hat{\mathfrak{h}}_0\mathfrak{K}^{j+1}$:

$$n(\mathfrak{K}^{\sqrt{n}+2}(\mathfrak{d}_0\mathfrak{K}^{j+1})\mathfrak{K}_3 n + \mathfrak{K}^{\sqrt{n}+1}((\mathfrak{K}^{j+2}\mathfrak{K}_4(j+1) + \hat{\mathfrak{h}}_0\mathfrak{K}^{j+1}) + \mathfrak{d}_0\mathfrak{K}^{j+1}\mathfrak{K}_1 d^{2n}) + \mathfrak{K}_4\mathfrak{d}_0\mathfrak{K}^{j+1}K^{\sqrt{n}+1})$$

Substitute $j \leq \sqrt{n}$:

$$n(\mathfrak{K}^{\sqrt{n}+2}(\mathfrak{d}_0\mathfrak{K}^{j+1})\mathfrak{K}_3 n + \mathfrak{K}^{\sqrt{n}+1}((\mathfrak{K}^{\sqrt{n}+2}\mathfrak{K}_4(\sqrt{n}+1) + \hat{\mathfrak{h}}_0\mathfrak{K}^{\sqrt{n}+1}) + \mathfrak{d}_0\mathfrak{K}^{\sqrt{n}+1}\mathfrak{K}_1 d^{2n}) + \mathfrak{K}_4\mathfrak{d}_0\mathfrak{K}^{\sqrt{n}+1}\mathfrak{K}^{\sqrt{n}+1})$$

Now finally substitute $\mathfrak{K} \leq 4d^{6n}$ and consider the asymptotic behavior to obtain the conclusion.

□

- $R'' = \mathsf{Union}(R, R')$, and the degree and height increase by at most a factor of $\sqrt{n}$:
  - $\mathfrak{d}_{R''} \leq \sqrt{n}\mathfrak{d}_{R'} \in O(n^{3/2}\mathfrak{d}_0 d^{24n^{3/2}})$
  - $\hat{\mathfrak{h}}_{R''} \leq \sqrt{n}\hat{\mathfrak{h}}_{R'} \in$

$$O\left((\mathfrak{d}_0\mathfrak{K}_4 + \mathfrak{h}_0)n^{3/2}d^{24n^{3/2}} + (\mathfrak{K}_4\sqrt{n} + \mathfrak{d}_0(\mathfrak{K}_1 + n\mathfrak{K}_3))n^{3/2}d^{30n^{3/2}}\right)$$

- $\mathsf{ChangeVariables}(\boldsymbol{A}^{-1}, R'')$:
  - $\mathfrak{d}_{R''} \leq \mathfrak{d}_{R''}$
  - $\hat{\mathfrak{h}}_{R''} \leq \hat{\mathfrak{h}}_{R'',j} + \mathfrak{K}_4\mathfrak{d}_{R'',j} \in$

$$O\left((\mathfrak{d}_0\mathfrak{K}_4 + \mathfrak{h}_0)n^{3/2}d^{24n^{3/2}} + (\mathfrak{K}_4\sqrt{n} + \mathfrak{d}_0(\mathfrak{K}_1 + n\mathfrak{K}_3))n^{3/2}d^{30n^{3/2}}\right).$$

Remark 5.4.12. *We make a great deal of effort to obtain inequalities on degree and height. We do this because in future work we intend to develop a modular algorithm where inequalities on the output size are necessary to properly estimate prime size. Clearly, from the proof we can derive an inequality. However, the expression is complex and we therefore decided to use asymptotic notation for simplification in the main result.*

5.4.4 Height of data structures

The output of Algorithm 2, main roadmap, is a one-dimensional parameterization of the roadmap. However, the degree and canonical height bounds just derived apply to the underlying algebraic set that is the roadmap itself. Let $\mathscr{R}$ denote this roadmap, and let

- $\mathfrak{d}_{\mathscr{R}}$ be the degree of $\mathscr{R}$,
- $\hat{\mathfrak{h}}_{\mathscr{R}}$ the canonical height of $\mathscr{R}$, and

- $\mathfrak{h}_{\mathscr{R}}$ the height of $\mathscr{R}$.

In what follows we bound the hight of the output of MainRoadmap in [42], a one-dimensional parameterization of $\mathscr{R}$.

From the previous analysis:

- The degree of $\mathscr{R}$ is at most
$$O(n^{3/2}\mathfrak{d}_0 d^{24n^{3/2}}).$$
- The canonical height of $\mathscr{R}$ is at most
$$O\left((\mathfrak{d}_0\mathfrak{K}_4 + \hat{\mathfrak{h}}_0)n^{3/2}d^{24n^{3/2}} + (\mathfrak{K}_4\sqrt{n} + \mathfrak{d}_0(\mathfrak{K}_1 + n\mathfrak{K}_3))n^{3/2}d^{30n^{3/2}}\right).$$

From [16, Proposition 2.39(5)],

- $\mathfrak{h}_0 \in O(\log(n)\mathfrak{d}_0 + \hat{\mathfrak{h}}_0)$
- $\mathfrak{h}_{\mathscr{R}} \in O(\log(n)\mathfrak{d}_{\mathscr{R}} + \hat{\mathfrak{h}}_{\mathscr{R}})$.

Therefore the height of $\mathscr{R}$ is at most

$$O\left((\mathfrak{d}_0\mathfrak{K}_4 + \mathfrak{h}_0)n^{3/2}d^{24n^{3/2}} + (\mathfrak{K}_4\sqrt{n} + \mathfrak{d}_0(\mathfrak{K}_1 + n\mathfrak{K}_3))n^{3/2}d^{30n^{3/2}}\right).$$

To bound the height of a one-dimensional parameterization of $\mathscr{R}$, we can apply [15, Theorem 1]. However, first we need to apply a certain generic change of coordinates to $\mathscr{R}$. Recall from the introduction of this Chapter (Chapter 5.1) the definition of a one-dimensional parameterization: a pair

$$\mathscr{Q} = ((q, v_1, \ldots, v_n), (\lambda, \lambda')),$$

with

- bi-variate polynomials $(q, v_1, \ldots, v_n) \in \mathbb{Q}[T, U]$, q square-free and monic in T and U, with $\mathrm{GCD}(q, v_1) = 1$ and $\deg(v_i) < \deg(q)$;
- $\lambda = \lambda_1 X_1 + \ldots + \lambda_n X_n, \lambda' = \lambda'_1 X_1 + \ldots + \lambda'_n X_n$ linear forms in $X_1, \ldots, X_n$, with
  - $$\lambda(v_1, \ldots, v_n) = T\frac{\partial q}{\partial T} \mod q;$$
  - $$\lambda'(v_1, \ldots, v_n) = U\frac{\partial q}{\partial T} \mod q.$$

Then $\mathscr{R}$ is the Zariski closure

$$\mathsf{Z}(\mathscr{Q}) := \overline{\left\{ \left(\frac{v_1(\tau,\rho)}{\partial q/\partial T(\tau,\rho)}, \ldots, \frac{v_n(\tau,\rho)}{\partial q/\partial T(\tau,\rho)} \right) \in \mathbb{Q}^n \mid q(\tau,\rho) = 0, \frac{\partial q}{\partial T}(\tau,\rho) \neq 0 \right\}}.$$

We require $\mathscr{R}$ to be in generic coordinates that allow a one-dimensional parameterization with the linear forms taken simply as:

- $\lambda = X_1$;
- $\lambda' = X_2$.

We achieve this by choosing the entries of a change of coordinate matrix

$$\left[\begin{array}{cc|ccc} b_{1,1} & b_{2,1} & 0 & \cdots & 0 \\ b_{1,2} & b_{2,2} & 0 & \cdots & 0 \\ b_{1,3} & b_{2,3} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ b_{1,n} & b_{2,n} & 0 & \cdots & 1 \end{array}\right] \in \mathbb{Z}^{n\times n}$$

in the following way, respectively for columns one and two.

1. Proposition 4.5 in [30] shows the existence of a polynomial of degree at most $2(\mathfrak{d}_{\mathscr{R}})^2$ with the property that when not canceled by integers $b_{1,1}, \ldots, b_{1,n}$, then $\mathscr{R}$ is in Noether position for π_1. Consequently, $\lambda = X_1$ can be chosen.
2. From [18, Corollary 3.5], there exists another polynomial, with the same degree bound, with the property that when not canceled by integers $b_{2,1}, \ldots, b_{2,n}$ then X_2 *is separating* in a sense that allows for $\lambda' = X_2$ to be the second linear form.

Proposition 5.4.13. *Let $f \in \mathbb{Q}[X_1, \ldots, X_n]$ be a polynomial with degree d. Put $S := \{0, 1, 2, \ldots, d\} \subset \mathbb{N}$. Then there exists $b_1, \ldots, b_n \in S$, $\mathrm{ht}(b_i) \leq \log(d)$, with*

$$f(b_1, \ldots, b_n) \neq 0.$$

Proof. Consider $S^n = \{0, 1, \ldots, d\}^n$ with cardinality $(d+1)^n$. By the DeMillo–Lipton–Schwartz–Zippel lemma,

$$|V(f) \cap S^n| \leq d|S|^{n-1} = d(d+1)^{n-1} < (d+1)^n.$$

Therefore S^n must contain a non-root $(b_1, \ldots, b_n)$ with $b_1, \ldots, b_n \in S$ and

$$\mathrm{ht}(b_i) \leq \log(d).$$

□

Therefore $b_{i,j}$ have height at most

$$\log(4(\mathfrak{d}_{\mathscr{R}})^2) \in O(\log(n) + \log(\mathfrak{d}_0) + n^{3/2}\log(d)),$$

which when added to the height of $\mathscr{R}$ does not change its asymptotic upper bound.

Now we can apply [15, Theorem 1], which increases the height slightly. The heights of polynomials in a one-dimensional parameter of $\mathscr{R}$ are bounded by

$$O\left((\mathfrak{d}_0\mathfrak{K}'_4 + \mathfrak{h}_0)n^{3/2}d^{24n^{3/2}} + (\mathfrak{K}'_4\sqrt{n} + \mathfrak{d}_0(\log(\mathfrak{d}_0)n^{1/2}\log(d)\mathfrak{K}'_1 + n\mathfrak{K}'_3))n^{3/2}d^{30n^{3/2}}\right)$$

with

- $\mathfrak{K}'_1 = n(d\log(n) + H)$
- $\mathfrak{K}'_2 = n(\log(n) + h_{\boldsymbol{A}})$
- $\mathfrak{K}'_3 = \mathfrak{K}'_1 + \mathfrak{K}'_2$
- $\mathfrak{K}'_4 = \log(n) + h_{\boldsymbol{A}^{-1}}$
- $H = h + n(h_{\boldsymbol{A}} + h_{\boldsymbol{u}} + \log(d))$.

After further simplification, particularly with soft-O notation which ignores polylogarithmic terms, the height of the data structure is bounded by

$$O^{\sim}\left(\mathfrak{h}_0 d^{24n^{3/2}} + \mathfrak{d}_0(h + h_{\boldsymbol{A}} + h_{\boldsymbol{u}} + h_{\boldsymbol{A}^{-1}})d^{30n^{3/2}+1}\right).$$

This completes the proof of Theorem 5.1.3.

Chapter 6

Computing 2^a-isogenies in Legendre form

Computing 2^a-isogenies in Legendre Form

Jesse Elliott*, Aaron Hutchinson†, David Jao‡

Overview

We introduce a method for efficiently computing 2^a-isogenies in Legendre form with applications in post-quantum cryptography. An example of a secure application is the Charles-Goren-Lauter (CGL) hash function [3], which recently saw significant improvement in complexity by Doliskani et al. 2022 [4].

The majority of work on isogeny computation uses elliptic curves in Montgomery form; this includes the original work on SIDH by Jao, De Feo and Plût [8] and the sate of the art implementation of SIKE [6]. Elliptic curves in twisted Edwards form have also been used due to their efficient elliptic curve arithmetic, and complete Edwards curves have been used for their benefit of providing added security against side channel attacks (see Azarderakhsh et al. 2018 [1]). As far as we know, elliptic curves in Legendre form have not yet been explored for isogeny-based cryptography. Legendre form has the benefit of a very simple defining equation, and the simplest possible representation of the 2-torsion subgroup. In this work, we develop a new framework for constructing 2^a-isogenies using elliptic curves in Legendre form, and in doing so optimize Legendre curve arithmetic and 2-isogeny computations on Legendre curves by avoiding any square root computations.

