Fast Algebraic Algorithms for Arithmetic Geometry and Polynomial Systems

Pierre-Jean Spaenlehauer

UNIVERSITÉ DE LORRAINE

École doctorale IAEM Lorraine

Fast algebraic algorithms for arithmetic geometry and polynomial systems

THÈSE

soutenue le 12 février 2025

pour l'obtention d'une

Habilitation de l'Université de Lorraine

(mention informatique)

par

Pierre-Jean Spaenlehauer

Composition du jury

Président :	Alin Bostan	Directeur de Recherche, Inria
Rapporteurs :	Laurent Busé	Directeur de Recherche, Inria
	Wouter Castryck	Research Expert, KU Leuven, Belgique
	Jean-Marc Couveignes	Professeur, Université de Bordeaux
Examinateurs :	Kirsten Eisenträger	Professor, Pennsylvania State University, USA
	Pierrick Gaudry	Directeur de Recherche, CNRS, LORIA
	Emmanuel Jeandel	Professeur, Université de Lorraine

Laboratoire Lorrain de Recherche en Informatique et ses Applications — UMR 7503

Mis en page avec la classe thesul.

Résumé

Ce document synthétise mes travaux de recherche depuis ma soutenance de thèse en 2012. Le dénominateur commun de mes recherches est l'étude d'objets géométriques par le biais du calcul symbolique et algébrique, en ayant pour boussole des applications pratiques en cryptographie.

Un premier volet de mes travaux incorpore une part importante d'arithmétique et de théorie algorithmique des nombres, via l'étude des courbes (hyper)elliptiques, des variétés abéliennes et des modules de Drinfeld. Plus précisément, on s'intéresse aux problèmes de comptage de points sur les courbes hyperelliptiques définies sur des corps finis, à l'algorithmique des isogénies de modules de Drinfeld, et au calcul d'espaces de Riemann-Roch sur des courbes algébriques nodales.

Le deuxième volet de mes recherches se concentre sur les systèmes polynomiaux et leur algorithmique. Nous étudions le calcul de points critiques de fonctions polynomiales par des méthodes algébriques et numériques, avec un focus sur le problème de l'approximation de faible rang structurée. Nous nous intéressons également à des méthodes combinatoires pour construire des familles de systèmes qui ont peu de monômes mais beaucoup de solutions réelles positives. Enfin, nous présentons quelques résultats récents sur les systèmes polynomiaux avec des structures monomiales du point de vue de la géométrie torique.

Abstract

This document summarizes my research work since my Ph.D. thesis defense in 2012. The common denominator of my research is the study of geometric objects by means of symbolic and algebraic computations, using practical applications in cryptography as a compass.

The first part of my work incorporates arithmetic and algorithmic number theory, via the study of (hyper)elliptic curves, abelian varieties and Drinfeld modules. More specifically, we focus on point counting problems on hyperelliptic curves defined over finite fields, algorithms for isogenies of Drinfeld modules, and computations of Riemann-Roch spaces over nodal algebraic curves.

The second part of my research investigates polynomial systems and their algorithms. We study the computation of critical points of polynomial functions by algebraic and numerical methods, with a focus on the problem of structured low-rank approximation. We are also interested in combinatorial methods for constructing families of systems that involve few monomials but many positive real solutions. Finally, we present some recent results on polynomial systems with monomial structures from the viewpoint of toric geometry.

Remerciements

Qu'est-ce qu'une Habilitation à Diriger les Recherches ? Je crois que la réponse m'a été apportée par les collègues qui m'ont exprimé soutien et encouragements au cours de la rédaction et de la préparation de la soutenance : il s'agit en premier lieu d'une occasion de partager un moment festif avec mes collègues, ma famille, et mes amis. C'est aussi l'opportunité de leur présenter les questions sur lesquelles je travaille (ou devrais-je plutôt dire: les questions qui me travaillent).

First, I would like to express my gratitude to the members of the jury. The works of Laurent Busé, Wouter Castryck, and Jean-Marc Couveignes have been inspirational for my research, and I am truly honored and thankful that they accepted to review my Habilitation thesis. I met Alin Bostan when I was a Master's student in Paris twenty years ago, and I have lost count of the number of times his articles contained precise answers to questions I had; thank you for accepting to be the president of the jury. I discovered Kirsten Eisenträger's work while preparing a talk on Bruhat-Tits trees for our local "isogeny reading club," and I am grateful that she accepted to be part of the jury. I am also grateful to Emmanuel Jeandel for joining the jury. Thank you, Pierrick Gaudry, for your guidance as my "parrain d'HDR" and for your support and advice throughout the preparation of the manuscript.

Cette HDR n'existerait pas sans toutes les personnes avec qui j'ai eu le privilège de collaborer, et avec qui j'ai partagé les moments de doutes, les papiers acceptés (et rejetés), les nuits blanches avant les deadlines, et les trouvailles inattendues. Un grand merci aux étudiant-e-s que j'ai eu le plaisir de cotoyer et qui ont joué un grand rôle dans l'évolution de mes recherches — Simon Abelard, Léo Barré, Joël Felderhoff, Aude Le Gluher, Antoine Leudière, Nicolas Lévy, Julien Soumier — ainsi qu'à mes co-auteurs Magali Bardet, Matías Bender, Frédéric Bihan, Jean-Charles Faugère, Pierrick Gaudry, Giorgio Ottaviani, Mohab Safey El Din, Bruno Salvy, Francisco Santos, Éric Schost, Bernd Sturmfels, Jules Svartz, Emmanuel Thomé.

Merci à toutes les personnes qui contribuent à rendre l'environnement de travail dans lequel j'évolue agréable et convivial. Je pense bien entendu à toute l'équipe CARAMBA (jeunes et moins jeunes) et à Emmanuelle Deschamps ; mais aussi à Gwendal Kervern et Pierre-Jean Panteix (chimistes experts de la transformation d'électricité en décibels), au club échecs et aux coureurs de la pause midi.

Merci à mes parents et à ma soeur d'avoir toujours soutenu mes choix de vie.

Pour finir, mes pensées vont aux trois personnes qui ont eu une influence considérable sur chaque page de ce manuscrit. Les travaux décrits dans cette HDR ont été effectués sur une période de douze ans. Merci Sophie pour ton soutien constant et sans faille pendant ces années, sans lequel ce manuscrit n'aurait pas vu le jour. Merci Daphné et Mathias pour la joie et l'énergie que vous m'apportez à chaque instant.

Contents

Introduction **vii**

1 Computational arithmetic geometry **1**

1.1 Panorama 1

1.2 Point-counting and zeta functions of curves over finite fields of large characteristic 3

1.2.1 Motivation 3

1.2.2 An asymptotical complexity bound for counting points on hyperelliptic curves of large genus 7

1.2.3 Genus-3 hyperelliptic curves with explicit real multiplication 11

1.3 Isogenies of Drinfeld modules and effective class field theory of hyperelliptic function fields 14

1.3.1 Motivation 14

1.3.2 Computing a group action from the class field theory of hyperelliptic function fields 16

1.3.3 Algorithms 17

1.3.4 An explicit computation 19

1.4 Computing Riemann-Roch spaces for nodal curves 20

1.4.1 Motivation 20

1.4.2 Revisiting the Brill-Noether's method 22

1.4.3 Algorithms and complexity 24

1.4.4 Implementation and experimental results 26

1.4.5 Follow-up works 28

2 Polynomial systems **29**

2.1 Panorama 29

2.2 Sparse polynomial systems 30

2.2.1 Preliminaries 30

2.2.2 Algorithms for sparse systems 32

2.2.3 Homogeneous coordinate rings for projective toric varieties 32

2.2.4 Dimensions and regular sequences in complete toric varieties from polyhedral fans . . . 36
2.2.5 Quadratic fewnomials . . . 39
2.3 Real polynomial systems with many positive solutions . . . 42
2.3.1 The number of positive solutions of fewnomials . . . 42
2.3.2 A polyhedral construction . . . 44
2.4 Computations of critical points . . . 47
2.4.1 Varieties defined by generic polynomials . . . 49
2.4.2 Smooth varieties . . . 52
2.5 Structured low-rank approximation . . . 52
2.5.1 Numerical approach . . . 53
2.5.2 Algebraic and symbolic approach . . . 57

3 Research project **61**
3.1 Point counting for general curves of large characteristic . . . 61
3.2 Isogenies of products of isogenous elliptic curves with complex multiplication . . . 62
3.3 Drinfeld modules . . . 63
3.4 Plane curves with non-degenerate singularities . . . 63
3.5 Gröbner basis engineering and technology . . . 66

Bibliography **69**

Introduction

My research is centered on *algorithms* for *multivariate polynomials*, from the point of view of *computer algebra* at the interface between *number theory* and *algebraic and arithmetic geometry*. In this thesis, I describe the main topics that I have been studying since my Ph.D. in 2012 and the evolution of my scientific focus over the years.

The general context of my work is to design efficient computational tools for manipulating mathematical objects, which is one of the central objectives of *computer algebra*. This point of view is strongly connected to applications, since the endgoal of such tools is to be used to solve concrete problems. My research is mainly centered on tools for algebraic objects, although some of my works are also connected to numerical computations. One large domain of applications where many algebraic computational tools are required is *public-key cryptology*. Indeed, the foundations of the security of cryptographic protocols rely on the difficulty of an underlying algebraic problem. Studying the complexity of such problems often requires a wide range of algorithmic and complexity tools. The number of problems that can serve as a foundation for cryptographic constructions has grown during the last years, in particular due to the fast developement of *post-quantum cryptography*. One of the main objective of post-quantum cryptography is to identify hard algebraic problems which cannot be efficiently solved with quantum algorithms, and which can be used to design cryptographic protocols that would be resistant even against an adversary who would have access to a large quantum computer.

Arithmetic geometry lies at the interface between algebraic geometry and number theory. Roughly speaking, it studies how arithmetic structures such as integers, polynomials, number fields, function fields, are connected to geometry. In this thesis, arithmetic structures arise frequently as endomorphisms of geometric objects. Among the most prominent such geometric objects are *elliptic curves*, whose theoretical and computational relevance has a long and rich history. On the computational side, perhaps the foundational results that have shed light on arithmetic geometry are the discovery in the eighties of *elliptic curve cryptography* and of the ECM algorithm to factor integer numbers. Since then, many computational uses of elliptic curves and abelian varieties have been found to study integer numbers. On the function field side, the analog of elliptic curves are *Drinfeld modules*, which were introduced by Drinfeld in [41]. During the last decades, there have been many developments of the algorithmic and computational toolbox for Drinfeld modules.

Polynomial systems are versatile tools to encode *inverse problems* in commutative algebra, i.e. finding values of parameters that lead causally to some observations. Indeed, many algebraic constructions end up being a sequence of ring operations (additions and multiplication) on discrete structures. It is therefore quite standard that polynomial system solving can model the inversion of algebraic maps. This motivates the need of having general reliable and efficient tools that can solve polynomial systems, in particular when there are finitely-many solutions over an algebraic closure. Gröbner bases provide such general computational tools, and this is probably

the reason why they attracted a lot of attention since their discovery by Bruno Buchberger in his Ph.D. thesis [25]. However, one of the main issue of Gröbner bases is the difficulty of estimating the cost of the computations. Works by Daniel Lazard in the 80s showed the relationship between Gröbner bases and linear algebra [82], building on works by Macaulay at the beginning of the 20th century. Studying polynomial systems from a computational viewpoint also requires a fine analysis of the computational tools for linear algebra and arithmetic. This combination of mathematics, complexity, and fine computational tools for basic operations is a major feature of modern computer algebra.

One of the aims of computer algebra is to provide computational tools. Probably the best way to make these tools available to other researchers is to provide them with reliable, versatile, and efficient software. Since software development often requires a lot of energy, engineering skills, and time, this is a difficult task. Yet, I believe that this is an important part of research in computer algebra and one of the endgoal of my work is to contribute to software in order to make algorithms practical and easily usable.

Contributions

During my Ph.D. thesis, my main focus was on zeros and on critical points of polynomial maps. These themes are strongly connected to geometry, since their study is often linked to the topology of associated geometrical objects. In applications, we often encounter 0-dimensional systems, i.e. systems of polynomial equations defined over a base field k, whose set of solutions over an algebraic closure $\overline{k}$ is finite. To compute these solutions, one of the most prominent algorithmic tool is *Gröbner bases*. At first sight, it might seem that Gröbner bases are a purely algebraic tool; however, it has several strong connections to geometry. Even just from the point of view of complexity analysis, the maximal degree that modern Gröbner bases algorithms should reach is related to topology via local cohomology.