Keywords— Post-quantum cryptography, supersingular elliptic curves, cryptographic hash functions

*David R. Cheriton School of Computer Science, University of Waterloo, On, Canada
†Cyber Engineering & Mathematics and Statistics, Louisiana Tech University
‡Department of Combinatorics & Optimization, University of Waterloo, On, Canada

1 Introduction

Isogeny-based cryptography is an active area of research in post-quantum cryptography. Isogeny-based schemes are generally compact with small key sizes, while computational time is relatively slower compared to other post-quantum frameworks. Several cryptosystems are built on the hypothesized computational hardness of the *supersingular isogeny path problem*: given a prime p, and two supersingular elliptic curves E and E' over $\mathbb{F}_{p^2}$, find a path from E to E' in the l-isogeny graph. An *l-isogeny graph*, for a prime l, is an $(l+1)$-regular graph with supersingular elliptic curves as vertices and l-isogenies as edges. l-isogeny graphs are *Ramanujan*, expander graphs with non-trivial eigenvalues having absolute value at most $2\sqrt{l-1}$; the bound is an optimal eigenvalue gap. Consequently the graph is highly connected and random walks on the graph have rapid mixing: by starting at a vertex and randomly moving from one vertex to another, a state is quickly reached where the probability of being at any given vertex is almost the same, regardless of the starting point. Therefore random paths in the graph approach a uniform distribution and recovering the path is thought to be computationally hard. In particular, the pre-image resistance of the CGL hash function [3] is based on the hardness of the l-isogeny path problem.

We introduce a method for efficiently computing 2^a-isogenies in Legendre form. An example of a secure application in post-quantum cryptography is the Charles-Goren-Lauter (CGL) hash function [3], which recently underwent a significant improvement in complexity by Doliskani et al. 2022 [4]. As far as we know, elliptic curves in Legendre form have not yet been explored for isogeny-based cryptography. Legendre form has the benefit of a very simple defining equation, and the simplest possible representation of the 2-torsion subgroup. We develop a new framework for constructing 2^a-isogenies using elliptic curves in Legendre form, and in doing so optimize Legendre curve arithmetic and 2-isogeny computations on Legendre curves by avoiding any square root computations.

1.1 Background

Elliptic Curves and Isogenies. Let $\mathbb{F}_q$ be a finite field with q elements, where $q = p^k$ for some prime p. Let E_1 and E_2 be elliptic curves over $\mathbb{F}_q$. An *isogeny* $\varphi : E_1 \to E_2$ is a non-constant rational map preserving the identity element, and also a group homomorphism from $E_1(\mathbb{F}_q)$ to $E_2(\mathbb{F}_q)$. The *degree* of an isogeny φ, denoted $\deg \varphi$, is its degree as a rational map. When an isogeny is separable and non-constant, then its degree is finite and equal to the size of its kernel. An isogeny of degree ℓ is called an *ℓ-isogeny*. Moreover, non-constant separable isogenies are determined by their kernel, up to isomorphism, in the following sense.

Theorem 1. ([9, Proposition 4.12]) Let E be an elliptic curve and let G be a finite subgroup of E. There exists a unique (up to isomorphism) elliptic curve E' and a separable isogeny $\varphi : E \to E'$ with $\ker(\varphi) = G$.

The curve E' in the theorem above is often labeled E/G and is called the *quotient of E by G*. Vélu's formulas (see Vélu 1971 [10]) provide the equations for the codomain curve E/G and

for the isogeny $\varphi : E \to E/G$. For a point P in E, Vélu's formulas give explicit equations $\varphi(P) = (\varphi_1(P), \varphi_2(P))$ where

$$\varphi_1(P) = x(P) + \sum_{Q \in G \setminus \{\mathcal{O}\}} [x(P+Q) - x(Q)]$$
$$\varphi_2(P) = y(P) + \sum_{Q \in G \setminus \{\mathcal{O}\}} [y(P+Q) - y(Q)].$$

Furthermore, non-constant separable isogenies can be factored into a composition of isogenies of prime degree over $\mathbb{F}_q$. In particular, if φ is a non-constant and separable isogeny with degree ℓ^e for some prime ℓ, then φ can be factored into a composition $\varphi = \varphi_e \varphi_{e-1} \cdots \varphi_1$ where $\deg \varphi_i = \ell$ for all $1 \leq i \leq e$.

For an elliptic curve E defined over $\mathbb{F}_q$ and an integer n, the multiplication by n map $[n] : E \to E$ is an isogeny. The kernel of $[n]$ is the n-torsion subgroup $E[n]$ of E, given by $E[n] = \{P \in E \ : \ [n]P = \mathcal{O}\}$. If n does not divide p, then $E[n]$ is isomorphic to $\mathbb{Z}/n\mathbb{Z} \oplus \mathbb{Z}/n\mathbb{Z}$ as an abelian group.

We say that E_1 and E_2 are *isogenous* if there exists a non-constant isogeny $\varphi : E_1 \to E_2$. For any such φ, there always exists a *dual isogeny* $\hat{\varphi} : E_2 \to E_1$ satisfying $\deg \varphi = \deg \hat{\varphi}$ and $\varphi\hat{\varphi} = \hat{\varphi}\varphi = [\deg \varphi]$, making the property of being isogenous an equivalence relation.

An *endomorphism* of an elliptic curve E defined over $\mathbb{F}_q$ is an isogeny $\varphi : E \to E$. The set of all endomorphisms, together with point-wise addition and function composition, forms the structure of a ring called the *endomorphism ring*, which is denoted by $\mathrm{End}(E)$. We say that E is *supersingular* when $\dim_{\mathbb{Z}}(\mathrm{End}(E)) = 4$ and we say that E is *ordinary* when $\dim_{\mathbb{Z}}(\mathrm{End}(E)) = 2$. Isogenous elliptic curves are either both supersingular or both ordinary. Every supersingular elliptic curve is isomorphic to an elliptic curve that is defined over a finite field of order p^2. The work in this paper focuses exclusively on supersingular elliptic curves, and therefore we assume in the remainder that $q = p^2$.

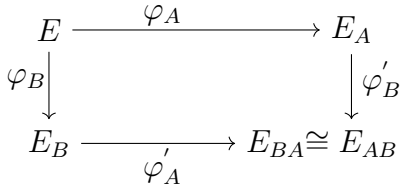


Figure 1: A high-level depiction of Supersingular Isogeny Diffie-Hellman. The initial curve E is given as a public parameter. In round 1, Alice (resp. Bob) constructs the secret isogeny φ_A (resp. φ_B). In round 2, Alice (resp. Bob) constructs another secret isogeny φ'_A (resp. φ'_B). The shared secret is the j-invariant of the final curve $E_{AB} \cong E_{BA}$.

Supersingular Isogeny Diffie-Hellman. SIDH is a key exchange algorithm proposed by Jao and De Feo 2011 [7] that was conjectured to be post-quantum secure before the attack by Castryck and Descru in 2022 [2]. Our isogeny computation builds upon the SIDH framework. We reference SIDH throughout the paper and employ the same terminology.

The security of SIDH is not based on the supersingular isogeny path problem but rather the *computational supersingular isogeny problem*: let $\varphi : E \to E_R$ be an isogeny whose kernel is $\langle P_R + \gamma Q_R \rangle$, where γ is chosen at random from $\mathbb{Z}/l_R^e\mathbb{Z}$ with $l_R \in \{2, 3\}$ and $\{P_R, Q_R\}$ generators of $E[l_R^e]$. Given E_R and the values $\varphi(P_S), \varphi(Q_S)$, for generators $\{P_S, Q_S\}$ of $E[l_S^f]$ with $l_S \in \{2, 3\}$ and $l_S \neq l_R$, find a generator R of $\langle P_R + \gamma Q_R \rangle$. The additional torsion data are exploited in the attack by Castryck and Descru 2022 [2].

We now describe SIDH at a high level. Let $p = 2^a 3^b f \pm 1$ be a prime number, where f is a small co-factor and $2^a \approx 3^b$. Let E be a supersingular elliptic curve defined over $\mathbb{F}_{p^2}$.

The key exchange provided by SIDH is a variation of Diffie-Hellman. The public parameters consist of a supersingular elliptic curve E and points $P_A, Q_A, P_B, Q_B \in E$ such that $E[2^a] = \langle P_A, Q_A \rangle$ and $E[3^b] = \langle P_B, Q_B \rangle$. In the *first round* of SIDH, Alice chooses integers $m_A, n_A \in \mathbb{Z}/2^a\mathbb{Z}$ as her secret key and computes an isogeny $\varphi_A : E \to E_A$ with

$$E_A := E/\langle m_A P_A + n_A Q_A \rangle, \qquad \ker(\varphi_A) = \langle m_A P_A + n_A Q_A \rangle.$$

Similarly, Bob chooses integers $m_B, n_B \in \mathbb{Z}/3^b\mathbb{Z}$ as his secret key and computes an isogeny $\varphi_B : E \to E_B$ with

$$E_B := E/\langle m_B P_B + n_B Q_B \rangle, \qquad \ker(\varphi_B) = \langle m_B P_B + n_B Q_B \rangle.$$

Now, akin to Diffie-Hellman, Alice and Bob exchange some information: Alice sends $(E_A, \varphi_A(P_B), \varphi_A(Q_B))$ to Bob, and Bob sends $(E_B, \varphi_B(P_A), \varphi_B(Q_A))$ to Alice. The *second round* then begins, where Alice computes an isogeny $\varphi'_A : E_B \to E_{BA}$ and Bob computes an isogeny $\varphi'_B : E_A \to E_{AB}$ such that:

$$E_{BA} := E_B/\langle m_A\varphi_B(P_A) + n_A\varphi_B(Q_A) \rangle, \qquad E_{AB} := E_A/\langle m_B\varphi_A(P_B) + n_B\varphi_A(Q_B) \rangle,$$
$$\ker(\varphi'_A) = \langle m_A\varphi_B(P_A) + n_A\varphi_B(Q_A) \rangle, \qquad \ker(\varphi'_B) = \langle m_B\varphi_A(P_B) + n_B\varphi_A(Q_B) \rangle.$$

The two curves E_{AB} and E_{BA} are isomorphic since they are both isomorphic to the curve

$$E/\langle m_B P_B + n_B Q_B, m_A P_A + n_A Q_A \rangle.$$

The shared key is then the j invariant of $E_{BA} \cong E_{AB}$.

Supersingular Isogeny Key Encapsulation. The most current and optimized version of SIDH is Supersingular Isogeny Key Encapsulation (SIKE) [6]. In 2016, Galbraith, Petit, Shani and Ti showed that if a static public key is used in SIDH then the protocol becomes vulnerable to an *active attack*, where an advisary can fully recover the corresponding secret key in a number of iterations equalling the bit-length of the secret [5]. In SIKE, one applies a variant of the Fujisaki-Okamoto transform to force Bob to reveal his encryption key to Alice, which Alice then uses to re-encrypt Bob's ciphertext and verify its validity, thus preventing the active attack.

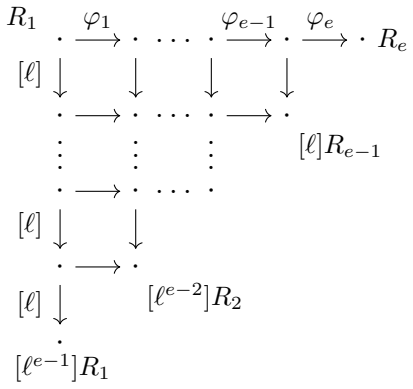


Figure 2: A visual representation of the computational structure of computing $\varphi = \varphi_e \dots \varphi_1$.

Computing Isogenies in SIDH: Strategies. In both rounds of SIDH, both Alice and Bob are tasked with computing (and possibly evaluating) an isogeny $\varphi : E \to E/\langle R \rangle$ for some curve E and some point $R \in E$ of order ℓ^e for $\ell \in \{2, 3\}$ and $e \in \{a, b\}$, such that $\ker(\varphi) = \langle R \rangle$. Applying Vélu's formulas directly to $\langle R \rangle$ to construct φ is not efficient; the computation requires $O(\deg \varphi) = O(\ell^e) = O(\sqrt{p})$ field operations. For better efficiency, the isogeny construction is instead broken up into a decomposition of prime degree isogenies φ_i, $1 \le i \le e$, with $\varphi = \varphi_e \varphi_{e-1} \cdots \varphi_1$. Using this method, φ can be computed with $e \times O(\deg \varphi_i) = O(e\ell)$ field operations. We now describe in detail how the construction of $\varphi_e, \dots, \varphi_1$ is made efficient.

As in Jao and De Feo. 2011 [8], we let $E_1 = E$ and $R_1 = R$, and for $1 \le i \le e$ define

$$E_{i+1} = E_i/\langle \ell^{e-i} R_i \rangle, \quad \varphi_i : E_i \to E_{i+1}, \quad R_{i+1} = \varphi_i(R_i).$$

Then $E/\langle R \rangle = E_e$ and $\varphi = \varphi_e \dots \varphi_1$ has kernel $\langle R \rangle$. Each curve E_{i+1} and isogeny φ_i (which has degree ℓ) can be computed using Vélu's formulas once the subgroup $\langle \ell^{e-i} R_i \rangle$ of E_i is known.