My early works during my Ph.D. thesis and during my postdoc years were centered around three related families of algebraico-geometric objects related to polynomial system solving:

1. Determinantal systems and determinantal varieties. On the algebraic side, this corresponds to systems of polynomials obtained by computing minors of a matrix whose entries are polynomials. On the geometric side, determinantal varieties are low-rank elements in families of polynomially parametrized matrices.

2. Multi-homogeneous systems and multi-projective varieties. On the algebraic side, multi-homogeneous systems are homogeneous (often of low degree) with respect to a partition of the variables. On the geometric side, multi-projective varieties are subvarieties of products of projective spaces.

3. Critical points of polynomial maps. Critical points can be encoded via a rank condition on the Jacobian matrix of the map, or via Lagrange multipliers, giving rise to determinantal or to multi-homogeneous systems.

During my postdoc years, I worked on a specific problem that it is related to these algebraic structures: *Structured Low-Rank Approximation*. This problem is about finding the closest low-rank matrix to a given real matrix, under linear constraints. This problem is a blueprint for many famous problems in symbolic-numeric computations (approximate GCD, approximate multivariate factorization, approximate tensor decomposition, etc.) and it can be modeled via critical points and determinantal or multi-homogeneous systems.

Contribution: In [107], with Éric Schost, we designed quadratically convergent numerical algorithms for Structured Low-Rank Approximation (Section 2.5.1). In [95], with Giorgio Ottaviani and Bernd Sturmfels, we proposed algebraic tools to analyze the algebraic difficulty of Structured Low-Rank Approximation (Section 2.5.2).

During the same period, I continued my work on the computations of critical points of polynomial maps, and I started working on systems with monomial structures, which generalize multi-homogeneous systems. The algebraic approach used during the work with Bernd Sturmfels and Giorgio Ottaviani brought useful tools to estimate the algebraic degree of critical points of polynomial maps. With Mohab Safey El Din, we used similar tools to bound the complexity of computations by using variants of the geometric resolution algorithm.

Contribution: In [111], I proposed new complexity bounds for the computation of critical points of polynomial maps restricted to an algebraic variety V with Gröbner bases under genericity assumptions on the coefficients of the input polynomials (Section 2.4.1). In [104], with Mohab Safey El Din, we proposed algorithms and complexity bounds for computing critical points in the case where V is smooth, in terms of numerical data associated to V (Section 2.4.2).

In 2014, I arrived in the CARAMBA team, and this is where my research interests started incorporating elements from computational number theory, function fields and arithmetic geometry, with a view towards applications in cryptology. Shortly after I arrived we started working on *point-counting algorithms* for hyperelliptic curves with Pierrick Gaudry and Simon Abelard. This topic has the nice feature that it is a mixture of my previous research themes (0-dimensional system solving, effective algebraic geometry) and the new elements that I wanted to incorporate in my research (computational number theory, arithmetic geometry). It also has strong connections with cryptography: low-genus curve-based cryptography requires fast point-counting algorithms to find curves suitable for cryptographic applications, and the point-counting toolbox is also very useful for isogeny-based cryptography.

Contribution: In [6], with Pierrick Gaudry and Simon Abelard, we proposed a new Schoof-like point-counting algorithm when the base field has a large characteristic and the genus is large. We proved a new upper bound on the complexity of this algorithm: this complexity is polynomial in $\log q$ for fixed g, and the exponent has a linear dependency on g, whereas the dependency was quadratic in the previous known bounds (Section 1.2.2).
In [5], with Pierrick Gaudry and Simon Abelard, in the case of hyperelliptic curves of genus 3 with explicit real multiplication, we proposed a new algorithm with improved asymptotic complexity compared to the state-of-art (Section 1.2.3).

At the same time, I started a long collaboration with Frédéric Bihan on the topic of sparse polynomial systems with many nondegenerate positive solutions. This work takes place at the interface between real geometry and sparse polynomial systems, and it aims at studying how monomial supports impact the number of real solutions of polynomial systems. The geometrical

framework of this investigation is Viro's combinatorial patchworking method, which is strongly connected to *tropical geometry*. This framework provides a way to describe asymptotic topological properties of real varieties by studying geometric combinatorial objects such as polytopes and polyhedral complexes. We worked with Francisco Santos, who is a reknowned expert on this topic.

> **Contribution:** In [22], with Frédéric Bihan and Francisco Santos, we proved new lower bounds on the number of nondegenerate isolated positive real solutions a system with prescribed number of variables and number of monomials can have (Chapter 2.3).

During the same period, I also started the development of a C++/NTL implementation of the F4 algorithm for computing Gröbner bases, see `https://gitlab.inria.fr/pspaenle/tinygb`. The aim of this implementation was to have an experimental platform to include improvements for systems with monomial structures.

In 2018, I started working with Aude Le Gluher on Riemann-Roch spaces during her Master's internship. The aim of this work was to study old methods by Brill and Noether related to birational geometry for the computation of Riemann-Roch spaces, by looking at them from the point of view of modern computer algebra.

> **Contribution:** In [85], with Aude Le Gluher, we proposed an algorithm to compute bases of Riemann-Roch spaces on nodal plane projective curves. We proved upper bounds on the complexity of this algorithm, and we provided a complete C++/NTL implementation which is faster than the state-of-the-art software on many examples (Section 1.4). This work was followed by several developments by other authors, who improved the asymptotic complexity and proposed variants that can compute with more general classes of curves.

During Aude Le Gluher's Ph.D. thesis (co-advised with Emmanuel Thomé), we worked on analytic number theory and on the complexity of the Number Field Sieve, which is the fastest known algorithm to factor integers and to compute discrete logarithms in the multiplicative group $\mathbb{F}_q^\times$. The following contribution is not described in detail in this manuscript, since the mathematical framework of this work is different from the rest of my work as it does not involve many geometrical aspects.

> **Contribution:** In [86], with Aude Le Gluher and Emmanuel Thomé, we provide a complexity analysis of the Number Field Sieve for factoring an integer N which provides evidence that the asymptotic complexity usually considered comes from the first term of a function series which converges only for N larger than $\exp(\exp(25)) \approx 2^{103881111194}$. This suggests than using the classical asymptotic complexity bound for estimating the difficulty of factoring a number in cryptographic applications is irrelevant, as cryptographic sizes are too small.

In 2021, I wanted to dig further in the computational theory of function fields, and this is why I started to work with Antoine Leudière on algorithms for Drinfeld modules during his Master's internship. After his Master's internship, Antoine Leudière continued working on Drinfeld modules during his Ph.D. thesis, co-advised by Emmanuel Thomé and myself. Drinfeld modules

are geometrical objects which have a strong relationship with function fields: endomorphisms of Drinfeld modules can be seen as functions on curves. This is similar to the strong relationship between elliptic curves and imaginary quadratic number fields, which is the root of a large part of modern arithmetic geometry and number theory.

> **Contribution:** In [87], with Antoine Leudière we proposed algorithms to compute efficiently and to invert a group action in the class field theory of hyperelliptic function fields (Section 1.3).

During the period 2020-2023, I also continued working on sparse polynomial systems with Matías Bender. In particular, we focused on the problem of deciding whether some toric homogenizations were compatible with monomial structures, in the sense that they do not create high-dimensional artefacts generically. This project is only starting: with this work, we identified geometrical conditions which set frameworks which we would like to use to design algorithms for solving 0-dimensional systems with special monomial structures.

> **Contribution:** In [15], with Matías Bender, we proved a combinatorial formula for the dimension of subvarieties of toric varieties built from complete polyhedral fans defined by generic systems with prescribed monomial support (Section 2.2.4).

Applications

Most of the contributions presented in this thesis are motivated by cryptographic applications. I have decided not to put too much emphasis on cryptography in this report since this is not the core of my research. However, I would like to point out that the algorithmic toolbox for arithmetic geometry plays an important role in isogeny-based cryptography, algebraically-geometric codes, and classical curve-based cryptography. The toolbox for polynomial systems is also quite important in cryptography. For instance, many recent works on the cryptanalysis of code-based cryptosystems rely on such advanced algorithmic tools for polynomial systems, see e.g. [13] and references therein. During the last decade, there has been a lot of progress on isogeny-based cryptography, which is a candidate for being secure against quantum computers. The theory is still growing, and new advanced tools are frequently found. These tools often rely on objects and results from classical arithmetic geometry. A recent example of the fast development of the isogeny toolbox is the design of algorithms around Kani's results on isogeny diamonds [74], which led to new cryptographic constructions, see e.g. [33].

Organization of the thesis

The presentations of the contributions is organized thematically. The first chapter focusses mainly on computational number theory and arithmetic geometry. The second chapter focusses on polynomial systems and computational algebraic geometry. These two chapters interact in several ways; For instance multi-homogeneous polynomials play an important role in the point-counting algorithms for hyperelliptic curves. The third chapter describes my research project and follow-ups of the contributions presented in this thesis.

Chapter 1

Computational arithmetic geometry

1.1 Panorama

A large part of my work during the last ten years is about *effective arithmetic geometry* and its applications in cryptography. Arithmetic geometry deals with algebraic varieties endowed with an *arithmetic structure*. By this, we often mean a module structure over a (subring of) a Dedekind ring. Classical examples of such objects include *abelian varieties*, i.e. smooth projective varieties whose points have a $\mathbb{Z}$-module structure. The theory of *complex multiplication* endows some abelian varieties with a richer $\mathcal{O}$-module structure, where $\mathcal{O}$ is an order in a number field. These abelian varieties appear with many flavors in cryptography, the most prominent incarnation being elliptic curves (a.k.a. abelian varieties of dimension 1) over finite fields and Jacobian varieties of hyperelliptic curves of genus 2 over finite fields. More recently, other families of abelian varieties have made a sensational entrance in the cryptographic world: products of maximal elliptic curves over finite fields, which play an important role in the recent groundbreaking works on isogeny-based cryptography.

Recently, I have also studied effective aspects of Drinfeld modules, which are 1-dimensional objects equipped with a structure of $\mathbb{F}_q[X]$-module (or more generally of A-module structure, where A is a Dedekind ring in a finite extension of $\mathbb{F}_q(X)$). These objects have many similarities with elliptic curves.

The most prominent applications of abelian varieties arise in public-key cryptography. In fact, these objects have been around the scene since the discovery of public-key cryptography in the 80s, see [80, 92], because the multiplication by an integer in the $\mathbb{Z}$-module structure can be computed straightforwardly, but the inverse problem — called *discrete logarithm* — is computationally difficult. This asymmetry can be exploited for constructing many cryptographic protocols for encryption, signature, key-exchange, etc. The discrete logarithm on abelian varieties of dimensions 1 and 2 are still nowadays among the most reliable mathematical problems on which the security of practical cryptographic protocols relies. For instance, it is used in passports or for authenticating websites.

One of the challenges to use such objects in cryptography is to identify large groups of points of prime orders in abelian varieties defined over finite fields. This can be achieved by computing the number of rational points of the abelian varieties over extensions of the base fields and then by factoring it. The problem of computing the number of rational points over extensions is known as the *point counting problem*. This problem has far-reaching connections with famous conjectures, for instance the Birch-Swinnerton-Dyer conjecture. The point-counting problem can expressed in a very elementary way. We start with a trivariate homogeneous polynomial

$Q(X,Y,Z) \in \mathbb{F}_q[X,Y,Z]$ which defines a nonsingular geometrically irreducible projective plane curve. This means that we assume that the polynomial $Q \otimes \overline{\mathbb{F}_q}$ is irreducible over the algebraic closure, and the homogeneous system

$$\frac{\partial Q}{\partial X}(X,Y,Z) = \frac{\partial Q}{\partial Y}(X,Y,Z) = \frac{\partial Q}{\partial Z}(X,Y,Z) = 0.$$

has no common solution in $\overline{\mathbb{F}_q}^3 \setminus \{(0,0,0)\}$. The number of nonzero solutions of the equation $Q(X,Y,Z) = 0$ over a finite extension $\mathbb{F}_{q^n}$ of $\mathbb{F}_q$ is divisible by $q^n - 1$ since there is a faithfully transitive action of $\mathbb{F}_{q^n}^\times$ on the nonzero solutions because of the homogeneity of Q. Then for $n > 0$, we let N_n denote

$$N_n \stackrel{\text{def}}{=} \frac{|\{(x,y,z) \in \mathbb{F}_{q^n}^3 : (x,y,z) \neq (0,0,0) \text{ and } Q(x,y,z) = 0\}|}{q^n - 1}.$$

Then Weil conjectures (see e.g. [89, Thm. 6.1]) imply that the series

$$(1-T) \cdot (1-qT) \cdot \exp\left(\sum_{n=1}^{\infty} N_n \frac{T^n}{n}\right) \in \mathbb{Q}[[T]]$$

is in fact a polynomial with integer coefficients. The point counting problem can then be formulated as

Problem 1.1.1 (Point counting problem for nonsingular curves)**.** *Given a trivariate polynomial $Q(X,Y,Z) \in \mathbb{F}_q[X,Y,Z]$ defining a nonsingular geometrically irreducible projective plane curve, compute the polynomial*

$$(1-T) \cdot (1-qT) \cdot \exp\left(\sum_{n=1}^{\infty} N_n \frac{T^n}{n}\right) \in \mathbb{Z}[T].$$

The point counting problem plays an important role in applications in classical curve-based cryptography: algebraic curves over finite fields and their rational points provide finite groups which can be used in cryptography via the discrete logarithm problem, provided that we can show that the order of this group contains a large prime factor. The order of this group can be computed via the point counting problem.