Figure 2 provides a visual representation of the computational structure of the problem. Vertices represent elliptic curve points, with points on the same positive diagonal having the same order, and points on the same vertical belonging to the same curve. Downward edges represent point multiplications by ℓ, and right edges represent ℓ-isogeny evaluations. The goal is to compute all the order ℓ points along the bottom diagonal, from which we can apply Vélu's formulas to compute the corresponding φ_i. Figure 2 motivates the idea of a strategy, originally defined in Jao De Feo and J. Plût 2014 [8].

Definition 1. For positive integers n, let $T_n = (V, E)$ be the graph defined as follows:

1. The set of vertices V consists of all points in the plane with integer coordinates which lie inside or on the boundary of the region bounded by the lines $x = 0, y = 0$, and $y = -x - n + 1$.

2. The edge set E consists of all line segments of unit length which connect two vertices in V.

It follows from Definition 1 that every edge of T_n is either horizontal or vertical. We turn T_n into a directed graph by orienting all horizontal edges to the right and all vertical edges downward. We say that a vertex is a *leaf* if it has no outgoing edges (consequently, the leaves of T_n are exactly the integral points on the line $y = -x - n + 1$).

Definition 2. A **strategy** S is a subgraph of T_n such that:

1. The vertex $(0, 0)$ and all vertices on the line $y = -x - n + 1$ are in S.
2. For each vertex v on the line $y = -x - n + 1$, there is a path from $(0, 0)$ to v in S.
3. No two edges of S share the same target.
4. There are no leaves in S distinct from the leaves of T_n.

We write $|S| = n$ when S is a strategy in T_n. Any strategy yields a valid algorithm to compute the isogeny $\varphi = \varphi_e \dots \varphi_1$ by decorating T_e as in Figure 2 (a consequence of Lemma 4.2 from Jao et al. 2014 [8]).

We now give a method for combining two strategies together to form a larger strategy. For strategies S_1 and S_2 with $|S_1| = n_1$ and $|S_2| = n_2$, define a strategy $S_1 \# S_2$ in $T_{n_1+n_2}$ by:

1. $S_1 \# S_2$ contains the (unique) path connecting $(0, 0)$ to $(n_2, 0)$.
2. $S_1 \# S_2$ contains the (unique) path connecting $(0, 0)$ to $(0, -n_1)$.
3. $S_1 \# S_2$ contains S_1 as a subgraph, shifted to the right n_2 units.
4. $S_1 \# S_2$ contains S_2 as a subgraph, shifted down n_1 units.

We refer to the binary operation # as *join*; note that it is nonassociative and noncommutative.

Definition 3. A strategy S in T_n is **canonical** if S can be expressed as $n - 1$ many applications of the join operator on the strategy T_1. That is, S is some parenthesization of $\underbrace{T_1 \# T_1 \# \dots \# T_1}_{\text{n}}$.

Two naive canonical strategies immediately come to mind. We define the nth *multiplication-based* strategy A_n and *isogeny-based* strategy B_n recursively as $A_1 = B_1 = T_1$ and $A_{n+1} = T_1 \# A_n$ and $B_{n+1} = B_n \# T_1$. Figure 3 shows 3 strategies in T_5: the multiplication-based strategy, the isogeny-based strategy, and a canonical strategy.

When a strategy is used in SIDH, horizontal edges correspond to isogeny evaluations and vertical edges correspond to point multiplications. Furthermore, all vertical edges have a common computational cost and all horizontal edges have a common computational cost. To get a sense of the computational effort required to construct an isogeny of degree ℓ^e with a given strategy, we assign weights to each edge in a strategy. We define a *measure* as a pair $(\mathfrak{p}, \mathfrak{q})$ of positive real numbers, and turn a strategy S into a weighted strategy as follows.

Definition 4. A **weighted strategy** with the measure $(\mathfrak{p}, \mathfrak{q})$ is a strategy turned into a weighted graph, where vertical edges have weight $\mathfrak{p}$ and horizontal edges have weight $\mathfrak{q}$. The **cost** of the weighted strategy is the sum of the weights of all the edges in the strategy. An **optimal strategy** is one of minimal cost.

In Jao et al. 2014 [8], Proposition 4.6 shows that every optimal strategy must be canonical.

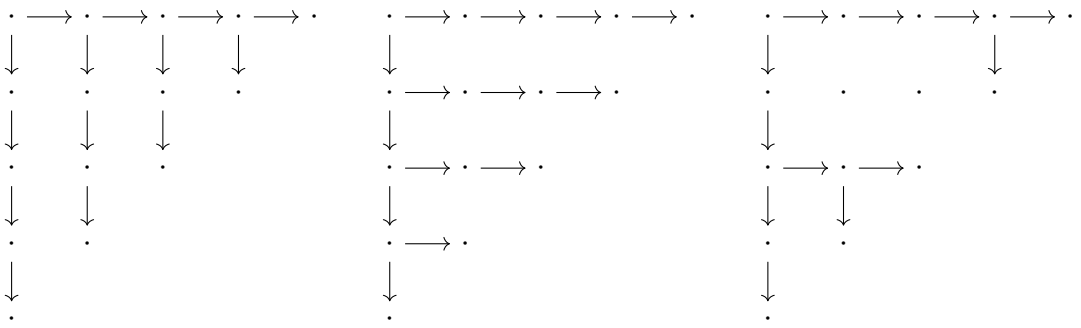


Figure 3: The multiplication-based strategy (left), the isogeny-based strategy (middle), and a canonical strategy (right) in T_5.

Montgomery Curves. A Montgomery curve $M_{\mathfrak{b},\mathfrak{a}}$ is given by an equation of the form $\mathfrak{b}y^2 = x^3 + \mathfrak{a}x^2 + x$, where $\mathfrak{a}, \mathfrak{b} \in \mathbb{F}_q$ are the curve coefficients. In isogeny-based cryptography, most formulas do not require the value of $\mathfrak{b}$ and it is often left unspecified. To avoid field inversions, it is often useful to projectivize the variable $\mathfrak{a}$ as $[\mathfrak{a} : 1] \in \mathbb{P}^1$, so that $[A : C] = [\mathfrak{a} : 1]$ if and only if $C\mathfrak{a} = A$. Furthermore, the most efficient implementations involving Montgomery curves, such as [6], use the quantity $[A_{24}^+ : C_{24}]$ in place of $[A : C]$, where $[A_{24}^+ : C_{24}] = [A + 2C : 4C]$.

The Choice of Curve Model. The majority of work based on an SIDH framework uses Elliptic curves in Montgomery form; this includes the original work by Jao, De Feo and Plût [7, 8] and the sate of the art, SIKE [6]. Elliptic curves in twisted Edwards form have also been used because of their efficient elliptic curve arithmetic, and complete Edwards curves have been used for their benefit of providing added security against side channel attacks (see Azarderakhsh et al. 2018 [1]).

As far as we know, elliptic curves in Legendre form have not yet been explored for isogeny-based cryptography. Legendre form has the benefit of a very simple defining equation, and the simplest possible representation of the 2-torsion subgroup. We therefore find motivation in using Legendre form for constructing 2^a-isogenies. Our algorithms are founded on the SIDH framework and find applications for instance in isogeny computation for the Charles-Goren-Lauter (CGL) hash function [3, 4].

1.2 Contributions

The main contributions of this work are as follows.

1. We describe in detail how a degree 2^a isogeny can be constructed between elliptic curves in Legendre form for use in isogeny-based cryptographic protocols such as the CGL hash function [3, 4]. To achieve this, we derive many results for elliptic curves in Legendre form and degree 2 isogenies between curves in Legendre form in Section 2. Naively constructing a degree 2 isogeny between Legendre curves requires the computation of a square root. Our formula for 2-isogenies shows that the computation of this square root can be avoided when an order 4 point above the kernel generator is known. Furthermore, we also describe how to convert between Montgomery form and Legendre form and give explicit isomorphisms for this task.

2. We detail algorithms which implement the theoretic results on Legendre curves from Section 2. In particular, we describe Algorithm 7, which can be viewed as an analogue of Algorithm 19 of SIKE [6] adapted to the setting of Legendre curves. Algorithm 7 takes as input the projectivized coefficients $[\kappa_1 : \mu_1]$ of a Legendre curve $L_{[\kappa_1:\mu_1]}$ and the coordinates $(X_R : Z_R)$ of a point on $L_{[\kappa_1:\mu_1]}$ of order exactly 2^a. The algorithm constructs a degree 2^a isogeny $\varphi : L_{[\kappa_1:\mu_1]} \to L_{[\kappa_{a+1}:\mu_{a+1}]}$ to some Legendre curve $L_{[\kappa_{a+1}:\mu_{a+1}]}$ such that $\ker(\varphi) = \langle R \rangle$; this is performed by constructing a sequence of 2-isogenies $\varphi_1, \ldots, \varphi_a$, such that $\varphi_i : L_{[\kappa_i:\mu_i]} \to L_{[\kappa_{i+1}:\mu_{i+1}]}$ is a degree 2 isogeny between Legendre curves $L_{[\kappa_i:\mu_i]}$ and $L_{[\kappa_{i+1}:\mu_{i+1}]}$, for $1 \leq i \leq a$, with $\varphi = \varphi_a \cdots \varphi_1$. Algorithm 7 relies on Algorithms 1, 2, 3, and 4 which efficiently implement the results from Section 2.

3. In Section 4 we further describe an open problem—the original motivation for this work—which if solved would skip the strategy traversal performed in SIDH altogether, using the fact that the 2-torsion of Legendre curves has a simple representation and is known in advance for all intermediate curves.

1.3 Paper Organization

In Section 2, we discuss elliptic curves in Legendre form. We give many results regarding their arithmetic, torsion structure, and construction of 2-isogenies between them. We also describe how to convert between Montgomery form and Legendre form and give explicit isomorphisms for this task. In Section 3, we provide a collection of algorithms which implement the results from Section 2; these are designed for 2^a-isogeny computation with applications for instance to the CGL hash function [3, 4]. We discuss these algorithms in detail and explain how they relate to the results from Section 2. In Section 4, we give concluding remarks and discuss some future work.

2 Formulas for Legendre Form

This section provides formulas for various operations on elliptic curves in Legendre form. Throughout we use $\mathbb{F}$ to denote an arbitrary field.

2.1 Legendre Form

For $\lambda \in \mathbb{F}$ with $\lambda \neq 0, 1$, the *Legendre curve* L_λ *with coefficient* λ is the elliptic curve in $\mathbb{P}^2(\mathbb{F})$ given by the affine equation

$$L_\lambda : y^2 = x(x-1)(x-\lambda), \tag{1}$$

where λ is called the *Legendre coefficient*. The *j-invariant* of L_λ is the value

$$j(\lambda) = 2^8 \frac{(\lambda^2 - \lambda + 1)^3}{\lambda^2(\lambda-1)^2} \in \mathbb{F},$$

and two curves are isomorphic if and only if they have the same j-invariant. The mapping $j : \mathbb{F} \setminus \{0, 1\} \to \mathbb{F}$ given by $\lambda \mapsto j(\lambda)$ is exactly six-to-one, except above $j(\lambda) = 0$ and $j(\lambda) = 1728$; the six values mapping to $j(\lambda) \neq 0, 1728$ under j are

$$\left\{\lambda, \frac{1}{\lambda}, 1-\lambda, \frac{1}{1-\lambda}, \frac{\lambda}{\lambda-1}, \frac{\lambda-1}{\lambda}\right\},$$

and therefore these six values define the same Legendre curve up to isomorphism.

2.2 Projective Coordinates and Coefficients

We embed affine $\mathbb{F}$ space into projective space $\mathbb{P}(\mathbb{F})$ using the usual mapping. Specifically, $\lambda \in \mathbb{F}$ is represented in projective form as $[\lambda : 1]$. Projective form has the property that for any $\kappa, \mu \in \mathbb{F}$ not both zero and $c \in \mathbb{F}$ nonzero, we have that $[\kappa : \mu] = [c\kappa : c\mu]$. We will sometimes abuse notation by writing $\lambda = [\lambda : 1]$, and so $[\kappa : \mu] = \kappa/\mu$ when μ is nonzero.

For efficiency reasons, it will be useful to use projective coordinates for the coefficient of our elliptic curves. We therefore give an alternative definition of a Legendre curve which uses a curve coefficient in projective form.

Definition 5. Let $[\kappa : \mu] \in \mathbb{P}(\mathbb{F})$ with $(\kappa - \mu)\kappa\mu \neq 0$. The *Legendre curve* $L_{[\kappa:\mu]}$ *with coefficient* $[\kappa : \mu]$ is the elliptic curve in $\mathbb{P}^2(\overline{\mathbb{F}})$ defined by the homogeneous polynomial

$$L_{[\kappa:\mu]} : \quad \mu Y^2 Z = X(X-Z)(\mu X - \kappa Z). \tag{2}$$

Using the projective form of the curve coefficient will help reduce the computational cost of constructing isogenies later in this section. Going forward, we work entirely with projective coefficients.

2.3 Legendre Arithmetic

Points on the Legendre curve $L_{[\kappa:\mu]}$ are elements of $\mathbb{P}^2(\overline{\mathbb{F}})$ which satisfy the defining homogeneous polynomial given in Equation 2. These points form an abelian group through the *group law*, with the single *point at infinity* $\mathcal{O} = [0 : 1 : 0]$ playing the role of the group identity. The following theorem describes part of the group law for Legendre form.

Theorem 2. Let $L_{[\kappa:\mu]}$ be a Legendre curve. The points of $L_{[\kappa:\mu]}$ form an abelian group with identity $\mathcal{O} = [0 : 1 : 0]$, and the doubling of $[X : Y : Z] \in L_{[\kappa:\mu]}$ is given by $2[X : Y : Z] = [X' : Y' : Z']$, where

$$\begin{aligned} X' &= (\mu X^2 - \kappa Z^2)^2, \\ Y' &= \frac{(\mu X^2 - 2\kappa XZ + \kappa Z^2)(\mu X^2 - 2\mu XZ + \kappa Z^2)(\mu X^2 - \kappa Z^2)}{2\mu YZ}, \\ Z' &= 4\mu XZ(\mu X^2 - (\kappa + \mu)XZ + \kappa Z^2) \end{aligned}$$

when $YZ \neq 0$.

Proof. Legendre form is a special case of generalized Weierstrass form, and so the theorem follows immediately from well-known results. See Silverman 2008 [9] for details. □

Notice that X' and Z' above depend only on X, Z, κ, and μ, and so even if the value of Y is not known repeated doubling may be partially computed by ignoring Y'. If the X and Z coordinates of some point are known, the value of Y can be determined up to sign through Equation 2.

2.4 The 2-Torsion of a Legendre Curve

One of the most appealing features of a Legendre curve is the simplicity of their 2-torsion points. Recall for $n \in \mathbb{N}$ that the *n-torsion* of a curve $L_{[\kappa:\mu]}$ is the subgroup $L_{[\kappa:\mu]}[n] = \{P \in L_{[\kappa:\mu]} : nP = \mathcal{O}\}$.

Theorem 3. The 2-torsion subgroup of $L_{[\kappa:\mu]}$ is isomorphic to $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$ and has the form

$$L_{[\kappa:\mu]}[2] = \{\mathcal{O}, [0 : 0 : 1], [1 : 0 : 1], [\kappa : 0 : \mu]\}.$$

Given $[X : Y : Z] \in L_{[\kappa:\mu]}[2]$, at first glance the nonuniqueness of projective representations of points may give the impression that determining which of the four above points $[X : Y : Z]$ corresponds to may require at least a field multiplication. Upon closer inspection we see the following algorithm, requiring at most three field equality checks:

$$[X : Y : Z] = \begin{cases} \mathcal{O} & \text{if } Z = 0, \\ [0 : 0 : 1] & \text{if } X = 0, \\ [1 : 0 : 1] & \text{if } X = Z, \\ [\kappa : 0 : \mu] & \text{otherwise.} \end{cases}$$

2.5 The 4-Torsion of a Legendre Curve

Here we give a description of $L_{[\kappa:\mu]}[4]$. The 4-torsion consists of 16 points, 4 of which are the 2-torsion subgroup $L_{[\kappa:\mu]}[2]$ given previously. Aside from these, there are 12 points of order 4, which can be partitioned into 3 subsets based upon which 2-torsion point they double to. That is, if $T \in L_{[\kappa:\mu]}[2]$ has order 2, then each of the subsets $S_T = \{P \in L_{[\kappa:\mu]} : 2P = T\}$ has size 4 and do not intersect each other. For such a fixed T, the points within S_T can be written to have a very similar expression, which only differs by at most two minus signs. The computation to determine each S_T is tedious, and the end result is given in Figure 4. The three points in the Order 4 row of Figure 4 each represent one of the sets S_T, where the value of T is the point in the Order 2 row directly below.

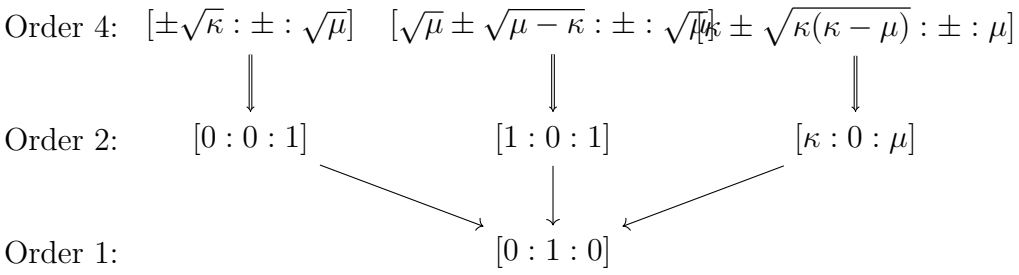


Figure 4: The 4-torsion of $L_{[\kappa:\mu]}$. Each point in the Order 4 row above actually represents four distinct points of order 4 on $L_{[\kappa:\mu]}$, found by taking all possible choices of sign; for brevity the Y-coordinates have been omitted. All arrows represent the doubling map, with double arrows indicating that all four points at the tail are mapped to the same point at the tip.

2.6 2-Isogenies: Legendre to Legendre

We will be primarily interested in 2-isogenies between Legendre curves. Such isogenies will have a kernel generated by some 2-torsion point, which were fully described in Section 2.4. This subsection gives constructions for 2-isogenies between Legendre curves, restricting to the cases in which the kernel is generated by either of the 2-torsion points $T_0 := [0 : 0 : 1]$ or $T_1 := [1 : 0 : 1]$.

One may attempt to apply Velu's formulas directly to $T \in \{[0 : 0 : 1], [1 : 0 : 1]\}$ to obtain a degree 2 isogeny $\psi : L_{[\kappa:\mu]} \to E$, and then post compose with an isomorphism which puts E into Legendre form. This approach works, except that the resulting expressions for the codomain curve coefficients and point image coordinates involve square roots of κ and μ. Since in most instances these square roots are not efficiently computable, we avoid this problem by using the coordinates of an order 4 point P_4 such that $2P_4 = T$. Theorems 4 and 5 below give formulas for a degree 2 isogeny between Legendre curves using the coordinates of P_4.

Theorem 4. Let $L_{[\kappa:\mu]}$ be a Legendre curve. Assume that $P_4 = [X_{P_4} : Y_{P_4} : Z_{P_4}]$ is a point of order 4 on the curve with $2P_4 = [0 : 0 : 1]$. The 2-isogeny whose kernel is $\langle [0 : 0 : 1] \rangle$

can be written in the form $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$ with $\varphi([X : Y : Z]) = [X' : Y' : Z']$ for any $[X : Y : Z] \in L_{[\kappa:\mu]}$, where

$$\begin{aligned} \kappa' &= \kappa Z_{P_4} + 2\mu X_{P_4} + \mu Z_{P_4}, & X' &= \mu Z_{P_4} X^2 + 2\mu X_{P_4} XZ + \kappa Z_{P_4} Z^2, \\ \mu' &= 4\mu X_{P_4}, & Y' &= \frac{\mu X^2 - \kappa Z^2}{2(\kappa/\mu)^{1/4} X} Z_{P_4} Y, \\ & & Z' &= \mu' XZ. \end{aligned}$$

Proof. The theorem can be verified as follows. Write $L_{[\kappa:\mu]}$ in affine form L_λ and apply Velu's formulas for generalized Weierstrass form on L_λ with kernel $\langle (0,0) \rangle$. This results in a 2-isogeny $\psi : L_\lambda \to E$ for some curve E. One can then construct an isomorphism $\theta : E \to L_{\lambda'}$ to a Legendre curve $L_{\lambda'}$ using well-known results (Proposition 2.16 of [11]). Projectivizing L_λ, $L_{\lambda'}$ and the composition $\theta\psi$ yields a 2-isogeny between Legendre curves with kernel $[0 : 0 : 1]$, but many of the expressions involve square roots. The expressions stated in the theorem can be obtained using the fact that $X_{P_4}\sqrt{\mu} = Z_{P_4}\sqrt{\kappa}$, where $P_4 = [X_{P_4} : Y_{P_4} : Z_{P_4}] \in L_{[\kappa:\mu]}$ satisfies $2P_4 = [0 : 0 : 1]$. □

For φ as above, notice that $\varphi([1 : 0 : 1]) = [\kappa' : 0 : \mu'] = \varphi([\kappa : 0 : \mu])$. Consequently, it follows from the theory of dual isogenies that the isogeny having kernel $\langle [\kappa' : 0 : \mu'] \rangle$ corresponds to the dual of φ.

Theorem 5. Let $L_{[\kappa:\mu]}$ be a Legendre curve. Assume that $P_4 = [X_{P_4} : Y_{P_4} : Z_{P_4}]$ is a point of order 4 on the curve with $2P_4 = [1 : 0 : 1]$. The 2-isogeny whose kernel is $\langle [1 : 0 : 1] \rangle$ can be written in the form $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$ with $\varphi([X : Y : Z]) = [X' : Y' : Z']$ for any $[X : Y : Z] \in L_{[\kappa:\mu]}$, where

$$\begin{aligned} \kappa' &= \kappa Z_{P_4} + 2\mu X_{P_4} - 4\mu Z_{P_4}, & X' &= \mu Z_{P_4} X^2 + (\kappa' - \kappa Z_{P_4}) XZ - \kappa' Z^2, \\ \mu' &= 4\mu X_{P_4} - 4\mu Z_{P_4}, & Y' &= \frac{\mu X^2 - 2\mu XZ + \kappa Z^2}{2(1 - \frac{\kappa}{\mu})^{1/4}(X - Z)} Z_{P_4} Y, \\ & & Z' &= \mu'(XZ - Z^2). \end{aligned}$$

The proof for Theorem 5 is very similar to that of Theorem 4. As before, for φ as in Theorem 5 we have that $\varphi([0 : 0 : 1]) = [\kappa' : 0 : \mu'] = \varphi([\kappa : 0 : \mu])$, and so $\langle [\kappa' : 0 : \mu'] \rangle$ is the subgroup corresponding to the dual of φ.

Because $[\kappa' : 0 : \mu'] \in L_{[\kappa:\mu]}$ always corresponds to the dual of φ when $\ker(\varphi) = \langle T \rangle$ for $T \in \{[0 : 0 : 1], [1 : 0 : 1]\}$, isogenies with kernel $\langle [\kappa : 0 : \mu] \rangle$ never need to be considered when constructing a degree 2^a isogeny with kernel $\langle R \rangle$ such that $2^{a-1}R \neq [\kappa : 0 : \mu]$. This fact will be used implicitly when formulating algorithms for 2^a-isogeny computation (see Algorithm 7 to come).

2.7 2-Isogenies: Legendre to Montgomery

The following theorem describes how to construct a 2-isogeny from a Legendre curve to a Montgomery curve, and also how to obtain the j-invariant of the codomain curve.

Theorem 6. Let $L_{[\kappa:\mu]}$ be a Legendre curve. Define the curves M_0 and M_1 and maps $\varphi_0 : L_{[\kappa:\mu]} \to M_0$ and $\varphi_1 : L_{[\kappa:\mu]} \to M_1$ by:

$$M_0 : \frac{\mu}{\kappa-\mu}Y^2Z = X^3 + \frac{2(\kappa+\mu)}{\kappa-\mu}X^2Z + XZ^2$$
$$M_1 : \quad \frac{\mu}{\kappa}Y^2Z = X^3 + \frac{2(\kappa-2\mu)}{\kappa}X^2Z + XZ^2$$
$$\varphi_0([X:Y:Z]) = [(X-Z)(\mu X-\kappa Z) : \frac{\mu X^2-\kappa Z^2}{X}Y : (\kappa-\mu)XZ]$$
$$\varphi_1([X:Y:Z]) = [X(\mu X-\kappa Z) : \mu(X-Z+\frac{(\kappa-\mu)Z^2}{(X-Z)\mu})Y : \kappa(X-Z)Z]$$

Then M_0 and M_1 are Montgomery curves and φ_0 and φ_1 are degree 2 isogenies with $\ker(\varphi_0) = \langle[0:0:1]\rangle$ and $\ker(\varphi_1) = \langle[1:0:1]\rangle$. The j-invariants of M_0 and M_1 are

$$j(M_0) = \frac{(\kappa^2+14\kappa\mu+\mu^2)^3}{(\kappa\mu(\kappa-\mu))^2}, \qquad j(M_1) = \frac{(\kappa^2-16\kappa\mu+16\mu^2)^3}{(\kappa\mu(\kappa-\mu))^2}.$$

Proof. Everything in the theorem can be verified as follows. We work in affine coordinates so that $\kappa = \lambda$ and $\mu = 1$. Apply the standard Velu's formulas for Weierstrauss form to L_λ in each case to arrive at the (affinized) curves

$$E_0 : y^2 = x^3 - (\lambda+1)x^2 - 4\lambda x + 4\lambda(\lambda+1),$$
$$E_1 : y^2 = x^3 - (\lambda+1)x^2 + (6\lambda-5)x - (\lambda-1)(4\lambda-3).$$

The curves E_0 and E_1 contain the 2-torsion points $P_0 := (\lambda+1, 0)$ and $P_1 := (\lambda-1, 0)$, respectively. The other 2-torsion points on each curve can then be found by factoring the cubic defining the curve. The curves M_0 and M_1 (respectively isomorphic to E_0 and E_1) can be derived by using an affine change of coordinates to some curve $By^2 = x^3 + Ax^2 + x$ such that P_i is sent to $(0,0)$ and the x-coordinates of the other 2-torsion points are sent to the zeros of $x^2 + Ax + 1$. Composing the Velu maps with these change of coordinates and projectivizing the result yields φ_0 and φ_1. □

Note of course the j-invariant is unaffected by what form the curve is in – Montgomery or Legendre. Therefore Theorem 6 can be used to compute the j-invariant in the last step of Algorithm 4.