Unfortunately, it is known that the discrete logarithm could be solved in polynomial time with a quantum computer. Such a machine does not yet exist, still cryptographers have to prepare for this eventually, since we need to trust that data which is encrypted today will not be broken in the upcoming decades. This observation led to the advent of *post-quantum cryptography*, i.e. the quest for efficient cryptosystems which would still be secure even against an adversary which would have access to a large quantum computer. Surprisingly, another feature of abelian varieties is useful in this context: some graphs of abelian varieties with their morphisms have nice mixing properties, and this can be used to build mathematical problems for which no polynomial quantum algorithm is known. These mathematical problems can then be used to construct cryptographic protocols whose security relies on their computational difficulty. Initially, this idea was proposed by Couveignes in [31]. This was rediscovered by Rostovtsev and Stolbunov [103]. Although the initial Couveignes-Rostovtsev-Stolbunov (CRS) cryptosystem suffered from efficiency issues, this paved the way to the SIDH cryptosystem, proposed in [34].

In 2022, a new family of attacks against SIDH have been proposed [29, 90, 102]. These attacks involve abelian varieties of dimensions $\geqslant 2$ and they have radically changed the landscape of isogeny-based cryptography. The have also paved the way for many new cryptographic constructions, which sometimes build on abelian varieties of larger dimensions, see e.g. [33].

Surprisingly, the mathematical and algorithmic toolboxes for the point-counting problem and for isogeny computations have many similarities, as behind the scene the *endomorphism rings* of the Jacobians of the curves plays an important role.

Since Drinfeld modules have many similarities with elliptic curves, it is quite natural to investigate whether these objects could be used as replacements of elliptic curves in cryptographic protocols. Several tries have been made in this direction. In classical cryptography, it has been known for several decades that using Drinfeld modules instead of elliptic curves for the discrete logarithm problem would be insecure [105, 24]. The question of using Drinfeld modules for isogeny-based cryptography is more complicated. In [73], the authors propose analogs to the cryptosystems SIDH and CSIDH based on supersingular Drinfeld modules instead of supersingular elliptic curves, and they show that these systems can be broken. With Antoine Leudière, in [88], we proposed an alternative to the CRS cryptosystem by using ordinary Drinfeld modules. This proposal was broken by Benjamin Wesolowski in [121], who found a way to compute efficiently isogenies between Drinfeld modules. Nevertheless, our initial proposal relied on class field theory of hyperelliptic function field, and the work of Wesolowski allowed us to turn our cryptographic proposal into an efficient algorithm to compute a group action from the point of view of computational number theory.

1.2 Point-counting and zeta functions of curves over finite fields of large characteristic

This section is about results that have been obtained with Simon Abelard and Pierrick Gaudry, and they are part of the Ph.D. thesis of Simon Abelard.

1.2.1 Motivation

Zeta functions of algebraic varieties defined over finite fields have been in the mathematical landscape for a very long time. They are objects which encode the number of rational solutions of polynomial equations over finite fields. Long before the modern mathematical formalism for L-functions was designed, such questions were already studied by Gauss, see e.g. [58, art. 358] or the introduction of [120].

Weil conjectures brought a lot of attention to this area, which became one of the most prominent topic in mathematics around the middle of the 20th century. These conjectures are very similar to the Riemann hypothesis for integers (and its generalization to number fields): it states that the distribution of the number of solutions of equations over extensions of finite fields has a strong structure. This structure is expressed via the rationality of a generating series, via a functional equation, and via a norm condition on the roots of the generating series.

In this section, we will focus on zeta functions of *algebraic curves*. In fact, the one-dimensional objects have an extra layer of structure as the associated rings of functions feature useful properties such as unique factorization of ideals (via Dedekind rings), similarly to integers and number fields which are also objects of dimension 1.

Let us start with the definitions of the geometrical objects that will be used throughout this section. In the formalism of schemes, an *affine algebraic curve* over a perfect field k is

a topological space whose points are the prime ideals in a finitely-generated commutative k-algebra R which has the property that for any chain $\mathfrak{p}_1 \subsetneq \mathfrak{p}_0 \subsetneq R$ of prime ideals, $\mathfrak{p}_0$ is maximal. The topology on this set — called the *Zariski topology* — is generated by the closed sets $V_f = \{\mathfrak{p} \text{ prime ideal in } R : f \in \mathfrak{p}\}$ for all $f \in R$. This topological space carries an extra structure of a *ringed space* as there is a natural sheaf of commutative rings such that the stalk at a prime ideal P is isomorphic to the localization R_P of the ring R at P.

Since any finitely-generated commutative k-algebra is isomorphic to a quotient of a polynomial ring $k[X_1, \ldots, X_n]/I$, an affine algebraic curve can be encoded by an ideal $I \subset k[X_1, \ldots, X_n]$ satisfying the following property: if $\mathfrak{p}_0, \mathfrak{p}_1$ are prime ideals such that $I \subset \mathfrak{p}_1 \subsetneq \mathfrak{p}_0 \subsetneq k[X_1, \ldots, X_n]$, then $\mathfrak{p}_0$ is maximal. The curve is *geometrically irreducible* if $I \otimes_k \overline{k} \subset k[X_1, \ldots, X_n] \otimes_k \overline{k}$ is a prime ideal, where $\overline{k}$ is an algebraic closure of k. It is *reduced* if I is radical.[1]

A projective algebraic curve over a field k is also a ringed topological space which can be encoded by a radical homogeneous ideal $I \subset k[X_0, \ldots, X_n]$ satisfying the following properties:

- if $\mathfrak{p}_0, \mathfrak{p}_1, \mathfrak{p}_2$ are prime ideals such that $I \subset \mathfrak{p}_2 \subsetneq \mathfrak{p}_1 \subsetneq \mathfrak{p}_0 \subsetneq k[X_0, \ldots, X_n]$, then $\mathfrak{p}_0$ is maximal;
- the ideal I is saturated: if $f \in k[X_0, \ldots, X_n]$ is such that $f \cdot \langle X_0, \ldots, X_n \rangle^\ell \subset I$ for some $\ell \geqslant 0$, then $f \in I$.

As in the affine case, a projective curve is geometrically irreducible if $I \otimes_k \overline{k} \subset k[X_0, \ldots, X_n] \otimes_k \overline{k}$ is a prime ideal, where $\overline{k}$ is an algebraic closure of k. It is reduced if I is radical. As in the affine case, there is a systematic construction from *graded* finitely-generated commutative k-algebras via the $\mathrm{Proj}(\cdot)$ construction.

Throughout this section, unless stated otherwise, all curves are assumed to be reduced and geometrically irreducible.

The simplest model of curves (and often most practical for computational purposes) are plane curve, i.e. when $n = 2$. The nice property that is convenient for computations is that affine (resp. projective) plane curves can be represented by *principal* ideals (resp. principal homogeneous ideal) in $k[X, Y]$ (resp. $k[X, Y, Z]$). From a computational perspective, this allows us to manipulate bivariate polynomials, instead of ideals in polynomial rings. Unfortunately, not every algebraic curve is isomorphic to a plane curve. However, if we allow finitely-many defects, then all curves can be represented by plane curves: two algebraic curves C_1, C_2 over k are birationally equivalent if there are dense open subsets $O_1 \subset C_1$, $O_2 \subset C_2$ for the Zariski topology such that O_1 and O_2 are isomorphic. Up to birational equivalence, every curve can be represented by a planar model. In fact, the situation is even better: over the field of complex numbers, every curve is birationally equivalent to a projective plane curve whose singularities are nodes, which are the simplest type of singularities [8, Appendix A], see Theorem 1.4.3.

A point $\mathfrak{p} \subset R \cong k[X, Y]/Q$ on an affine curve encoded by a polynomial $Q(X, Y) \in k[X, Y]$ is *singular* if $\partial Q/\partial X$ and $\partial Q/\partial Y$ belong to $\mathfrak{p}$. A curve is nonsingular if it does not have any singular point. A point $\mathfrak{p}$ is *closed* if the singleton $\{\mathfrak{p}\}$ is closed in the Zariski topology. Notice that $\mathfrak{p}$ is closed on an affine curve if and only if it is a maximal ideal. Then we define the *degree* of a closed point as the degree of the field extension $[(R/\mathfrak{p}) : k]$. For projective curves, the degree

[1] Most results also hold for nonperfect base fields, although they require some subtle ajustments; in particular, geometric irreducibility should often be replaced by geometric connectness, which is a slightly stronger notion for nonperfect fields. For simplicity, since all instanciations of the base fields will be perfect in this document, we will assume that k is perfect.

of a closed point is defined similarly by restricting to an affine chart which contains the point. A point is called *rational* if it has degree 1.

Definition 1.2.1. *[89, Ch. 8, Def. 5.4] Let C be a nonsingular projective geometrically irreducible reduced curve C defined over $\mathbb{F}_q$. For $d \in \mathbb{Z}_{>0}$, let b_d denote the number of closed points of degree d and set $N_n \stackrel{def}{=} \sum_{d|n} d\, b_d$. The* zeta-function *of C is the power series*

$$\mathbf{Z}(C;T) = \exp\left(\sum_{n=1}^{\infty} N_n \frac{T^n}{n}\right) \in \mathbb{Q}[[T]].$$

The zeta-function encodes in a generating series the number of closed points of each degree of a nonsingular curve. The Weil conjectures describe how this zeta-function contains some structural information about the distributions of points on algebraic curves. Before stating the Weil conjectures, we need to define a numerical value associated to an algebraic curve: its *geometric genus*. The genus gives topological information about the curve. When $k = \mathbb{C}$, the curve can be seen as a surface over $\mathbb{R}$ and the genus describes topological properties of this surface. Over more general fields, the geometric genus needs some prerequisite to be cleanly defined. Perhaps we can just say that this number has a cohomological flavour, and it is related to Poincaré duality via Riemann-Roch theorem. A nice feature of the genus is that for nonsingular curves embedded in a projective space, it can be quite conveniently computed from the degree of the curve and from the critical points of a projection to $\mathbb{P}^1$ via Riemann-Hurwitz formula [89, Ch. 9, Sec. 8].

Theorem 1.2.2 (Weil conjectures for nonsingular projective curves)**.** *Let C be a curve of genus g as in Theorem 1.2.1. Then:*

- *(Rationality [89, Ch. 8, Thm. 6.1]) There exists a polynomial $f \in \mathbb{Z}[T]$ of degree $2g$ such that*
$$\mathbf{Z}(C;T) = \frac{f(T)}{(1-T)(1-qT)}.$$
- *(Functional equation [89, Ch. 8, Thm. 7.1])*
$$\mathbf{Z}(C;(qT)^{-1}) = (qT^2)^{1-g}\mathbf{Z}(C;T).$$
- *(Riemann hypothesis [89, Ch.10, Sec. 5]) The roots of f in $\mathbb{C}$ have complex norm $\sqrt{q}$.*

The rationality of the zeta-function implies that there is a well-defined computational problem:

Problem 1.2.3 (Point Counting Problem)**.** *Given a geometrically irreducible complete curve C, compute the numerator of the zeta-function of a nonsingular curve birationally equivalent to C.*

The reason why this is called the *point counting problem* is because we can recover the number of rational closed points of C from the zeta-function. Moreover, one nice application of the functional equation and Riemann hypothesis is to obtain bounds on the coefficients of the numerator of the zeta-function. Indeed, by writing $f(T) = a_0 + a_1\, T + \cdots + a_{2g}T^{2g}$ and by expanding the functional equation, we obtain that $a_{2g-i} = q^{i-g}a_i$. Also, we have $a_{2g} = 1$ and Riemann hypothesis implies that $|a_i| \leqslant \binom{2g}{i} q^{i/2}$. Consequently, the size of the output of the point counting problem is well-bounded in terms of the input size.