2.8 Isomorphisms: Montgomery and Legendre

Here we state formulas for converting between Legendre and Montgomery forms of elliptic curves.

Lemma 1. Let $M_{A,B}$ be a Montgomery curve. Define $\kappa = A^2 - A\sqrt{A^2-4} - 2$ and $\mu = 2$. Then

$$\varphi_{M,L} : M_{A,B} \to L_{[\kappa:\mu]}, \qquad \varphi_{M,L}([X:Y:Z]) = \left[2X : \sqrt{\frac{8B}{-A-\sqrt{A^2-4}}}Y : -(A+\sqrt{A^2-4})Z\right]$$

$$\varphi_{L,M} : L_{[\kappa:\mu]} \to M_{A,B}, \qquad \varphi_{L,M}([X:Y:Z]) = \left[-(A+\sqrt{A^2-4})X : \frac{(-A-\sqrt{A^2-4})^{3/2}}{\sqrt{8B}}Y : 2Z\right]$$

are inverse isomorphisms.

Proof. The Lemma follows from well-known results. See Silverman 2008 [9] and Washington [11] 2008 for details. □

If one instead starts with a Legendre curve $L_{[\kappa:\mu]}$ and defines $A = (\kappa+\mu)/\sqrt{\kappa\mu}$ and $B = 1$, the same maps above allow one to convert from Legendre to Montgomery form.

3 Algorithms

Here we discuss our algorithms that implement the results given in previous sections.

3.1 The Main Algorithm

Algorithm 7 is the top-level algorithm that computes the 2^a-isogeny $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$ with $\ker(\varphi) = \langle R \rangle$ for a point R of exact order 2^a with $2^{a-1}R \neq [\kappa : 0 : \mu]$. Algorithm 7 can be seen as the analog of Algorithm 19 from SIKE [6] adapted to our setting. However, note that Algorithm 7 uses 2-isogenies whereas Algorithm 19 of SIKE [6] uses 4-isogenies. Furthermore, since points of order 4 are used as a means of constructing 2-isogenies, we use a strategy S in T_{a-1}. The strategy S is computed ahead of time using a dynamic programming algorithm (see D. Jao et al. [8, Section 4.2.2] for details), and is given as a parameter to Algorithm 7 in linearized form. The input of Algorithm 7 consists of the Legendre curve coefficients $[\kappa : \mu] \in \mathbb{P}^1$ and the coordinates $[X_R : Z_R]$ of the point $R \in L_{[\kappa:\mu]}$ with the Y coordinate omitted. The j-invariant of the final curve $L_{[\kappa':\mu']}$ is given as output.

3.2 Subroutines

By traversing the strategy S, Algorithm 7 constructs a many 2-isogenies of the form $\varphi_i : L_{[\kappa_i:\mu_i]} \to L_{[\kappa_{i+1}:\mu_{i+1}]}$ for $1 \leq i \leq a$. The composition $\varphi_a \ldots \varphi_1$ is the desired isogeny φ from Section 3.1. To ease the presentation, Algorithm 7 uses Algorithms 1, 2, 3 and 4 as subroutines, which we give an overview of now. Throughout all algorithms, the Y coordinate of every point is omitted.

Algorithm 1 computes the codomain curve coefficients $[\kappa_{i+1} : \mu_{i+1}]$ for an isogeny of the form $\varphi_i : L_{[\kappa_i:\mu_i]} \to L_{[\kappa_{i+1}:\mu_{i+1}]}$, as well as coefficients c_0, c_1, c_2 used to evaluate φ_i at a point. Algorithm 1 takes the coefficients $[\kappa_i : \mu_i]$ and a point $P_4 \in L_{[\kappa_i:\mu_i]}$ of order (exactly) 4 as

input, where it is assumed that $2P_4 \in \{[0:0:1],[1:0:1]\}$. First, the algorithm checks whether $2P_4 = [0:0:1]$ is true by checking the value of $\mu X_{P_4}^2 - \kappa Z_{P_4}^2$, the partial doubling of P_4 (see Theorem 2); a bit b is set as 0 if $\mu X_{P_4}^2 = \kappa Z_{P_4}^2$ and 1 otherwise, where $b = 0$ indicates that $2P_4 = [0:0:1]$. Depending on the value of b, the values of $[\kappa_{i+1} : \mu_{i+1}]$ and c_0, c_1, c_2 are computed according to Theorem 4 or Theorem 5. Note that $\overline{b} = 1$ if and only if $b = 0$. The quantities c_0, c_1, c_2 are the coefficients of X^2, XZ, Z^2, respectively, in the expression for X' of Theorem 4 or Theorem 5. Algorithm 1 uses conditional swaps to avoid branching based on the value of $2P_4$.*

Algorithm 2 evaluates the 2-isogenies $\varphi_i : L_{[\kappa_i:\mu_i]} \to L_{[\kappa_{i+1}:\mu_{i+1}]}$ at a given point. It receives the codomain curve coefficient μ_{i+1} (κ_{i+1} is not needed), the evaluation coefficients c_0, c_1, c_2 and the bit variable b computed in Algorithm 1, and a point $Q \in L_{[\kappa_i:\mu_i]}$ to evaluate under φ_i. The output of Algorithm 2 is the coordinates of the point $\varphi(Q) \in L_{[\kappa_{i+1}:\mu_{i+1}]}$, computed using the formulas given in Theorems 4 and 5. Again, conditional swaps are used with the value of b to avoid branching.

Algorithm 3 performs coordinate doubling for Legendre curves. The input consists of curve coefficients $[\kappa : \mu]$ for a Legendre curve $L_{[\kappa:\mu]}$ and the coordinates $(X_P : Z_P)$ for a point $P \in L_{[\kappa:\mu]}$. The output consists of the coordinates $(X_{[2]P} : Z_{[2]P})$ corresponding to $[2]P$, computed using Theorem 2.

Algorithm 4 computes the j-invariant of the final Legendre curve $L_{[\kappa_{a+1}:\mu_{a+1}]}$ from the isogeny φ_{a-1} and the point $P_4 \in L_{[\kappa_{a-1}:\mu_{a-1}]}$ used to generate φ_{a-1}. The input to Algorithm 4 is the same as that of Algorithm 1. Similar to Algorithm 1, Algorithm 4 performs a partial evaluation of the input point P_4 to determine if the image is $[0:0:1]$ on $L_{[\kappa_a:\mu_a]}$. Depending on the result, the proper formula from Theorem 6 is used to compute the j-invariant of a Montogmery curve $M_{\mathfrak{b},\mathfrak{a}}$ using conditional swaps; this curve is isomorphic to the Legendre curve $L_{[\kappa_{a+1}:\mu_{a+1}]}$ therefore having the same j-invariant. In this way, the curve coefficients of $L_{[\kappa_{a+1}:\mu_{a+1}]}$ and the isogeny φ_a coefficients are not computed.

3.3 Montgomery algorithms

Algorithm 5 computes the quantities $[A_{24}^+ : C_{24}]$ of the codomain Montgomery curve $M_{\mathfrak{b},\mathfrak{a}}$ of an isogeny $\varphi' : L_{[\kappa':\mu']} \to M_{\mathfrak{b},\mathfrak{a}}$, as well as an isogeny evaluation coefficient C. The input to Algorithm 5 consists of the curve coefficients $[\kappa' : \mu']$, a point $(X_{P_4} : Z_{P_4}) \in L_{[\kappa:\mu]}$ of exact order 4 on the curve $L_{[\kappa:\mu]}$, and the evaluation coefficients c_0, c_1, c_2 for the isogeny $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$ with $\ker(\varphi) = \langle 2P_4 \rangle$. The point P_4 here can be seen as lying in the upper right corner of the strategy S (see Section 1). Since $\ker(\varphi) = \langle 2P_4 \rangle$, we have that $\varphi(P_4) \in \{[0:0:1],[1:0:1]\}$. Algorithm 5 partially evaluates P_4 under φ and checks whether $X_{\varphi(P_4)}$ is zero. A bit variable b' is assigned 0 if $X_{\varphi(P_4)}$ is zero and 1 otherwise. The values of $[A_{24}^+ : C_{24}]$ are then computed according to Theorem 6, where $b = 0$ indicates using the formula for M_0 and $b = 1$ indicates using that of M_1. The evaluation coefficient C is also computed, whose value is either $\kappa' - \mu'$ when $b = 0$ or κ' when $b = 1$. Again, conditional swaps are used based on b.

*Note that we use the command $\text{cswap}(\alpha, \beta, b)$, which swaps the values of α and β conditioned on $b = 1$.

Algorithm 6 evaluates $\varphi' : L_{[\kappa':\mu']} \to M_{\mathfrak{b},\mathfrak{a}}$ at a given point. It receives as input the coordinates $(X_Q : Z_Q)$ of a point $Q \in L_{[\kappa':\mu']}$, the quantity C from Algorithm 5 for evaluating φ', and the bit variable b' for which $\ker(\varphi') = \langle [b' : 0 : 1] \rangle$. The output is the point $(X_{\varphi'(Q)} : Z_{\varphi'(Q)})$, computed according to Theorem 6.

3.4 Algorithm Costs

Table 1 summarizes the computational costs of the subroutines Algorithms 1–6. The costs are given in terms of **m**, **s**, and **i**, which denote the costs of multiplication, squaring, and inversion in $\mathbb{F}_{p^2}$. We assume the costs of addition, negation, flipping bits, and the cswap function are negligible.

Alg.	Function	Cost
1	`2IsoLegCurve`	$5\mathbf{m}$
2	`2IsoLegEval`	$4\mathbf{m} + 3\mathbf{s}$
3	`coordinateDBL`	$5\mathbf{m} + 4\mathbf{s}$
4	`jInv`	$8\mathbf{m} + 6\mathbf{s} + \mathbf{i}$
5	`2IsoMontCurve`	$4\mathbf{m} + 2\mathbf{s}$
6	`2IsogMontEval`	$5\mathbf{m}$

Table 1: Computational costs of each algorithm in terms of field arithmetic costs. Here, **m**, **s**, and **i** denote the costs of multiplication, squaring, and inversion in $\mathbb{F}_{p^2}$, respectively. The costs of addition and negation are considered to be negligible and are not reported in the cost values.

4 Conclusion

Future Work. Here we expand on our initial motivation for using Legendre curves for isogeny-based cryptography. As explained in Section 1, at a low level the SIDH framework constructs an ℓ^e-isogeny φ as a sequence of ℓ-isogenies $\varphi_i : E_i \to E_{i+1}$ with $\ker(\varphi_i) = \langle R_i \rangle$. The points R_i are obtained through the laborious task of strategy traversal and are only needed in order to construct φ_i. When $\ell = 2$, each R_i is a nontrivial 2-torsion point on the curve E_i. If E_i is a Legendre curve L_{λ_i}, then $L_{\lambda_i}[2] = \{\mathcal{O}, (0,0), (1,0), (\lambda,0)\}$ and so R_i has a particularly simple form.

The appeal for using Legendre curves is that since the 2-torsion is known in advance and is identical for all curves, one may consider an alternative approach for constructing the isogeny φ. If each φ_i is constructed so that R_i is chosen as $(0,0)$ or $(1,0)$, Section 2 showed that $(\lambda, 0)$ will correspond to the dual isogeny; in other words, the value of each R_i corresponds to some bit β_i, where $R_i = (\beta_i, 0)$. In this way, one could use some bitstring $(\beta_1, \ldots, \beta_e)$ and *choose* to use the value of $R_i = (\beta_i, 0)$ for each kernel generator rather than derive R_i through strategy traversal. This would have the effect of skipping strategy traversal altogether—improving

efficiency by a $O(\log d)$ factor where d is the tree depth—since all points R_i are known from the start.