One of the main computational tool to compute the numerator of the zeta-function of the curve is the study of the action of the Frobenius endomorphism on the Jacobian variety of the curve. The Jacobian variety $\mathrm{Jac}(C)$ of a nonsingular projective curve defined over k is an abelian variety of dimension g over k. Its points correspond to degree-0 divisors on C modulo linear equivalence. The group of its $\overline{k}$-points is isomorphic to the degree-0 part of the Picard group $\mathrm{Pic}(C/\overline{k})$ of the curve, seen as a curve defined over $\overline{k}$. The absolute Galois group $\mathrm{Gal}(\overline{k}/k)$ acts on $\mathrm{Jac}(C)$ in a way that is compatible with the group structure of the abelian variety. A nice feature of the Jacobian variety is that for any prime ℓ distinct from the characteric of k, the ℓ^i-torsions linked together with the multiplication-by-ℓ map form a projective system whose projective limit — called the *Tate module* — is a free module of rank $2g$ over the ring $\mathbb{Z}_\ell$ of ℓ-adic integers. Therefore, every automorphism $\mathrm{Gal}(\overline{k}/k)$ acts on the Tate module as an invertible morphism $\mathbb{Z}_\ell^{2g} \to \mathbb{Z}_\ell^{2g}$, and its characteristic polynomial (in $\mathbb{Z}_\ell[T]$) is independent of the choice of the basis.

In the context of a finite field $\mathbb{F}_q$, a generator for $\mathrm{Gal}(\overline{\mathbb{F}_q}/\mathbb{F}_q)$ is the Frobenius automorphism. Surprisingly, it can be proved that the characteristic polynomial of the Frobenius endomorphism on the ℓ-Tate module has in fact integer coefficients which do not depend on the choice of ℓ [89, Ch. 11, Thm. 5.2].

Moreover, this characteristic polynomial is in fact exactly the numerator of the zeta-function of the curve:

Theorem 1.2.4. *[89, Ch. 11, Thm. 5.2]. The reciprocal of the characteristic polynomial $\chi_F \in \mathbb{Z}[T]$ of the Frobenius endomorphism on* $\mathrm{Jac}(C)$ *equals the numerator of the zeta-function* $\mathbf{Z}(C;T)$.

Therefore, there is a clear computational plan to compute the zeta function:

- Compute an algebraic representation for $\mathrm{Jac}(C)$;
- Compute the ℓ-torsion of $\mathrm{Jac}(C)$;
- Compute the action of the Frobenius endomorphism on the ℓ-torsion; Its characteristic polynomial provide us with the numerator of the zeta-function modulo ℓ;
- Start again with other prime numbers ℓ and reconstruct the zeta-function by using the Chinese Remainder Theorem.

This approach has been designed first by Schoof for computing in time polynomial in $\log(q)$ the number of rational points on elliptic curves ($g = 1$) in his landmark article [106]. This approach was generalized in [98] where Pila showed how to extend Schoof's approach to compute the characteristic polynomial of the Frobenius endomorphism on any abelian variety.

These algorithms found strong applications in the realm of cryptography. Perhaps the most direct approach comes from the fact that many cryptographic protocols require cyclic finite groups G in which the *discrete logarithm problem* is hard: given a generator g of G and $h \in G$, finding an integer n such that $g^n = h$ should be computationally difficult. One of the most reliable source of such groups is obtained by considering a cyclic subgroup of prime order in a group of rational points of an abelian variety defined over a finite field. In order to find such a subgroup, the most classical approach is to generate abelian varieties at random, then count their number of rational points; if this number is divisible by a large prime number N, then we can build a cyclic group of this order by considering random rational points and by multiplying them by the cofactor $|G|/N$.

1.2.2 An asymptotical complexity bound for counting points on hyperelliptic curves of large genus

The results described in this section are part of the thesis of Simon Abelard. This is joint work with Simon Abelard and Pierrick Gaudry, and it has been published in the journal *Foundations of Computational Mathematics* [6].

Here we consider the point counting problem on a special family of abelian varieties: Jacobians of hyperelliptic curves of large genus. These special abelian varieties have several nice properties. First, there is a convenient embedding of such abelian varieties in $\mathbb{P}^g \times \mathbb{P}^g$ via the Mumford coordinates. This gives a compact way to represent rational points via $O(g)$ coordinates. Moreover, the group law on such abelian varieties can be computed very efficiently via *Cantor's algorithm* [27].

First, we need to define what is a hyperelliptic curve. We first state the definition, and we will define the terms used afterwards.

Definition 1.2.5. *A* hyperelliptic curve *over a perfect field k is the data of a geometrically irreducible curve C together with a map $\pi : C \to \mathbb{P}^1$ of degree 2. A hyperelliptic curve is called* imaginary *if the place at infinity of $\mathbb{P}^1$ ramifies, and it is called* real *otherwise.*

The map π (which is part of the data defining the hyperelliptic curve) defines a finite extension $k(X) \hookrightarrow k(C)$ of degree 2. A place of $\mathbb{P}^1$ is by definition the unique maximal ideal of a *Discrete Valuation Ring* (abbreviated DVR) in $k(X)$. Similarly, a place of C is the maximal ideal of a DVR in $k(C)$. The *place at infinity* of $\mathbb{P}^1$ is the maximal ideal $\mathfrak{m}_\infty = \{f/X^\ell : f \in k[X], \ell > 0, f(0) \neq 0\}$ of the DVR $k + \mathfrak{m}_\infty$. This place is said to *ramify* with respect to the field extension $k(X) \hookrightarrow k(C)$ if there is a unique place of $k(C) \otimes \overline{k}$ which contains $\mathfrak{m}_\infty \otimes \overline{k}$.

In practice, hyperelliptic curves are often manipulated via a convenient model which explicitly shows the degree-2 map, called the *Weierstrass equation*. This equation presents the hyperelliptic curve as a smooth curve in the affine plane, so that its affine coordinate ring $k[C]$ is a *Dedekind ring*. There is a construction that builds a finite multiplicative group from a Dedekind ring R, called its *class group*. Elements of this group are classes of fractional ideals $I/x \subset \mathrm{Frac}(R)$ — where $I \subset R$ is a nonzero ideal and $x \in R \setminus \{0\}$ — under the following equivalence relation: $I/x \sim I'/x'$ if and only if there exist $\alpha, \alpha' \in R$ such that $\alpha I = \alpha' I'$.

Proposition 1.2.6 (Weierstrass equation and Mumford coordinates)**.** *[30, Thm. 14.5] Every genus-g hyperelliptic curve over k is birationally equivalent to a smooth imaginary hyperelliptic curve $\mathscr{H}$ over a finite extension K of k in the affine plane defined by an equation of the form $Y^2 + h(X)Y = f(X)$, with $\deg(h) \leqslant g$, f monic and squarefree, $\deg(f) = 2g+1$. The map π in Theorem 1.2.5 is given by $(X, Y) \mapsto (X : 1)$.*

Then $K[\mathscr{H}] \stackrel{def}{=} K[X, Y]/(Y^2 + h(X)Y - f(X))$ is a Dedekind ring. For any fractional ideal I of $K[\mathscr{H}]$, there exist unique polynomials $u, v \in K[X]$ such that:

- *u is monic;*
- *$\deg(u) \leqslant g$;*
- *$\deg(v) < \deg(u) \leqslant g$;*
- *u divides $v^2 + h \cdot v - f$;*
- *The class of the ideal $\langle u(X), Y - v(X)\rangle$ equals that of I in the class group of $K[\mathscr{H}]$.*

The coefficients of the pair (u, v) (or by slight abuse of notation the pair itself) are called the Mumford coordinates *of I.*

Remark 1.2.7. *If the characteristic of the base field is not 2, then we can assume that $h(X) = 0$ because of the birational map $Y \mapsto Y + h(X)/2, X \mapsto X$. If we write $Y' = Y + h(X)/2$, $X' = X$, then X', Y' satisfy*

$$Y'^2 = f(X') + h(X')^2/4.$$

The smoothness assumption implies that the partial derivatives w.r.t X' and Y' cannot vanish simultaneously on the curve, and hence $f(X') + h(X')^2/4$ must be squarefree.

Remark 1.2.8. *When $\mathscr{H}$ is given via a Weierstrass equation, the group of K-rational points of $\mathrm{Jac}(\mathscr{H})$ is isomorphic to the class group of the Dedekind ring $K[\mathscr{H}]$. The isomorphism is given explicitly via the Mumford coordinates. In the sequel, we will use freely this identification between the rational points of the Jacobian variety and the class group of the affine coordinate ring.*

We consider Problem 1.2.3 specialized to the case of hyperelliptic curves defined over finite fields of large characteristic. Our goal is to study the complexity in terms of q for fixed large genus. In what follows, $\mathscr{H}$ is an imaginary hyperelliptic curve given in Weierstrass form. We also assume that q is not a power of two and that $h = 0$ in the Weierstrass equation.

The main result of this section is:

Theorem 1.2.9. *[6, Thm. 1] We give a probabilistic Las Vegas algorithm to compute the numerator of the zeta function of a genus-g hyperelliptic curve defined over a finite field $\mathbb{F}_q$ (given via an imaginary Weierstrass equation) with expected time bounded by $O_g(\log(q)^{cg})$, where c is an explicitly computable absolute constant and where $O_g(\cdot)$ means that the multiplicative constant depends on g.*

For fixed g sufficiently large, and large q, this result improves significantly the best previously known upper bounds, which were of the form $\log(q)^{cg^2 \log(g)}$ [7].

Our algorithmic framework follows the general ideas of Pila's and Schoof's algorithms: our goal is to compute efficiently the ℓ-torsion of the Jacobian of the curve for many small ℓ. Once we have access to the ℓ-torsion, we compute the action of the Frobenius endomorphism on it and we can deduce from this its characteristic polynomial modulo ℓ. Consequently, our main task is the efficient computation of the ℓ-torsion of the Jacobian of $\mathscr{H}$.

To this end, we use the *Cantor's polynomials*, which describe the image of the multiplication by ℓ of the generic point of the curve in its Jacobian. Before introducing Cantor's polynomials, we define a shifted variant:

Definition-Proposition 1.2.10. *[28] Let $\mathscr{H}$ be an imaginary hyperelliptic curve of genus g defined by a polynomial in Weierstrass form $Y^2 - f(X) \in \mathbb{F}_q[X, Y]$. Let $K = \mathrm{Frac}(\mathbb{F}_q[x, y]/(y^2 - f(x)))$ be the function field of the hyperelliptic curve. For $\ell > g$, there exist uniquely defined polynomials $\widetilde{u}_\ell, \widetilde{v}_\ell \in K[X]$ such that:*

- *$\widetilde{u}_\ell$ is monic and $\deg(\widetilde{u}_\ell) = g$;*
- *$\deg(\widetilde{v}_\ell) < g$;*
- *the class of $\langle \widetilde{u}_\ell(\frac{x-X}{4y^2}), Y - \widetilde{v}_\ell(\frac{x-X}{4y^2})\rangle$ is the same as the class of $\langle X - x, Y - y\rangle^\ell$ in the class group of $\mathbb{F}_q[\mathscr{H}] \otimes_{\mathbb{F}_q} K$.*

Note that it is not restrictive to ask $\ell > g$: for $\ell \leqslant g$, the class of $\ell \cdot (P - \infty)$ in $\mathrm{Jac}(\mathscr{H})$ is easy to describe with Mumford coordinates since $\deg(\ell \cdot P) \leqslant g$.

Cantor's polynomials $\delta_\ell, \varepsilon_\ell$ are shifted versions of $\widetilde{u}_\ell, \widetilde{v}_\ell$ in order to obtain easier polynomial expressions. More precisely:

$$\begin{aligned} \delta_\ell(\tfrac{x-X}{4y^2}) &= \gamma\widetilde{u}_\ell, \\ \varepsilon_\ell(\tfrac{x-X}{4y^2}) &= \widetilde{v}_\ell, \end{aligned}$$

for some $\gamma \in K$ which is described in Cantor's paper [28].

The coefficients of the polynomials $\widetilde{u}_\ell, \widetilde{v}_\ell$ are elements in $K = \mathrm{Frac}(\mathbb{F}_q[x,y]/(y^2 - f(x)))$ which can be expressed as *polynomials* in x and y. The next proposition bounds the degrees of such polynomials, which is crucial for obtaining our complexity bounds:

Proposition 1.2.11. *[6, Lem. 10] There exist polynomials $d_0, \ldots, d_{g-1}, d_g \in \mathbb{F}_q[T]$ of degree bounded by $\frac{1}{3}g\ell^3 + O_g(\ell^2)$ such that*

$$\delta_\ell(X) = \sum_{i=0}^{g} d_i(x) X^i.$$

There exists polynomials $e_0, \ldots, e_g \in \mathbb{F}_q[T]$ of degree bounded by $\frac{2}{3}g\ell^3 + O_g(\ell^2)$ such that

$$\varepsilon_\ell(X) = y \sum_{i=0}^{g-1} \frac{e_i(x)}{e_g(x)} X^i.$$

Furthermore, all roots of e_g are roots of d_g.

Remark 1.2.12. *Experiments show that these bounds are not optimal. Getting tighter bounds would not have much impact on the results in this section. However, the problem of having tighter bounds is related to questions arising when we consider non-hyperelliptic curves. Such topics are mentioned in my research project, see Section 3.1.*

Computing the ℓ-torsion. Now that we know bounds on the degrees of the coefficients of Cantor's polynomials, we can use them to compute efficiently the ℓ-torsion of $\mathrm{Jac}(\mathscr{H})$. Cantor's polynomials allow us to a have a description of $\ell \cdot (P - \infty)$ for a generic point P on $\mathscr{H}$ (where ∞ is the unique place at infinity of a hyperelliptic curve given by an imaginary Weierstrass model). A generic element of $\mathrm{Jac}(\mathscr{H})$ is given by a set of g generic points $\{P_1, \ldots, P_g\}$ on $\mathscr{H}$. Such a set of points belongs to the ℓ-torsion of $\mathrm{Jac}(\mathscr{H})$ if the divisor $\sum_{1 \leqslant i \leqslant g} \ell(P_i - \infty)$ is principal. By using Cantor's polynomials, we get a description of $\ell(P_i - \infty)$ for each i. The sum is principal if only if there exists a regular function on $\mathscr{H} \setminus \infty$ whose zeros are the g^2 points that are provided by the Cantor's polynomials. We can bound the degree of polynomials representing this function and introduce new indeterminates for their coefficients. Under genericity assumptions, using this observation we can build a polynomial systems whose zeros correspond to the coordinates of the sets of points $\{P_1, \ldots, P_g\}$ which describe ℓ-torsion points. In order to obtain sufficiently tight complexity bounds, we had to exploit structural properties of this polynomial system, in particular its bi-homogeneity. To do so, we relied on the *geometric resolution* algorithmic framework, see [60].

The genericity properties that we need are summed up in the following definition:

Definition 1.2.13 (ℓ-generic divisor)**.** *[6, Def. 11] The weight of a divisor class $[D]$ is the degree of its u-polynomial. Let $[D]$ be a weight-g divisor class and write $[D] = \sum_{i=1}^{g} [P_i - \infty]$*

over an algebraic closure of $\mathbb{F}_q$. *Equivalently,* $P_i = (x_i, y_i) \in \overline{\mathbb{F}_q}^2$ *are such that we have the following equality of ideals in* $\mathbb{F}_q[\mathscr{H}] \otimes \overline{\mathbb{F}_q}$*:*

$$\langle u_{[D]}(X), Y - v_{[D]}(X) \rangle = \prod_{i=1}^{g} \langle X - x_i, Y - y_i \rangle$$

.

We say that $[D]$ *is* ℓ*-generic if*

- $[D]$ *has weight* g*;*
- *for each* $i \in \{1, \ldots, g\}$, $\ell \cdot [P_i - \infty]$ *has weight* g*;*
- *for each* $i, j \in \{1, \ldots, g\}$, $i \neq j$, *the* u*-polynomials of* $\ell \cdot [P_i - \infty]$ *and* $\ell \cdot [P_j - \infty]$ *are coprime.*

Our main point is that under the ℓ-generic assumptions, the ℓ-torsion of $\mathrm{Jac}(\mathscr{H})$ can be computed by building a polynomial system from the Cantor's polynomials. We represent an element of $\mathrm{Jac}(\mathscr{H})$ as a list of g points on $\mathscr{H}$. This is well-defined for weight-g points in the Jacobian for which the uniquely defined representation as a sum of g points involves only distinct points, which is ensured in our definition of ℓ-generic elements. Next, we encode the multiplication by ℓ by using evaluating the ℓ-Cantor's polynomials at our g points. This provides us with g points in the Jacobian, encoded via Mumford coordinates. Under the assumptions that all these g points in the Jacobian have weight g, this provides us with g^2 points on $\mathscr{H}$. Under the ℓ-generic assumptions that the u-polynomials of the ℓ-multiples are coprime, these g^2 points are distinct, and none of them is the opposite of another one. Therefore, testing whether their sum vanishes in the Jacobian amounts to testing if there exists a function of the hyperelliptic curve which has a pole of multiplicity g^2 at infinity and zeros at these g^2 points. Using projective coordinates, we can write such a function by using indeterminate coefficients, since its degree is bounded by the condition on the number of zeros and poles.

We end up with a polynomial system whose indeterminates are the coordinates of the g points on $\mathscr{H}$ and the unknown coefficients of the function which asserts the vanishing of the sum of the ℓ-multiples in the Jacobian.

A crucial fact is that the variables corresponding to the points and those corresponding to the coefficients of the function provide us with a partition that gives a bi-homogeneous structure to the system.

The last ingredient is to use Bertini's theorem in order to transform the input system into a system which is a regular sequence. Although this transformation loses some information and creates extra solutions, this allows us to use complexity results on polynomial system solving by using the geometric resolution algorithm [60]. Indeed, the complexity is bounded by a quantity which is computed from multi-homogeneous Bézout bounds. Since the extra solutions can be filtered out afterwards by checking if they correspond to points in the kernel of the multiplication-by-ℓ endomorphism, they do not have any impact on the correctness of the algorithm.

This allows us to compute ℓ-generic divisors in the ℓ-torsion of $\mathrm{Jac}(\mathscr{H})$. Although we expect that computing ℓ-generic ℓ-torsion divisors should be sufficient in most cases to compute the ℓ-torsion for sufficiently-many ℓ in order to apply Schoof's method, we were unable to prove this. Therefore, we had to build specific polynomial systems to encode the ℓ-torsion divisors which do not satisfy the ℓ-generic properties.

The complexity for computing the ℓ-torsion under these genericity assumptions is summed up in the following statement:

Proposition 1.2.14. *[6, Prop. 12] For any $\varepsilon > 0$, there is a constant C such that for all primes $\ell > g$ coprime to q, there is a Monte-Carlo algorithm which computes an $\mathbb{F}_{q^e}$-geometric resolution of the subvariety of $\mathrm{Jac}(\mathscr{H})[\ell]$ of ℓ-generic ℓ-torsion elements of the Jacobian variety of $\mathscr{H}$, where $e = O_g(\ell)$. The time and space complexities of this algorithm are bounded by $O_g\left(\ell^{Cg}(\log q)^{2+\varepsilon}\right)$, and it returns the correct result with probability at least $5/6$.*

Computing the zeta function from the description of the ℓ-torsion. Once we have a convenient description of the ℓ-torsion, computing the action of the Frobenius endomorphism is quite easy. For instance, we can compute a basis $(b_1, \ldots, b_{2g})$ of the ℓ-torsion as a $\mathbb{Z}/\ell\mathbb{Z}$-vector space of dimension $2g$, and then compute the decompositions of $(b_1^q, \ldots, b_{2g}^q)$ with respect to this basis.

Nongeneric cases. In order to prove our main complexity result, we need that sufficiently-many ℓ-torsion divisor classes satisfy the genericity assumptions in Definition 1.2.13. Experimental results seem to indicate that these assumptions are almost always satisfied and that this is sufficient is practice so that the polynomial systems that we designed faithfully represent the ℓ-torsion. However, we were not able to prove that the nongeneric cases are few enough to have no impact on our main complexity results.

Therefore, to prove our main theorem (Theorem 1.2.9), we had to model specific polynomial systems for all nongeneric cases that can appear. This way, we cover all possible cases, and this completes our proof of the main complexity result. Studying all the possible non-generic cases leads to a tedious and technical analysis, which we do not detail here (see [6, Sec. 5] for technical details).

1.2.3 Genus-3 hyperelliptic curves with explicit real multiplication

This section presents results that have been obtained with Simon Abelard and Pierrick Gaudry, and they are part of the Ph.D. thesis of Simon Abelard. They have been published in the proceedings of the *Thirteenth Algorithm Number Theory Symposium 2018* [5].

Real multiplication for hyperelliptic curves has a long history, as this topic was already studied by Georges Humbert at the end of the 19th century. Our main motivation comes from cryptography, where rational points on Jacobian varieties of elliptic curves or genus-2 and genus-3 hyperelliptic curves provide finite groups that have nice cryptographic properties, provided that the order of the group is divisible by a sufficiently large prime. Point-counting algorithms provide tools to compute such group orders. Over a finite field $\mathbb{F}_q$ of large characteristic, Schoof's algorithm and its variants provide polynomial-time algorithms for fixed genus. However the exponent w.r.t. $\log(q)$ is large, and this implies that practical computations are often challenging.

One way to leverage this issue is to pick curves which are structured in a way that helps point-counting algorithms, and which hopefully does not hinder the cryptographic properties. One way to achieve that is to consider *curves with explicit real multiplication*, namely curves for which we know explicitly a non-scalar endomorphism which generates a totally real field over $\mathbb{Q}$. These curves are often obtained as reductions of known families of curves over $\mathbb{Q}$ which have real multiplication, see e.g. [81].

This was used in [56] in order to construct large cryptographic genus-2 curves. Our goal in this section is to show how to use these techniques for genus-3 hyperelliptic curves. The algorithm that we present in this section allowed us to compute the zeta function of a genus-3 hyperelliptic curve with explicit real multiplication over the field $\mathbb{F}_{2^{64}-59}$. The previous record

for hyperelliptic genus-3 point counting (without explicit real multiplication) was computed by Andrew Sutherland over the field $\mathbb{F}_{2^{61}-1}$ by using generic group methods [112].

Explicit real multiplication. In this work, our main goal is to develop techniques to compute points in practice for genus-3 hyperelliptic curves. Since the complexity becomes quickly prohibitive, we consider curves with *explicit real multiplication*, which are curves for which we know a part of the endomorphism ring. The extra structure allows us to speed up the computations.

Let $\mathscr{H}$ be a genus-3 imaginary hyperelliptic curve, and let $\eta \in \mathrm{End}(\mathrm{Jac}(\mathscr{H}))$ be an endomorphism whose minimal polynomial over $\mathbb{Q}$ has degree 3 and has only real roots. We say that it has *explicit real multiplication* by $\mathbb{Z}[\eta]$ if we have explicit formulas for computing $\eta([P-\infty])$ for a point $(x,y) \in \mathscr{H}(\overline{\mathbb{F}_q})$.

By explicit formulas, we mean analog of Cantor's polynomials for the multiplication by ℓ: for $i \in \{0,1,2,3\}$, polynomials $\eta_i^{(u)} \in \mathbb{F}_q[X]$, $\eta_i^{(v)} \in \mathbb{F}_q[X,Y]$ such that for the generic point (x,y) of $\mathscr{H}$ the Mumford coordinates of $\eta([(x,y)-\infty])$ are

$$\left(\sum_{i=0}^{3} \eta_i^{(u)}(x)X^i, \sum_{i=0}^{2} \frac{\eta_i^{(v)}(x,y)}{\eta_3^{(v)}(x,y)}X^i\right).$$

Our main complexity result is:

Theorem 1.2.15. *[5, Thm. 1] Let $\mathscr{H}$ be a genus-3 hyperelliptic curve over a finite field $\mathbb{F}_q$ (where q is odd) given by an imaginary Weierstrass model, and $\eta \in \mathrm{End}(\mathrm{Jac}(\mathscr{H}))$ be an endomorphism given by rational functions such that $\mathbb{Z}[\eta]\otimes\mathbb{Q}$ is a real cubic number field. Then we present a probabilistic Las Vegas algorithm to compute the characteristic polynomial of the Frobenius endomorphism on $\mathrm{Jac}(\mathscr{H})$ in expected time bounded by $c(\log q)^6(\log\log q)^k$, where k is an absolute constant, and c depends on the degrees of the polynomials describing η and on the ring $\mathbb{Z}[\eta]$.*

Remark 1.2.16. *We abbreviate the complexity as $\widetilde{O}_\eta((\log q)^6)$, where the subscript η indicates that the constant hidden in the $\widetilde{O}$ depends on the data associated to η.*

The main technique is to use the fact that η generates an order $\mathbb{Z}[\eta]$ in a totally real number field of degree 3 in the endomorphism algebra of the Jacobian of $\mathscr{H}$. If $\ell\mathbb{Z}[\eta]$ decomposes as a product of ideals in $\mathbb{Z}[\eta]$, then the ℓ-torsion may be decomposed accordingly. We consider split prime numbers ℓ where $\ell\cdot\mathbb{Z}[\eta] = \mathfrak{p}_1\mathfrak{p}_2\mathfrak{p}_3$. Then $\mathrm{Jac}(\mathscr{H})[\mathfrak{p}_i]$ is the set of points where all endomorphisms in $\mathfrak{p}_i$ simultaneously vanish. This kernel contains a $\mathbb{Z}/\ell\mathbb{Z}$-vector space of dimension 2, instead of the usual ℓ^6 ℓ-torsion points. This is where we obtain a complexity improvement.