The drawback to this approach (as shown in Section 2) is that while applying Velu's formulas to L_{λ_i} and $(\beta_i, 0)$ is easy and results in an efficiently computable isogeny $\varphi_i : L_{\lambda_i} \to E_{i+1}$, it is rarely the case that E_{i+1} is also in Legendre form and so this efficiency is lost when attempting to make the approach iterable. In particular, the isomorphism which puts E_{i+1} in Legendre form requires computing a square root, which is relatively costly. It is true that for sufficient tree depth d, avoiding the strategy traversal would provide improvements in efficiency even after accounting for the square root computation. The main difficulty is that, for isogenies of Vélu type, the order of the kernel generator (that is, the degree of the isogeny) is always less than the group order, which is typically not much greater than p, and under these conditions the tree depth will never be large enough that trading off strategy traversal for square root computations saves time. Consequently, Legendre form would outperform the strategy traversal approach only if we can find a way to avoid the square root computation.

Concluding Remarks. In this work, we detailed many results regarding Legendre curves, including point doubling formulas, the 4-torsion structure, and formulas for 2-isogenies between Legendre curves. Furthermore, we used these theoretical results to devise algorithms that construct 2^a-isogenies between Legendre curves. The work has applications in isogeny-based cryptography where 2^a-isogeny computation is required, in particular the CGL hash function [3, 4].

References

[1] R. Azarderakhsh, E. Bakos Lang, D. Jao, and B. Koziel. Edsidh: Supersingular isogeny diffie-hellman key exchange on edwards curves. In Anupam Chattopadhyay, Chester Rebeiro, and Yuval Yarom, editors, *Security, Privacy, and Applied Cryptography Engineering*, pages 125–141, Cham, 2018. Springer International Publishing.

[2] W. Castryck and T. Decru. An efficient key recovery attack on sidh. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023*, pages 423–447, Cham, 2023. Springer Nature Switzerland.

[3] D. X. Charles, E. Z. Goren, and K. E. Lauter. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22:93–113, 2009.

[4] J. Doliskani, G. C. C. F. Pereira, and P. S. L. M. Barreto. Faster cryptographic hash function from supersingular isogeny graphs. 2022.

[5] S. D. Galbraith, C. Petit, B. Shani, and Y. B. Ti. On the security of supersingular isogeny cryptosystems. *Advances in Cryptology – ASIACRYPT*, page 63–91, 2016.

[6] D. Jao, R. Azarderakhsh, M. Campagna, C. Costello, L. De Feo, B. Hess, A. Hutchinson, A. Jalali, K. Karabina, B. Koziel, B. LaMacchia, P. Longa, M. Naehrig, G. Pereira, J. Renes, V. Soukharev, and D. Urbanik. Supersingular isogeny key encapsulation, 2020.

[7] D. Jao and L. De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In Bo-Yin Yang, editor, *Post-Quantum Cryptography*, pages 19–34, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[8] D. Jao, L. De Feo, and J. Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.

[9] J. H. Silverman. *The Arithmetic of Elliptic Curves*. Graduate Texts in Mathematics. Springer, 2nd edition, 2008.

[10] J. Vélu. Isogénies entre courbes elliptiques. *C. R. Acad. Sci., Paris, Sér. A*, 273:238–241, 1971.

[11] L. C. Washington. *Elliptic Curves: Number Theory and Cryptography*. CRC press, 2008.

References

[1] R. Azarderakhsh, E. Bakos Lang, D. Jao, and B. Koziel. Edsidh: Supersingular isogeny diffie-hellman key exchange on edwards curves. In Anupam Chattopadhyay, Chester Rebeiro, and Yuval Yarom, editors, *Security, Privacy, and Applied Cryptography Engineering*, pages 125–141, Cham, 2018. Springer International Publishing.

[2] Chandrajit L. Bajaj. Some applications of constructive real algebraic geometry. In *Algebraic Geometry and its Applications: Collections of Papers from Shreeram S. Abhyankar's 60th Birthday Conference*, pages 393–405. Springer New York, New York, NY, 1994.

[3] B. Bank, M. Giusti, J. Heintz, and G.-M. Mbakop. Polar varieties and efficient real equation solving: The hypersurface case. *Journal of Complexity*, 13(1):5–27, 1997.

[4] B. Bank, M. Giusti, J. Heintz, and G.-M. Mbakop. Polar varieties and efficient real elimination. *Mathematische Zeitschrift*, 238(1):115–144, 2001.

[5] Andrea Basso, Giulio Codogni, Deirdre Connolly, Luca De Feo, Tako Boris Fouotsa, Guido Maria Lido, Travis Morrison, Lorenz Panny, Sikhar Patranabis, and Benjamin Wesolowski. Supersingular curves you can trust. In *Advances in Cryptology – EUROCRYPT 2023: 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23–27, 2023, Proceedings, Part II*, page 405–437, Berlin, Heidelberg, 2023. Springer-Verlag.

[6] S. Basu, R. Pollack, and M.-F. Roy. *Algorithms in Real Algebraic Geometry*, volume 10 of *Algorithms and computation in mathematics*. Springer-Verlag, 2003.

[7] J. Böhm, W. Decker, C. Fieker, and G. Pfister. The use of bad primes in rational reconstruction. *Math. Comput.*, 84:3013–3027, 2012.

[8] John Canny. *The complexity of robot motion planning*. MIT Press, 1988.

[9] Jose Capco, Mohab Safey El Din, and Josef Schicho. Robots, computer algebra and eight connected components. In *Proceedings of the 45th International Symposium on Symbolic and Algebraic Computation*, ISSAC '20, page 62–69, New York, NY, USA, 2020. Association for Computing Machinery.

[10] W. Castryck and T. Decru. An efficient key recovery attack on sidh. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology – EUROCRYPT 2023*, pages 423–447, Cham, 2023. Springer Nature Switzerland.

[11] Damien Chablat, Ranjan Jha, Fabrice Rouillier, and Guillaume Moroz. Workspace and joint space analysis of the 3-RPS parallel robot. In *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, volume 46360. American Society of Mechanical Engineers, 2014.

[12] Damien Chablat, Rémi Prébet, Mohab Safey El Din, Durgesh H. Salunkhe, and Philippe Wenger. Deciding cuspidality of manipulators through computer algebra and algorithms in real algebraic geometry. In *Proceedings of the 2022 International Symposium on Symbolic and Algebraic Computation*, ISSAC '22, page 439–448, New York, NY, USA, 2022. Association for Computing Machinery.

[13] D. X. Charles, E. Z. Goren, and K. E. Lauter. Cryptographic hash functions from expander graphs. *Journal of Cryptology*, 22:93–113, 2009.

[14] David A. Cox, John Little, and Donal O'Shea. *Ideals, Varieties, and Algorithms*. Undergraduate Texts in Mathematics. Springer Cham, 4 edition, 2015. Published: 13 May 2015.

[15] Xavier Dahan, Abdulilah Kadri, and Éric Schost. Bit-size estimates for triangular sets in positive dimension. *Journal of Complexity*, 28(1):109–135, 2012.

[16] Carlos D'Andrea, Teresa Krick, and Martín Sombra. Heights of varieties in multiprojective spaces and arithmetic nullstellensatz. *Annales scientifiques de l'École Normale Supérieure*, 46(4):549–627, August 2013.

[17] J. Doliskani, G. C. C. F. Pereira, and P. S. L. M. Barreto. Faster cryptographic hash function from supersingular isogeny graphs. In *Proceedings of the Selected Areas in Cryptography (SAC)*, 2022.

[18] Clémence Durvye and Grégoire Lecerf. A concise proof of the kronecker polynomial system solver from scratch. *Expositiones Mathematicae*, 26(2):101–139, 2008.

[19] C. D'Andrea, A. Ostafe, I. Shparlinski, and M. Sombra. Reductions modulo primes of systems of polynomial equations and algebraic dynamical systems. *Trans. Amer. Math. Soc.*, 371(2):1169–1198, 2019.

[20] D. Eisenbud. *Commutative Algebra with a View Toward Algebraic Geometry*, volume 150 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1st. edition, 1995.

[21] J. Elliott, M. Giesbrecht, and É. Schost. On the bit complexity of finding points in connected components of a smooth real hypersurface. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation (ISSAC)*, pages 170–177, 2020.

[22] Jesse Elliott. Randomized algorithms in real algebraic geometry. Master's thesis, University of Waterloo, 2020. http://hdl.handle.net/10012/16236.

[23] Jesse Elliott, Mark Giesbrecht, and Éric Schost. Bit complexity for computing one point in each connected component of a smooth real algebraic set. *Journal of Symbolic Computation*, 116:72–97, 2023.

[24] M. England and J. H. Davenport. The complexity of cylindrical algebraic decomposition with respect to polynomial degree. In *Proceedings of the 18th International Workshop on Computer Algebra in Scientific Computing (CASC)*, pages 172–192, 2016.

[25] David Harvey and Joris van der Hoeven. Integer multiplication in time $o(n \log n)$. *Annals of Mathematics*, 193(2):563–617, 2021. https://doi.org/10.4007/annals.2021.193.2.4.

[26] J. Heintz and C.P. Schnorr. Testing polynomials which are easy to compute. *STOC '80: Proceedings of the twelfth annual ACM symposium on Theory of computing*, 1980.

[27] D. Jao, R. Azarderakhsh, M. Campagna, C. Costello, L. De Feo, B. Hess, A. Hutchinson, A. Jalali, K. Karabina, B. Koziel, B. LaMacchia, P. Longa, M. Naehrig, G. Pereira, J. Renes, V. Soukharev, and D. Urbanik. Supersingular isogeny key encapsulation. https://sike.org/files/SIDH-spec.pdf, 2020.

[28] D. Jao and L. De Feo. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. In Bo-Yin Yang, editor, *Post-Quantum Cryptography*, pages 19–34, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.

[29] D. Jao, L. De Feo, and J. Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.

[30] T. Krick, L.-M. Pardo, and M. Sombra. Sharp estimates for the arithmetic nullstellensatz. *Duke Mathematical Journal*, 109(3):521–598, 2001.

[31] Daniel Lazard and Fabrice Rouillier. Solving parametric polynomial systems. *Journal of Symbolic Computation*, 42(6):636–667, 2007.

[32] H. F. Mattson, Jr. The theory of error-correcting codes (f. j. macwilliams and n. j. a. sloane). *SIAM Review*, 22(4):513–519, 1980.

[33] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, USA, 10th edition, 2011.

[34] C. Pernet. *High Performance and Reliable Algebraic Computing*. HDR, Université Joseph Fourier, Grenoble 1, November 2014.

[35] R. Piene. Polar classes of singular varieties. *Annales Scientifiques de l'École Normale Supérieure*, 11:247–276, 1978.

[36] Rémi Prébet, Mohab Safey El Din, and Éric Schost. Computing roadmaps in unbounded smooth real algebraic sets ii: algorithm and complexity, 2024.

[37] Mihai Putinar. Jean bernard lasserre: Moments, positive polynomials and their applications. *Foundations of Computational Mathematics*, 11(4):489–497, 2011.

[38] Alban Quadrat and Eva Zerz, editors. *Algebraic and Symbolic Computation Methods in Dynamical Systems*, volume 9 of *Advances in Delays and Dynamics*. Springer, 2020.

[39] É. Schost and M. Safey El Din. Polar varieties and computation of one point in each connected component of a smooth real algebraic set. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation (ISSAC)*, pages 224–231, 2003.

[40] E. Schost and M. Safey El Din. A nearly optimal algorithm for deciding connectivity queries in smooth and bounded real algebraic sets. *Journal of the ACM*, 63(6):1–48, February 2017.

[41] É. Schost and M. Safey El Din. Bit complexity for multi-homogeneous system solving application to polynomial minimization. *Journal of Symbolic Computation*, 87:176–206, May 2018.

[42] Eric Schost and Mohab Safey El Din. A baby steps/giant steps probabilistic algorithm for computing roadmaps in smooth bounded real hypersurfaces. *Discrete and Computational Geometry*, 5:181–220, 2011.

[43] Abraham Seidenberg. A new decision method for elementary algebra. *Annals of Mathematics*, 60:365–374, 1954.

[44] Igor Shafarevich. *Basic Algebraic Geometry 1*. Springer Verlag, 1977.

[45] J. H. Silverman. *The Arithmetic of Elliptic Curves*. Graduate Texts in Mathematics. Springer, 2nd edition, 2008.

[46] Alfred Tarski. A decision method for elementary algebra and geometry. In Bob F. Caviness and Jeremy R. Johnson, editors, *Quantifier Elimination and Cylindrical Algebraic Decomposition*, pages 24–84, Vienna, 1998. Springer Vienna.