Still, there is an obstacle: usually we want to compute the action of the Frobenius endomorphism π on the ℓ-torsion, but π does not let the kernel subspaces $\mathrm{Jac}(\mathscr{H})[\mathfrak{p}_i]$ invariant. In order to leverage this issue, we study the action of $\pi+\pi^\vee$ (where $\pi^\vee$ is the dual of the Frobenius endomorphism), instead of the action of π on these kernels. Indeed, it is known that $\psi \stackrel{\text{def}}{=} \pi+\pi^\vee$ belongs to the degree-3 real subfield of $\mathbb{Q}(\pi)$. Consequently, $\psi \in \mathbb{Q}[\eta]$ and therefore there exist rational numbers a,b,c such that $\psi = a+b\eta+c\eta^2$. In fact, the common denominator of a,b,c must divide the index of $\mathbb{Z}[\eta]$ in the maximal order of $\mathbb{Q}[\eta]$.

Moreover, there is a known formula relating the characteristic polynomial χ_ψ of ψ regarded as an element in the degree-3 real subfield of $\mathbb{Q}[\pi]$, and the characteristic polynomial of π in $\mathbb{Q}[\pi]$: $\chi_\pi(T) = T^3\chi_\psi(T+q/T)$, see e.g. [68]. The characteristic polynomial χ_ψ is defined globally

as usual (by noticing that all its specialization on the Tate modules are compatible) and it has integer coefficients. If we are able to compute the rational numbers a, b, c modulo ℓ (provided ℓ does not divide Δ), then we have access to χ_ψ modulo ℓ. The last ingredient is that ψ acts on the 2-dimensional $\mathbb{Z}/\ell\mathbb{Z}$-vectors spaces in the kernel subspaces $\mathfrak{p}_i$ as the multiplication by a scalar.

Our global strategy is then the following:

- We start by computing "small" elements $\alpha_i = a_i + b_i\eta + c_i\eta^2$ in each of the prime ideals $\mathfrak{p}_i \supset \ell\mathbb{Z}[\eta]$, where $a_i, b_i, c_i \in \mathbb{Z}$. By small, we mean that it has norm $\lambda\ell$ for some small cofactor λ.

- These elements correspond to endomorphisms α_i whose Cantor's polynomials can be computed from the Cantor's polynomials for η.

- Using these Cantor's polynomials, we compute their kernels $K_i \stackrel{\text{def}}{=} \mathrm{Jac}(\mathscr{H})[\alpha_i]$ by proceeding as in Section 1.2.2. In the genus-3 case, the dimension of the Jacobian variety is small so we do not need to use general-purpose solving algorithms; using resultants, we manage to get more precise complexity bounds.

- The abelian group $G_i \stackrel{\text{def}}{=} K_i \cap \mathrm{Jac}(\mathscr{H})[\ell]$ is then a 2-dimensional $\mathbb{Z}/\ell\mathbb{Z}$ vector space. The endomorphism η acts as a scalar multiplication on this subspace, and the corresponding eigenvalue λ_i can be computed from the 2-element representation of $\mathfrak{p}_i$. On the other hand, ψ also acts on G_i as a scalar multiplication by an unknown integer k_i. Unfortunately, we cannot find directly k_i since we are not able to evaluate ψ. Therefore, we use a trick which uses the relationship between ψ and π: $\pi\psi = \pi^2 + \pi\pi^\vee = \pi^2 + q$. Since the righthand side on this equality involves only the Frobenius endomorphism and the multiplication-by-q map, we can evaluate it. Consequently, we are able to evaluate $\pi\psi$ on G_i, and therefore we can pick some nonzero point $D_i \in G_i$ and compute k_i as the only integer such that $\pi\psi(D) = k_i\pi(D_i)$.

- From λ_i and k_i, we obtain the following relation on the searched rational numbers a, b, c: $a + b\lambda_i + c\lambda_i^2 \equiv k_i \bmod \ell_i$.

- By using this approach for sufficiently many small ℓ which split in $\mathbb{Q}[\eta]$, and by using bounds that we can obtain on a, b, c from Weil's conjectures, the CRT allows us to recover the integers a, b, c. From these values, we get the characteristic polynomial of the Frobenius endomorphism.

In order to compute the asymptotic complexity in a tighter way than in Section 1.2.2 and obtain Theorem 1.2.15, we use the fact that the low dimension of Jacobian variety allows us to compute the kernel subspaces by using resultants instead of more general techniques such as Gröbner bases or geometric resolutions whose complexity analyses require extra regularity assumptions which can be only be acquired at some cost.

An explicit computation. As a proof-of-concept for our method, we consider the curves obtained via the reduction modulo a prime of the genus-3 hyperelliptic curve $\mathscr{H}$ defined over $\mathbb{Q}$ by the equation $y^2 = x^7 - 7x^5 + 14x^3 - 7x + t$, where $t \neq \pm 2$. This curve has explicit real multiplication by an endomorphism η_7 which is such that $\mathbb{Z}[\eta_7] \subset \mathrm{End}(\mathrm{Jac}(\mathscr{H}))$ is isomorphic to $\mathbb{Z}[2\cos(2\pi/7)]$. Notice that $\mathbb{Z}[2\cos(2\pi/7)] \otimes \mathbb{Q}$ is a totally real degree-3 extension of $\mathbb{Q}$.

Cantor's polynomials $U_{\eta_7}(x, y, X), V_{\eta_7}(x, y, X) \in \mathbb{Q}(\mathscr{H})[X]$ for η_7 are given by

$$\begin{aligned} U(x,y,X) &= X^2 + 11xX/2 + x^2 - 16/9, \\ V(x,y,X) &= y \end{aligned}$$

This family of curves with explicit real multiplication was found in [113].

For our computation, we instantiate this curve with $t = 42$ and we reduce it modulo $p = 2^{64} - 59$. For kernel computations, we used the implementation of Gröbner bases (instead of the resultants that were used in the complexity analysis) in the Magma computer algebra software v2.23-4. Also, we do not restrict our computations to split ℓ. In particular, the primes 2 and 3 are inert, but the 2-torsion can be obtained for free and the 3-torsion can be computed without making use of the real multiplication. Also, 7 ramifies completely in $\mathbb{Z}[2\cos(2\pi/7)] \otimes \mathbb{Q}$ and this can be used to recover one linear relation between a, b, c modulo 7. The first split prime is 13, and we used the real multiplication to compute a, b, c modulo 13 by using the strategy described in Section 1.2.3. The computation for each kernel subspace lasted three days, and it required 41GB of RAM on a Xeon E7-4850v3 at 2.20GHz, with 1.5 TB RAM. Such a computation would not have been possible without using the real multiplication, since the 13-torsion is too large to be computed directly (it has order $13^6 \approx 2^{22}$).

When sufficiently-many modular information on a, b, c is known, the computation is finished by using a multi-dimensional kangaroo-type low memory parallel collision search, see [57].

Once we find the coefficients a, b, c, we get the relation

$$\psi = 2551309006 + 2431319810\eta_7 - 847267802\eta_7^2,$$

from which we can recover the characteristic polynomial of the Frobenius endomorphism, which is:

$$\chi_\pi(T) = T^6 - \sigma_1 T^5 + \sigma_2 T^4 - \sigma_3 T^3 + q\sigma_2 T^2 - q^2\sigma_1 T + q^3,$$
$$\sigma_1 = 986268198, \quad \sigma_2 = 35389772484832465583, \quad \sigma_3 = 109560528621042368187702122 44.$$

1.3 Isogenies of Drinfeld modules and effective class field theory of hyperelliptic function fields

This section presents results that have been obtained with Antoine Leudière, and they are part of the Ph.D. thesis of Antoine Leudière. This work has been published in the Journal of Symbolic Computation [87].

1.3.1 Motivation

Drinfeld modules — initially called elliptic modules — were introduced by Vladimir Drinfeld in [41]. One of the goal was to mimic the class field theory of imaginary quadratic number fields, by constructing geometric objects which are related to abelian extensions of function fields. This is related to the Langlands conjectures for function fields, and this line of work culminated when Laurent Lafforgue developed the theory to the point of proving parts of Langlands' conjectures for function fields.

As Drinfeld modules share many similarities with elliptic curves, we may ask whether the known algorithmic and cryptographic applications of elliptic curves may be translated in the world of Drinfeld modules. A decade ago, computer algebraists have started the investigation of Drinfeld modules from the point of view of modern computer algebra. One nice feature of Drinfeld modules is that they provide efficient algorithms for factoring univariate polynomials over finite fields [39].

Another viewpoint — that we shall adopt in this section — is that (isomorphism classes) of elliptic curves are related to the class field theory of imaginary quadratic number fields, via the theory of *complex multiplication*. Similarly, isomorphism classes of rank-2 Drinfeld modules are related to arithmetic properties of Jacobians of hyperelliptic curves, via a similar theory of complex multiplication.

One of the fundamental algorithmic feature of the theory of complex multiplication is that there is an action of a class group of a set of isomorphism classes, and this action can be represented via *isogenies*, i.e. morphisms between geometric objects.

Our initial motivation was to design a Drinfeld module analog of the CRS cryptosystem [31, 103], which relies on this group action on isomorphism classes of elliptic curves. However, Wesolowski showed that isogenies between Drinfeld modules can be computed efficiently [121], which hinders the cryptographic potential. Nevertheless, this group action is an important feature of the class field theory of hyperelliptic function fields, and designing algorithms to compute it is a nice tool to have in the quickly growing algorithmic toolbox for Drinfeld modules.

Similarly to elliptic curves, Drinfeld modules have both an analytic and an algebraic description. The algebraic construction relies on the noncommutative ring of *Ore polynomials*.

Definition 1.3.1 (Ore polynomials)**.** *Let $\mathbb{F}_q \hookrightarrow K$ be a field extension. The ring of* Ore polynomials *$K\{\tau\}$ is the K-linear subspace of $K[X]$ generated by $\{X, X^q, X^{q^2}, \ldots\}$ equipped with the usual addition and the composition as multiplication law. We use the shorthand notation $\tau^i \stackrel{def}{=} X^{q^i}$. The ring $K\{\tau\}$ is isomorphic (as a $\mathbb{F}_q$-algebra) to the ring $\mathrm{End}_{\mathbb{F}_q}(\overline{K})$ of $\mathbb{F}_q$-linear endomorphisms of the affine scheme $\mathbb{A}^1(K)$ endowed with its natural structure of $\mathbb{F}_q$-vector space K-scheme.*

The category of $\mathbb{F}_q$-vector space K-schemes can be defined as follows: an object is a K-scheme S equipped with a functor $F : (\mathsf{AffineSchemes})^{\mathsf{op}} \to \mathbb{F}_{\mathsf{q}}\text{–}\mathsf{vectorspaces}$ from the category of K-schemes to the category of $\mathbb{F}_q$-vector spaces such that $F = \mathrm{Hom}(\_, S)$; A morphism $(S_1, F_1) \to (S_2, F_2)$ is the data (φ, ψ) where φ is a morphism of K-schemes $\varphi : S_1 \to S_2$ and ψ is a natural transformation from F_1 to F_2.

Remark 1.3.2. *Historically, authors often use the notation $\mathbf{G}_a(K)$ for $\mathbb{A}^1(K)$ endowed with the structure of $\mathbb{F}_q$-vector space scheme. This notation refers only to the additive group scheme structure. However, as mentioned in [99, Sec. 3.1], considering only the group scheme structure instead of the $\mathbb{F}_q$-vector space scheme structure loses a bit of structure since the endomorphism ring of $\mathbf{G}_a(K)$ in the category of group schemes is slightly different compared to what we want: if $q = p^r$ with $r > 1$ then $\mathrm{End}(\mathbf{G}_a(K))$ contains for instance the p-Frobenius, which is additive but not $\mathbb{F}_q$-linear.*

Definition 1.3.3. *[96, Def. A.2] Let F be a finite extension of $\mathbb{F}_q(X)$, ∞ be a place of F, and let $A \subset F$ be the ring of functions which are regular (i.e. which have nonnegative valuation) at all places distinct from ∞. An A-*field *(K, γ) is the data of a field extension $\mathbb{F}_q \hookrightarrow K$ together with a morphism $\gamma : A \to K$.*

A Drinfeld module *over the A-field (K, γ) is a homomorphism of $\mathbb{F}_q$-algebras $\phi : A \to K\{\tau\}$ such that*

- *$D \circ \phi = \gamma$, where $D : K\{\tau\} \to K$ is the map that sends $\sum_{i=0}^{\ell} a_i \tau^i$ to a_0;*
- *there exists $a \in A$ such that $\phi(a) \neq \gamma(a)$.*

Classically, we write ϕ_a for $\phi(a)$.

We now define the category of Drinfeld modules over a given A-field.

Definition 1.3.4. *[96, Def. A.4] Let ϕ, ϕ' be Drinfeld modules over the A-field (K, γ). A morphism (defined over K) $\phi \to \phi'$ is an Ore polynomial $\iota \in K\{\tau\}$ such that $\iota \cdot \phi_a = \phi'_a \cdot \iota$ for all $a \in A$. A nonzero morphism is called an* isogeny*.*

Drinfeld modules have a rank, which is defined as follows

Definition 1.3.5. *[96, Def. A.6] For any Drinfeld module ϕ, there is an integer $r \in \mathbb{Z}_{\geqslant 0}$ such that $\deg_\tau(\phi_a) = r \deg(a)$ for all $a \in A$, where $\deg(a) = \log_q|A/(a)|$. The integer r is called the* rank *of ϕ.*

The rank of a Drinfeld module encode its rank as a A-module in the analytic description. Therefore, the closest analogs to elliptic curves are Drinfeld modules of rank 2 with $A = \mathbb{F}_q[X]$, since they mimic elliptic curves which can be regarded analytically as rank-2 $\mathbb{Z}$-modules in $\mathbb{C}$.

1.3.2 Computing a group action from the class field theory of hyperelliptic function fields

In this section, we describe the algorithms that we designed with Antoine Leudière to compute the action of the Jacobian of a hyperelliptic curve $\mathscr{H}$ on the set of isomorphism classes of Drinfeld modules which have complex multiplication by the function field of $\mathscr{H}$.

The Frobenius endomorphism and complex multiplication of finite Drinfeld modules. Similarly to elliptic curves, Drinfeld modules have a rich theory of complex multiplication. Following [59], we call a Drinfeld module *finite* if K is a finite extension of $\mathbb{F}_q$. In that case, it is usual to use the letter L instead of K to denote the field of definition for the Drinfeld modules.

Remark 1.3.6. *In fact, for computational purposes, a convenient setting is to start with a prime ideal $\mathfrak{p} \subset \mathbb{F}_q[X]$, to consider a finite extension L of $\mathbb{F}_q[X]/\mathfrak{p}$ and the $\mathbb{F}_q[X]$-field (L, γ), where $\gamma : \mathbb{F}_q[X] \to L$ is the composition of the canonical projection $\mathbb{F}_q[X] \twoheadrightarrow \mathbb{F}_q[X]/\mathfrak{p}$ with the inclusion $\mathbb{F}_q[X]/\mathfrak{p} \hookrightarrow L$.*

An interesting property of finite Drinfeld modules is that they always have a nontrivial endomorphism: the *Frobenius endomorphism* $\tau_L \stackrel{\text{def}}{=} \tau^{[L:\mathbb{F}_q]}$. Similarly to elliptic curves defined over finite fields, rank-2 finite Drinfeld modules with $A = \mathbb{F}_q[X]$ are categorized as *ordinary Drinfeld modules* or *supersingular Drinfeld modules* depending on the $\mathbb{F}_q[X]$-rank of their endomorphism ring. In this section, we focus only on the ordinary case, i.e. when the endomorphism ring has rank-2 as a $\mathbb{F}_q[X]$-module and is an order in a imaginary quadratic extension of $\mathbb{F}_q(X)$.

A nice feature of the Frobenius endomorphism is that it has a "characteristic polynomial" [96, Sec. 4.2], which behaves well in the ordinary case. If $\mathfrak{l}$ is a prime ideal in A, the Tate $\mathfrak{l}$-module is the set of points $x \in \overline{\mathbb{F}_q}$ for which there exists some $i \in \mathbb{Z}_{\geqslant 0}$ s.t. $\phi_a(x) = 0$ for all $a \in \mathfrak{l}^i$. The Tate $\mathfrak{l}$-module is a free module of rank r over the completion of the local ring $A_\mathfrak{l}$. Therefore any endomorphism has a characteristic polynomial over the Tate $\mathfrak{l}$-module, which has degree r and coefficients in the completion of $A_\mathfrak{l}$. By [96, Thm. 3.6.6] the coefficients of this characteristic polynomial are all in A and their values do not depend on $\mathfrak{l}$. The following states this property in the case of the Frobenius endomorphism:

Theorem 1.3.7. *[96, Thm. 4.2.2] [96, Cor. 4.1.12] Let ϕ be an ordinary finite Drinfeld module of rank r. Then the minimal polynomial $\xi \in A[Y]$ of τ_L in the ring extension $A \hookrightarrow \mathrm{End}(\phi)$ is a degree-r polynomial, which equals the characteristic polynomial of the Frobenius endomorphism on the Tate $\mathfrak{l}$-modules.*

Therefore, to a finite Drinfeld module ϕ of rank 2 defined over $A = \mathbb{F}_q[X]$ is associated a characteristic polynomial $\xi = Y^2 + t(X)Y + n(X) \in \mathbb{F}_q[X, Y]$. Degree bounds on t and n are known: these bounds are similar to the Weil bounds on the trace of the Frobenius endomorphism for elliptic curves. In particular, these bounds imply that the place at infinity of $\mathbb{F}_q(X)$ ramifies in the field extension $\mathrm{Frac}(\mathbb{F}_q[X, Y]/\xi)$ when $[L : \mathbb{F}_q]$ is odd. Consequently, ξ defines a imaginary hyperelliptic curve $\mathscr{H}$, provided that there is no singularity in the affine plane.

In what follows, we shall assume that this ξ defines a hyperelliptic curve. We emphasize that this happens with large probability, in particular when q is large: there is a discriminant associated to ξ which must be squarefree.

Next, we notice that under this assumption, $\mathbb{F}_q[X, \tau_K] \cong \mathbb{F}_q[X, Y]/\xi = \mathbb{F}_q[\mathscr{H}]$ is precisely the endomorphism ring of the Drinfeld module, and its ideals correspond to isogenies: to an ideal $I \subset \mathbb{F}_q[X, Y]/\xi$ we associate the rgcd in $L\{\tau\}$ of the elements $\{f(\phi_X, \tau_L) : f \in I\}$, which is an isogeny to another Drinfeld module, which is uniquely defined up to isomorphism. Notice that if I is principal, then the rgcd is actually an endomorphism. This action of ideals extends to a simply transitive action — noted $\star_L$ — of the class group of $\mathbb{F}_q[\mathscr{H}]$ on the set of isomorphism classes of finite Drinfeld modules isogenous to ϕ.

The last ingredient that we need is the fact that rank-2 Drinfeld modules over $\mathbb{F}_q[X]$ with complex multiplication by $\mathbb{F}_q[\mathscr{H}]$ can be regarded as Drinfeld modules of rank 1 over the ring $\mathbb{F}_q[\mathscr{H}]$. In the sequel of the section, we let $\mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi$ denote the set of rank-2 Drinfeld modules over A with coefficients in L and whose characteristic polynomial of the L-Frobenius endomorphism is ξ. We also let $\mathrm{Dr}_1(\mathbb{F}_q[\mathscr{H}], L)$ denote the set of rank-1 Drinfeld modules over $\mathbb{F}_q[\mathscr{H}]$ with coefficients in L.

A group action over rank-1 Drinfeld modules is already described in the literature in the case where L is a function field, and it appears that the case of finite Drinfeld modules can be obtained by reduction modulo primes [66, Thm. 9.3].

Our main result is:

Theorem 1.3.8. *[87, Thm. 2.7] If* $\mathrm{Dr}_1(\mathbb{F}_q[\mathscr{H}], L)$ *is nonempty, then the set of* $\overline{L}$*-isomorphism classes of Drinfeld modules in* $\mathrm{Dr}_1(\mathbb{F}_q[\mathscr{H}], L)$ *is a principal homogeneous space for* $\mathrm{Cl}(\mathbb{F}_q[\mathscr{H}]) \cong \mathrm{Pic}^0(\mathscr{H})$ *under the* $\star_L$ *action.*

The general framework of the proof of Theorem 1.3.8 is to connect the result to the case of Drinfeld modules over function fields [66, Thm. 9.3] via reduction theory [63, Sec. 4.10] and lifting [9, Thm. 3.4].

1.3.3 Algorithms

In this section, we describe algorithms to compute and invert the group action. More precisely, $\star_L$ lets the class group of $\mathbb{F}_q[\mathscr{H}]$ act faithfully and transitively on the set of L-isomorphism classes of rank-2 Drinfeld modules over L. We emphasize that we have natural data structures to represent these objects: elements in the class group of $\mathbb{F}_q[\mathscr{H}]$ can be represented via Mumford coordinates, i.e. pairs of polynomials (u, v), and L-isomorphism classes of rank-2 Drinfeld modules can be encoded via their j-invariant. In fact the j-invariant (which belongs to L) classifies the $\overline{L}$-isomorphism classes, not L-isomorphism classes. However, two Drinfeld modules are L-isomorphic if and only if they are $\overline{L}$-isomorphic and their Frobenius endomorphisms have the same characteristic polynomial. Therefore, once the characteristic polynomial ξ of the Frobenius endomorphism is fixed, the j-invariant is a sufficient data to characterize a L-isomorphism class.

Computation of the group action

One of our algorithmic contribution is to present an algorithm that computes efficiently the map

$$\begin{array}{ccc} \mathrm{Cl}(\mathbb{F}_q[\mathscr{H}]) \times (\mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi / L\text{-isomorphisms}) & \to & (\mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi / L\text{-isomorphisms}) \\ ((u, v), j) & \mapsto & (u, v) \star_L j. \end{array}$$

Algorithm 1 GroupAction

1: **function** GroupAction(A j-invariant $j \in L$ encoding an isomorphism class $\mathcal{C}$ of Drinfeld modules in $\mathrm{Dr}_1(\mathbb{F}_q[\mathscr{H}], L)$, and Mumford coordinates $(u, v) \in \mathbb{F}_q[X]^2$ for a divisor class $[D]$ in $\mathrm{Pic}^0(\mathcal{H})$.)
2: $\quad \widetilde{u} \leftarrow u(j^{-1}\tau^2 + \tau + \omega) \in L\{\tau\}$
3: $\quad \widetilde{v} \leftarrow v(j^{-1}\tau^2 + \tau + \omega) \in L\{\tau\}$
4: $\quad \iota \leftarrow \mathrm{rgcd}(\widetilde{u}, \tau_L - \widetilde{v})$ $\quad \triangleright\ \iota = \sum_{0 \leqslant k \leqslant \deg_\tau(\iota)} \iota_k \tau^k$ $\widehat{g} \leftarrow \iota_0^{-q}(\iota_0 + \iota_1(\omega^q - \omega))$
5: $\quad \widehat{\Delta} \leftarrow j^{-q^{\deg_\tau(\iota)}}$
6: $\quad$ Return $\widehat{g}^{q+1}/\widehat{\Delta}$.
7: $\quad \triangleright$ *Returns the j-invariant obtained by making $[D]$ act on $\mathcal{C}$ by the $\star_L$ action.*
8: **end function**

This algorithm is detailed in Algorithm 1 and its complexity is given in the following statement:

Proposition 1.3.9. *[87, Prop. 3.7] Algorithm 1 requires $O(d^2)$ operations in L and $O(d^2)$ applications of the Frobenius endomorphism.*

Inverting the group action

The goal of this section is to explain how to invert the group action $\star_L$. Indeed, using the same notation as in the previous section, for any $j_1, j_2 \in L$ representing L-isomorphism classes of rank-2 Drinfeld modules over L, there exists a unique $(u, v) \in \mathrm{Cl}(\mathbb{F}_q[\mathscr{H}])$ such that $(u, v) \star_L j_1 = j_2$.

The computation of this $(u, v) \in \mathrm{Cl}(\mathbb{F}_q[\mathscr{H}])$ starts by computing two Drinfeld modules ϕ_1, ϕ_2 which have respective j-invariants j_1, j_2, and whose Frobenius endomorphism have characteristic polynomial equal to ξ. This step can be done quite efficiently: there are explicit formulas for Drinfeld modules with given j-invariant. Then finding a $\overline{L}$-isomorphism to obtain the desired characteric polynomial can be done by computing the roots of a univariate polynomial ($\overline{L}$-isomorphisms of Drinfeld modules correspond to elements in $\overline{L}$).