[47] B. Teissier. Quelques points de l'histoire des variétés polaires, de poncelet à nos jours. *Séminaire Annales Univ. Blaise Pascal*, 4:247–276, 1988.

[48] Lloyd N. Trefethen and David Bau. *Numerical Linear Algebra, Twenty-fifth Anniversary Edition*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2022.

[49] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, third edition, 2013.

[50] L. C. Washington. *Elliptic Curves: Number Theory and Cryptography.* CRC press, 2008.

Appendix A

Pseudocode

Appendix A Pseudocode

Algorithm 1: Computing a 2-isogenous Legendre curve

Function: `2IsoLegCurve`

Input: *Curve coefficients* $(\kappa : \mu)$ *defining a curve* $L_{[\kappa:\mu]}$*;a point* $(X_{P_4} : Z_{P_4}) \in L_{[\kappa:\mu]}$ *with exact order* 4.

Output: *Curve coefficients* $(\kappa' : \mu')$ *defining a curve* $L_{[\kappa':\mu']}$*; evaluation coefficients* c_0, c_1, c_2 *for an isogeny* $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$*; a bit variable* b *satisfying* $\ker(\varphi) = \langle [b : 0 : 1] \rangle$.

$t_0 \leftarrow \mu \cdot X_{P_4}$

$t_1 \leftarrow \kappa \cdot Z_{P_4}$

$t_2 \leftarrow \mu \cdot Z_{P_4}$

$s_0 \leftarrow t_0 \cdot X_{P_4}$ // $s_0 = \mu X_{P_4}^2$

$s_1 \leftarrow t_1 \cdot Z_{P_4}$ // $s_1 = \kappa Z_{P_4}^2$

$b \leftarrow s_0 \overset{?}{\neq} s_1$ // $b = 0$ `(Theorem 4)`; $b = 1$ `(Theorem 5)`

// κ'

$r_0 \leftarrow -t_2$ // $r_0 = -\mu Z_{P_4}$

$r_0 \leftarrow r_0 + r_0$ // $r_0 = -2\mu Z_{P_4}$

$r_0 \leftarrow r_0 + r_0$ // $r_0 = -4\mu Z_{P_4}$

$\mathrm{cswap}(r_0, t_2, \overline{b})$ // $b = 0$: $r_0 = \mu Z_{P_4}$; $b = 1$: $r_0 = -4\mu Z_{P_4}$

$\kappa' \leftarrow r_0$ // $b = 0$: $\kappa' = \mu Z_{P_4}$; $b = 1$: $\kappa' = -4\mu Z_{P_4}$

$\kappa' \leftarrow \kappa' + t_1$ // $b = 0$: $\kappa' = \kappa Z_{P_4} + \mu Z_{P_4}$; $b = 1$: $\kappa' = \kappa Z_{P_4} - 4\mu Z_{P_4}$

$r_1 \leftarrow t_0 + t_0$ // $r_1 = 2\mu X_{P_4}$

$\kappa' \leftarrow \kappa' + r_1$ // $b = 0$: $\kappa' = \kappa Z_{P_4} + 2\mu X_{P_4} + \mu Z_{P_4}$; $b = 1$: $\kappa' = \kappa Z_{P_4} + 2\mu X_{P_4} - 4\mu Z_{P_4}$

// μ'

$r_2 \leftarrow t_0$ // $r_2 = \mu X_{P_4}$

$r_2 \leftarrow t_0 + t_0$ // $r_2 = 2\mu X_{P_4}$

$r_2 \leftarrow t_0 + t_0$ // $r_2 = 4\mu X_{P_4}$

$\mathrm{cswap}(r_0, 0, \overline{b})$ // $b = 0$: $r_0 = 0$; $b = 1$: $r_0 = -4\mu Z_{P_4}$

$r_3 \leftarrow r_2 + r_0$ // $b = 0$: $r_3 = 4\mu X_{P_4}$; $b = 1$: $r_3 = 4\mu X_{P_4} - 4\mu Z_{P_4}$

$\mu' \leftarrow r_3$ // $b = 0$: $\mu' = 4\mu X_{P_4}$; $b = 1$: $\mu' = 4\mu X_{P_4} - 4\mu Z_{P_4}$

// c_0, c_1, c_2 `(coefficients for` X'`)`

$c_0 \leftarrow t_2$ // $c_0 = \mu Z_{P_4}$

$r_4 \leftarrow -t_1$ // $r_4 = -\kappa Z_{P_4}$

$r_4 \leftarrow r_4 + \kappa'$ // $r_4 = \kappa' - \kappa Z_{P_4}$

$r_2 \leftarrow t_0$ // $r_2 = \mu X_{P_4}$

$r_2 \leftarrow r_2 + t_0$ // $r_2 = 2\mu X_{P_4}$

$\mathrm{cswap}(r_4, r_2, \overline{b})$ // $b = 0$: $r_4 = 2\mu X_{P_4}$; $b = 1$: $r_4 = \kappa' - \kappa Z_{P_4}$

$c_1 \leftarrow r_4$ // $b = 0$: $c_1 = 2\mu X_{P_4}$; $b = 1$: $c_1 = \kappa' - \kappa Z_{P_4}$

$r_5 \leftarrow -\kappa'$ // $r_5 = -\kappa'$

$\mathrm{cswap}(r_5, t_1, \overline{b})$ // $b = 0$: $r_5 = \kappa Z_{P_4}$; $b = 1$: $r_5 = -\kappa'$

$c_2 \leftarrow r_5$ // $b = 0$: $c_2 = \kappa Z_{P_4}$; $b = 1$: $c_2 = -\kappa'$

Return $(\kappa' : \mu'), c_0, c_1, c_2, b$

Algorithm 2: Evaluating a 2-isogenous Legendre curve at a point

Function: `2IsoLegEval`

Input: *A point* $(X_Q : Z_Q)$ *on some curve* $L_{[\kappa:\mu]}$*; evaluation coefficients* c_0, c_1, c_2 *for an isogeny* $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$*; the codomain curve coefficient* μ'*; a bit variable* b *satisfying* $\ker(\varphi) = \langle [b : 0 : 1] \rangle$.

Output: *A point* $(X_{Q'} : Z_{Q'})$*, where* $\varphi(Q) = Q'$.

$t_0 \leftarrow X_Q^2$

$t_1 \leftarrow (X_Q + Z_Q)^2$ // $t_1 = X_Q^2 + 2X_QZ_Q + Z_Q^2$

$t_2 \leftarrow Z_Q^2$

$t_1 \leftarrow t_1 - t_0$ // $t_1 = 2X_QZ_Q + Z_Q^2$

$t_1 \leftarrow t_1 - t_2$ // $t_1 = 2X_QZ_Q$

$t_1 \leftarrow t_1/2$ // $t_1 = X_QZ_Q$

// $X_{Q'}$

$X_{Q'} \leftarrow c_0 \cdot t_0$ // $X_{Q'} = c_0X_Q^2$

$s_0 \leftarrow c_1 \cdot t_1$ // $s_0 = c_1X_QZ_Q$

$X_{Q'} \leftarrow X_{Q'} + s_0$ // $X' = c_0X_Q^2 + c_1X_QZ_Q$

$s_1 \leftarrow c_2 \cdot t_2$ // $s_1 = c_2Z_Q^2$

$X_{Q'} \leftarrow X_{Q'} + s_1$ // $X_{Q'} = c_0X_Q^2 + c_1X_QZ_Q + c_2Z_Q^2$

// $Z_{Q'}$

$r_0 \leftarrow t_1$ // $r_0 = X_QZ_Q$

$r_0 \leftarrow r_0 - t_2$ // $r_0 = X_QZ_Q - Z_Q^2$

$\mathrm{cswap}(r_0, t_1, \overline{b})$ // $b = 0$: $r_0 = X_QZ_Q$; $b = 1$: $r_0 = X_QZ_Q - Z_Q^2$

$Z_{Q'} \leftarrow \mu' \cdot r_0$ // $b = 0$: $Z_{Q'} = \mu' X_QZ_Q$; $b = 1$: $Z_{Q'} = \mu'(X_QZ_Q - Z_Q^2)$

Return $(X_{Q'} : Z_{Q'})$

Algorithm 3: Coordinate doubling

Function: `coordinateDBL`

Input: *Curve coefficients* $(\kappa : \mu)$ *defining a curve* $L_{[\kappa:\mu]}$*; a point* $(X_P : Z_P) \in L_{[\kappa:\mu]}$.

Output: *A point* $(X_{[2]P} : Z_{[2]P})$*, the doubling of* P.

1	$t_0 \leftarrow X_P^2$	
2	$t_1 \leftarrow \mu \cdot t_0$	// $t_1 = \mu X_P^2$
3	$t_2 \leftarrow Z_P^2$	
4	$t_3 \leftarrow \kappa \cdot t_2$	// $t_3 = \kappa Z_P^2$
		// $X_{[2]P}$
5	$X_{[2]P} \leftarrow t_1 - t_3$	// $X_{[2]P} = \mu X_P^2 - \kappa Z_P^2$
6	$X_{[2]P} \leftarrow X_{[2]P}^2$	// $X_{[2]P} = (\mu X_P^2 - \kappa Z_P^2)^2$
		// $Z_{[2]P}$
7	$r_0 \leftarrow X_P + Z_P$	
8	$r_0 \leftarrow r_0^2$	// $r_0 = (X_P + Z_P)^2 = X_P^2 + Z_P^2 + 2X_P Z_P$
9	$r_0 \leftarrow r_0 - t_0$	// $r_0 = Z_P^2 + 2X_P Z_P$
10	$r_0 \leftarrow r_0 - t_2$	// $r_0 = 2X_P Z_P$
11	$r_0 \leftarrow r_0/2$	// $r_0 = X_P Z_P$
12	$r_1 \leftarrow \kappa + \mu$	
13	$r_1 \leftarrow r_0 \cdot r_1$	// $r_1 = (\kappa + \mu) X_P Z_P$
14	$Z_{[2]P} \leftarrow t_1 - r_1$	// $Z_{[2]P} = \mu X_P^2 - (\kappa + \mu) X_P Z_P$
15	$Z_{[2]P} \leftarrow Z_{[2]P} + t_3$	// $Z_{[2]P} = \mu X_P^2 - (\kappa + \mu) X_P Z_P + \kappa Z_P^2$
16	$Z_{[2]P} \leftarrow r_0 \cdot Z_{[2]P}$	// $Z_{[2]P} = X_P Z_P (\mu X_P^2 - (\kappa + \mu) X_P Z_P + \kappa Z_P^2)$
17	$Z_{[2]P} \leftarrow \mu \cdot Z_{[2]P}$	// $Z_{[2]P} = \mu X_P Z_P (\mu X_P^2 - (\kappa + \mu) X_P Z_P + \kappa Z_P^2)$
18	$Z_{[2]P} \leftarrow Z_{[2]P} + Z_{[2]P}$	// $Z_{[2]P} = 2\mu X_P Z_P (\mu X_P^2 - (\kappa + \mu) X_P Z_P + \kappa Z_P^2)$
19	$Z_{[2]P} \leftarrow Z_{[2]P} + Z_{[2]P}$	// $Z_{[2]P} = 4\mu X_P Z_P (\mu X_P^2 - (\kappa + \mu) X_P Z_P + \kappa Z_P^2)$
20	**Return** $(X_{[2]P} : Z_{[2]P})$	

Algorithm 4: j-invariant computation

Function: `jInv`

Input: *Curve coefficients* $(\kappa' : \mu') \in \mathbb{P}^1$*; a point* $(X_{P_4} : Z_{P_4}) \in L_{[\kappa:\mu]}$ *where* P_4 *has exact order* 4 *on some curve* $L_{[\kappa:\mu]}$*; evaluation coefficients* c_0, c_1, c_2 *for an isogeny* $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$ *with* $\ker(\varphi) = \langle 2P_4 \rangle$.

Output: *The* j*-invariant* $j(L_{[\kappa'':\mu'']})$ *where* $L_{[\kappa'':\mu'']}$ *is the codomain of the isogeny* $\varphi' : L_{[\kappa':\mu']} \to L_{[\kappa'':\mu'']}$ *whose kernel is generated by* P_4 *with* $\ker(\varphi') = \langle \varphi(P_4) \rangle$.

$t_0 \leftarrow X_{P_4}^2$

$t_1 \leftarrow X_{P_4} \cdot Z_{P_4}$

$t_2 \leftarrow Z_{P_4}^2$

// X'_{P_4}

$X_{P_4'} \leftarrow c_0 \cdot t_0$ // $X_{P_4'} = c_0 X_{P_4}^2$

$s_0 \leftarrow c_1 \cdot t_1$ // $s_0 = c_1 X_{P_4} Z_{P_4}$

$X_{P_4'} \leftarrow X_{P_4'} + s_0$ // $X' = c_0 X_{P_4}^2 + c_1 X_{P_4} Z_{P_4}$

$s_1 \leftarrow c_2 \cdot t_2$ // $s_1 = c_2 Z_{P_4}^2$

$X_{P_4'} \leftarrow X_{P_4'} + s_1$ // $X_{P_4'} = c_0 X_{P_4}^2 + c_1 X_{P_4} Z_{P_4} + c_2 Z_{P_4}^2$

$b' \leftarrow X_{P_4'} \overset{?}{\neq} 0$ // $b' = 0$ `if` $X_{P_4'} = 0$; $b' = 1$ `otherwise`

// j-`invariant numerator`

$t_0 \leftarrow \kappa'$

$t_1 \leftarrow t_0^2$ // $t_1 = \kappa'^2$

$t_2 \leftarrow \mu'$

$t_3 \leftarrow t_2^2$ // $t_3 = \mu'^2$

$s_0 \leftarrow t_3$ // $s_0 = \mu'^2$

$s_0 \leftarrow s_0 + s_0$ // $s_0 = 2\mu'^2$

$s_0 \leftarrow s_0 + s_0$ // $s_0 = 4\mu'^2$

$s_0 \leftarrow s_0 + s_0$ // $s_0 = 8\mu'^2$

$s_0 \leftarrow s_0 + s_0$ // $s_0 = 16\mu'^2$

$\mathrm{cswap}(t_3, s_0, b')$ // $b' = 0$: $t_3 = \mu'^2$; $b' = 1$: $t_3 = 16\mu'^2$

$r_0 \leftarrow \kappa \cdot \mu'$

$r_1 \leftarrow r_0 + r_0$ // $r_1 = 2\kappa'\mu'$

$r_1 \leftarrow r_1 + r_1$ // $r_1 = 4\kappa'\mu'$

$r_1 \leftarrow r_1 + r_1$ // $r_1 = 8\kappa'\mu'$

$r_1 \leftarrow r_1 + r_1$ // $r_1 = 16\kappa'\mu'$

$r_2 \leftarrow -r_1$ // $r_2 = -16\kappa'\mu'$

$r_3 \leftarrow r_1 - r_0$ // $r_3 = 15\kappa'\mu'$

$r_3 \leftarrow r_3 - r_0$ // $r_3 = 14\kappa'\mu'$

$\mathrm{cswap}(r_3, r_2, b')$ // $b' = 0$: $r_3 = 14\kappa'\mu'$; $b' = 1$: $r_3 = -16\kappa'\mu'$

$N \leftarrow t_1$ // $N = \kappa^2$

$N \leftarrow N + r_3$ // $b' = 0$: $N = \kappa'^2 + 14\kappa'\mu'$; $b' = 1$: $N = \kappa'^2 - 16\kappa'\mu'$

$N \leftarrow N + t_3$ // $b' = 0$: $N = \kappa'^2 + 14\kappa'\mu' + \mu'^2$; $b' = 1$: $N = \kappa'^2 - 16\kappa'\mu' + 16\mu'^2$

// j-`invariant denominator`

$D_0 \leftarrow r_0$ // $D_0 = \kappa'\mu'$

$D_1 \leftarrow \kappa' - \mu'$ $D_0 \leftarrow D_0 \cdot D_1$ // $D_0 = \kappa'\mu'(\kappa' - \mu')$

$D_0 \leftarrow D_0^2$ // $D_0 = (\kappa'\mu'(\kappa' - \mu'))^2$

$j \leftarrow 1/D_0$ // $j = \frac{1}{(\kappa'\mu'(\kappa'-\mu'))^2}$

$j \leftarrow j \cdot N$ // $b' = 0$: $j = \frac{(\kappa'^2+14\kappa'\mu'+\mu'^2)^3}{(\kappa'\mu'(\kappa'-\mu'))^2}$; $b' = 1$: $j = \frac{(\kappa'^2-16\kappa'\mu'+16\mu'^2)^3}{(\kappa'\mu'(\kappa'-\mu'))^2}$

Return j

Algorithm 5: Computing a 2-isogenous Montgomery curve from a Legendre curve

Function: `2IsoMontCurve`

Input: *Curve coefficients* $(\kappa' : \mu') \in \mathbb{P}^1$*; a point* $(X_{P_4} : Z_{P_4}) \in L_{[\kappa:\mu]}$ *where* P_4 *has exact order* 4 *on some curve* $L_{[\kappa:\mu]}$*; evaluation coefficients* c_0, c_1, c_2 *for an isogeny* $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$ *with* $\ker(\varphi) = \langle 2P_4 \rangle$.

Output: *Curve coefficients* $[A_{24}^+ : C_{24}]$ *for a Montgomery curve* $M_{\mathfrak{b},\mathfrak{a}}$ *satisfying* $4A_{24}^+ = C_{24}(A+2)$*; an isogeny evaluation coefficient* C *for an isogeny* $\varphi' : L_{[\kappa':\mu']} \to M_{\mathfrak{b},\mathfrak{a}}$ *with* $\ker(\varphi') = \langle \varphi(P_4) \rangle$*; a bit variable* b' *satisfying* $\ker(\varphi') = \langle [b' : 0 : 1] \rangle$.

$t_0 \leftarrow X_{P_4}^2$
$t_1 \leftarrow X_{P_4} \cdot Z_{P_4}$
$t_2 \leftarrow Z_{P_4}^2$
// X'_{P_4}
$X_{P_4'} \leftarrow c_0 \cdot t_0$ // $X_{P_4'} = c_0 X_{P_4}^2$
$s_0 \leftarrow c_1 \cdot t_1$ // $s_0 = c_1 X_{P_4} Z_{P_4}$
$X_{P_4'} \leftarrow X_{P_4'} + s_0$ // $X' = c_0 X_{P_4}^2 + c_1 X_{P_4} Z_{P_4}$
$s_1 \leftarrow c_2 \cdot t_2$ // $s_1 = c_2 Z_{P_4}^2$
$X_{P_4'} \leftarrow X_{P_4'} + s_1$ // $X_{P_4'} = c_0 X_{P_4}^2 + c_1 X_{P_4} Z_{P_4} + c_2 Z_{P_4}^2$
$b' \leftarrow X_{P_4'} \stackrel{?}{\neq} 0$ // $b' = 0$ `if` $X_{P_4'} = 0$; $b' = 1$ `otherwise`
// C
$t_0 \leftarrow \kappa'$
$C \leftarrow t_0 - \mu'$ // $C = \kappa' - \mu'$
$\mathrm{cswap}(C, t_0, b')$ // $b' = 0$: $C = \kappa' - \mu'$; $b' = 1$: $C = \kappa'$
// A
$s_0 \leftarrow \kappa'$
$s_1 \leftarrow \mu'$
$s_2 \leftarrow -\mu'$
$s_2 \leftarrow s_2 + s_2$ // $s_2 = -2\mu'$
$\mathrm{cswap}(s_1, s_2, b')$ // $b' = 0$: $s_1 = \mu'$; $b' = 1$: $s_1 = -2\mu'$
$A' \leftarrow s_1$ // $b' = 0$: $A' = \mu'$; $b' = 1$: $A' = -2\mu'$
$A' \leftarrow A' + \kappa'$ // $b' = 0$: $A' = \kappa' + \mu'$; $b' = 1$: $A' = \kappa' - 2\mu'$
$A' \leftarrow A' + A'$ // $b' = 0$: $A' = 2(\kappa' + \mu')$; $b' = 1$: $A' = 2(\kappa' - 2\mu')$
// A_{24}^+
$A_{24}^+ \leftarrow C$ // $A_{24}^+ = C$
$A_{24}^+ \leftarrow A_{24}^+ + C$ // $A_{24}^+ = 2C$
$A_{24}^+ \leftarrow A_{24}^+ + A$ // $A_{24}^+ = A + 2C$
// C_{24}
$C_{24} \leftarrow C$ // $C_{24} = C$
$C_{24} \leftarrow C_{24} + C_{24}$ // $C_{24} = 2C$
$C_{24} \leftarrow C_{24} + C_{24}$ // $C_{24} = 4C$
Return A_{24}^+, C_{24}, C, b'

Algorithm 6: Evaluating $\varphi' : L_{[\kappa':\mu']} \to M_{\mathfrak{b},\mathfrak{a}}$ at a point

Function: `2IsogMontEval`

Input: *Curve coefficients* $(\kappa' : \mu')$*; an evaluation coefficient* C *for an isogeny* $\varphi' : L_{[\kappa':\mu']} \to M_{\mathfrak{b},\mathfrak{a}}$*; a point* $Q = (X_Q : Z_Q) \in L_{[\kappa':\mu']}$ *to be evaluated; a bit variable* b' *satisfying* $\ker(\varphi') = \langle [b' : 0 : 1] \rangle$.

Output: $(X_{Q'} : Z_{Q'})$*, where* $Q' = \varphi'(Q)$.

		// $X_{Q'}$
1	$t_0 \leftarrow X_Q$	
2	$t_1 \leftarrow t_0 - Z_Q$	// $t_1 = X_Q - Z_Q$
3	$\mathrm{cswap}(t_1, t_0, b')$	// $b' = 0$: $t_1 = X_Q - Z_Q$; $b' = 1$: $t_1 = X_Q$
4	$s_0 \leftarrow \mu' \cdot X_Q$	
5	$s_1 \leftarrow \kappa' \cdot Z_Q$	
6	$s_2 \leftarrow s_0 - s_1$	// $s_2 = \mu' X_Q - \kappa' Z_Q$
7	$X_{Q'} \leftarrow t_1 \cdot s_2$	// $b' = 0$: $X_{Q'} = (X_Q - Z_Q)(\mu' X_Q - \kappa' Z_Q)$; $b' = 1$: $X_{Q'} = X_Q(\mu' X_Q - \kappa' Z_Q)$
		// $Z_{Q'}$
8	$r_0 \leftarrow X_Q$	
9	$t_1 \leftarrow r_0 - Z_Q$	// $r_1 = X_Q - Z_Q$
10	$\mathrm{cswap}(r_0, r_1, b')$	// $b' = 0$: $r_0 = X_Q$; $b' = 1$: $r_0 = X_Q - Z_Q$
11	$Z_{Q'} \leftarrow r_0$	// $b' = 0$: $Z_{Q'} = X_Q$; $b' = 1$: $Z_{Q'} = X_Q - Z_Q$
12	$Z_{Q'} \leftarrow r_0 \cdot Z_Q$	// $b' = 0$: $Z_{Q'} = X_Q Z_Q$; $b' = 1$: $Z_{Q'} = (X_Q - Z_Q)Z_Q$
13	$Z_{Q'} \leftarrow C \cdot Z_Q$	// $b' = 0$: $Z_{Q'} = C X_Q Z_Q$; $b' = 1$: $Z_{Q'} = C(X_Q - Z_Q)Z_Q$
14	**Return** $(X_{Q'} : Z_{Q'})$	

Algorithm 7: Computing and evaluating a 2^a-isogeny

Function: `2aIso`

Parameters: *Integer* $a \in \mathbb{N}$, *Strategy* $(s_1, \ldots, s_{a-2}) \in (\mathbb{N}^+)^{a-2}$

Input: *Legendre curve coefficients* $(\kappa : \mu) \in \mathbb{P}^1$*; a point* $(X_R : Z_R) \in L_{[\kappa:\mu]}$ *of exact order* 2^a *with* $2^{a-1}R \neq (\kappa : 0 : \mu)$.

Output: *The* j*-invariant of* $L_{[\kappa':\mu']}$ *such that* $\varphi : L_{[\kappa:\mu]} \to L_{[\kappa':\mu']}$ *is an isogeny with* $\ker(\varphi) = \langle R \rangle$.

```
1  Initialize empty deque S
2  push(S, (a − 1, (X_R : Z_R)))
3  i ← 1
4  while S is not empty do
5      (h, (X : Z)) ← pop(S)
6      if h = 1 then
7          ([κ : μ], (c_0, c_1, c_2), b) ← 2IsoLegCurve((κ : μ), (X : Z))      // Alg. 1
8          Initialize empty deque S'
9          while S is not empty do
10             (h, (X : Z)) ← pull(S)
11             (X : Z) ← 2IsoLegEval((X : Z), (c_0, c_1, c_2), μ, b)          // Alg. 2
12             push(S', (h − 1, (X : Z)))
13         S ← S'
14     else if 0 < s_i < h then
15         push(S, (h, (X : Z)))
16         for j = 1 to s_i do
17             (X : Z) ← coordinateDBL((κ : μ), (X : Z))                      // Alg. 3
18         push(S, (h − s_i, (X : Z)))
19         i ← i + 1
20     else
21         Error: invalid strategy
22 jInv ← jInv((κ : μ), (X : Z), (c_0, c_1, c_2))                             // Alg. 4
23 Return jInv;
```