Then we compute an isogeny ι of minimal τ-degree between ϕ_1 and ϕ_2 via linear algebra, as explained in Wesolowski's paper [121]. There is a correspondance between isogenies between ϕ_1 and ϕ_2 and ideals in $\mathbb{F}_q[\mathscr{H}]$, as described in the following lemma:

Lemma 1.3.10. *[87, Lemma 3.8] Let $\phi \in \mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi$ be an ordinary Drinfeld module. Then there is a one-to-one correspondence between monic isogenies with domain ϕ and nonzero ideals in $\mathbb{F}_q[\mathscr{H}]$. Moreover, let $\phi_1, \phi_2, \phi_3 \in \mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi$ be Drinfeld modules and $\iota_1 : \phi_1 \to \phi_2$, $\iota_2 : \phi_2 \to \phi_3$ be isogenies; the ideal associated to $\iota_2 \cdot \iota_1$ in $\mathbb{F}_q[\mathscr{H}]$ is the product of the ideals associated to ι_1 and ι_2.*

Our method to invert the group action is to compute the ideal corresponding to the isogeny ι. Since $\mathbb{F}_q[\mathscr{H}]$ is a Dedekind ring, we can proceed by computing a factored form of the ideal. There is a subtlety about what happens on the $\mathfrak{p}$-torsion, so we treat this part separately.

We start by considering the case where the norm of the isogeny is prime (and does not generate $\mathfrak{p}$), in Algorithm 2. This serves as a building block for Algorithm 3, which works for any norm. This algorithm returns a factored form of the algorithm corresponding to the isogeny. Mumford coordinates for the classes of its of the factors can then be computed, and finally combined using for instance Cantor's algorithm [27].

Algorithm 2 PrimeIsogenyToPrimeIdeal

function PrimeIsogenyToPrimeIdeal(An ordinary Drinfeld module $\phi \in \mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi$; A monic prime $r \in \mathbb{F}_q[X]$ such that $r \notin \mathfrak{p}$; An r-isogeny $\iota : \phi \to \psi$ between $\phi, \psi \in \mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi$.)
  $y \leftarrow$ remainder in the right-division of τ_L by ι
  $\iota^{(0)} \leftarrow 1$
  for $1 \leqslant n \leqslant \deg(r)$ **do**
    $\iota^{(n+1)} \leftarrow$ remainder in the right-division $\phi_T \cdot \iota^{(n)}$ by ι
  end for
  Using linear algebra, find $(v_0, \ldots, v_{\deg(r)-1}) \in \mathbb{F}_q^{\deg(r)}$ such that $y - (v_0\iota^{(0)} + \cdots + v_{\deg(r)-1}\iota^{(\deg(r)-1)}) = 0$
  Return $v_0 + v_1 X + \cdots + v_{\deg(r)-1}X^{\deg(r)-1}$.
  ▷ *The polynomial $v \in \mathbb{F}_q[X]$ returned is such that the left-ideal $\langle \phi_r, \tau_L - \phi_v \rangle \subset L\{\tau\}$ is generated by ι.*
end function

We state now the complexity of the main algorithm to invert the group action.

Proposition 1.3.11. *[87, Prop. 3.13] Let m denote the degree of u. Using the Cantor-Zassenhaus algorithm for polynomial factorization, Algorithm 3 is a probabilistic Las Vegas algorithm requiring $\widetilde{O}(dm^\omega + m^3 + m\log(q))$ expected operations in L and $O(dm + m^3)$ expected applications of the Frobenius endomorphism, where ω is a feasible exponent for the complexity of square matrix multiplication.*

1.3.4 An explicit computation

We implemented Algorithm 1 in C++/NTL in order to check its efficiency on objects of cryptographic size. We chose the following parameters: $q = 2, [L : \mathbb{F}_2] = 521$. With such parameters, we built Drinfeld modules at random until the characteristic polynomial of the Frobenius endomorphism defined a genus-260 hyperelliptic curve $\mathscr{H}$ over $\mathbb{F}_2$. The order of $\mathrm{Pic}^0(\mathscr{H}) \cong \mathrm{Cl}(\mathbb{F}_q[\mathscr{H}])$ can be computed efficiently by using the Denef-Kedlaya-Vercauteren algorithm [75, 36]. Using the implementation in the Magma Computer Algebra Software, we computed this order in 53 hours on a Intel(R) Xeon(R) CPU E7-4850.

Our experiments for the computation of the group action were conducted on an Intel i5-8365U@1.60GHz CPU, 8 cores, 16 GB RAM. We picked an irreducible degree-35 divisor on $\mathscr{H}$. By using the 8 cores of the laptop, computing the group action with the class of such an element on a j-invariant takes 24 ms and is therefore quite efficient. Our C++/NTL code is available at `https://gitlab.inria.fr/pspaenle/crs-drinfeld-521`.

Algorithm 3 IsogenyToIdeal

function IsogenyToIdeal(An ordinary Drinfeld module $\phi \in \mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi$; A (non-necessarily prime) monic polynomial $u \in \mathbb{F}_q[X]$, such that $u \notin \mathfrak{p}$; A u-isogeny $\iota : \phi \to \psi$ between ordinary Drinfeld modules in $\mathrm{Dr}_2(\mathbb{F}_q[X], L)_\xi$.)
  if $u = 1$ **then**
    Return $\mathbb{F}_q[X, Y]/(\xi)$
  end if
  $r \leftarrow$ a nonconstant monic prime factor of u
  $\widetilde{\iota} \leftarrow \mathrm{rgcd}(\iota, \phi_r)$
  if $\widetilde{\iota} = 1$ **then**
    Return IsogenyToIdeal$(\phi, u/r^{\mathrm{val}_r(u)}, \iota)$.
  else if $\widetilde{\iota} = \lambda\phi_r$ for some $\lambda \in L^\times$ **then**
    Return $\langle r(\overline{X})\rangle \cdot$ IsogenyToIdeal$(\phi, u/r, \iota \cdot \phi_r^{-1})$.
  else
    $v \leftarrow$ PrimeIsogenyToPrimeIdeal$(\phi, r, \widetilde{\iota})$
    $\widetilde{\phi} \leftarrow$ the codomain of $\widetilde{\iota}$, computed from ϕ and $\widetilde{\iota}$;
    Return $\langle u(\overline{X}), \overline{Y} - v(\overline{X})\rangle \cdot$ IsogenyToIdeal$(\widetilde{\phi}, u/r, \iota \cdot \widetilde{\iota}^{-1})$.
  end if
  ▷ *This function returns a factorization of the ideal $\mathfrak{a} \subset \mathbb{F}_q[X, Y]/(\xi)$ associated to ι in Lemma 1.3.10.*
end function

1.4 Computing Riemann-Roch spaces for nodal curves

This section is about results that have been obtained with Aude Le Gluher, and they are part of the Ph.D. thesis of Aude Le Gluher. More details can be found in our paper [85] published in the journal *Mathematics of Computation* or in the Ph.D. thesis of Aude Le Gluher [84].

1.4.1 Motivation

The celebrated Riemann-Roch theorem is one central tool for studying algebraic curves. This theorem provides an upper bound on the dimension of the space of global sections of a sheaf $\mathscr{O}(D)$ associated to a divisor D on a projective curve. In its modern form, this theorem can be understood as a duality result which relates spaces which are linked via Poincaré duality.

Although Riemann-Roch spaces are classically defined for nonsingular curves, its classical definition can be easily extended to singular curves, provided that the divisor avoids singularities:

Definition 1.4.1. *Let $\mathscr{C}$ be a projective curve over a perfect field k, and D be a Weil divisor whose support contains only nonsingular points. Then the* Riemann-Roch space $L(D)$ *is defined as the vector space*

$$L(D) \stackrel{def}{=} \{0\} \cup \{f : f \in k(\mathscr{C}) s.t. \operatorname{div}(f) \geqslant -D\}.$$

We let $\ell(D)$ denote the dimension of $L(D)$.

An important fact is that line bundles on curves are all isomorphic to line bundles from sheaves of the form $\mathscr{O}(D)$, see [65, Ch. IV, Sec. 1, Exercise 1.9.(c)] and the Riemann-Roch space $L(D)$ corresponds to the space of global sections of $\mathscr{O}(D)$.

If we write finite sums $D_+ = \sum_i a_i P_i$ and $D_- = \sum_i b_i Q_i$, where P_i, Q_i are closed nonsingular points, and a_i, b_i are positive integers, nonzero elements in the Riemann-Roch space $L(D_+ -$

D_-) are functions such that $\mathrm{val}_{P_i}(f) \geqslant -a_i$, meaning that poles are authorized up to some multiplicity, and $\mathrm{val}_{Q_i}(f) \geqslant b_i$, meaning that D_- imposes some zeros on the functions.

Theorem 1.4.2 (Riemann-Roch theorem)**.** *Let D be a divisor on a smooth projective algebraic curve of genus g with canonical divisor K. Then $\ell(D) - \ell(K - D) = \deg(D) - g + 1$.*

We emphasize that smoothness is required in Theorem 1.4.2. In fact, a rule of thumb is that Theorem 1.4.2 (and related statements) can be adapted to singular curves provided that we understand the structure of singularities sufficiently well. Understanding the singularities usually means that we can compute a desingularization of the curve. However, a problem about desingularization is that it often increases the dimension of the ambient space, which complicates the data structures required to represent the geometric objects.

For computational purposes we like working with plane curves as they can be conveniently manipulated via bivariate polynomials: curves are hypersurfaces in the plane, so they can be encoded by *principal ideals*. Unfortunately, not every algebraic curve is birationally equivalent to a smooth plane curve. However, if we allow *nodes*, which are the simplest types of singularities, then every algebraic curve is birationally equivalent to a nodal plane projective curve (under some restriction on the base field, which should be sufficiently large).

Let $\mathscr{C}$ be an affine plane $\mathscr{C}$ given via a polynomial equation $Q(X, Y) = 0$, with $Q \in k[X, Y]$, and let $(\alpha, \beta) \in \overline{k}^2$ be such that $Q(\alpha, \beta) = 0$. Then write $Q(X + \alpha, Y + \beta) = \sum_{i \geqslant 1} Q_i(X, Y)$, where Q_i is a homogeneous polynomial of degree i. The multiplicity of the point (α, β) is the smallest i such that $Q_i \neq 0$. A singularity (α, β) is a node if its multiplicity is 2 and if Q_2 is not a square in $\overline{k}[X, Y]$.

Proposition 1.4.3. *[8, Appendix A] Every complex algebraic curve is birationally equivalent to a projective nodal plane curve.*

Theorem 1.4.3 is not true in general for finite fields. We shall assume without proof that Theorem 1.4.3 holds when $\mathbb{C}$ is replaced by a finite field of sufficiently large characteristic.

Next, we introduce the Riemann-Roch problem, which will focus on in this section:

Problem 1.4.4. *Given a geometrically irreducible complete curve $\mathscr{C}$ defined over a perfect field k and a divisor D on $\mathscr{C}$ with support outside the singular locus of the curve, compute a basis for the vector space $L(D) \subset k(\mathscr{C})$.*

This problem has strong applications in coding theory and in arithmetic geometry. The first modern occurence of an algorithm to compute Riemann-Roch spaces appear in a paper by Goppa for the construction of algebraico-geometrical linear codes (abbreviated AG-codes) [61]. Such codes are generalization of Reed-Solomon codes. The main idea for the construction of AG-codes is to consider a $\mathbb{F}_q$-linear coding map

$$L(D) \to \mathbb{F}_q^m$$

where the map is an evaluation map at m rational points which are distinct from the support of D. Reed-Solomon codes are special cases of this setting. The main interest of this construction compared to Reed-Solomon codes is that m is not bounded above by q for curves of large genus. Moreover, it can be shown that this linear map has good decoding properties.

The second application domain of the computation of Riemann-Roch spaces lies in arithmetic geometry and computer algebra, as this is a basic tool to compute the group law in the Picard group (or the Jacobian variety) of a curve $\mathscr{C}$. Indeed, once a rational point ∞ has been fixed

on $\mathscr{C}$, we can represent a class $\alpha \in \mathrm{Pic}^0(\mathscr{C})$ as an effective divisor D of degree g such that $[D - g\infty] = \alpha$. If $\alpha_1, \alpha_2 \in \mathrm{Pic}^0(\mathscr{C})$ are two classes represented via two effective divisors D_1, D_2 of degree g, finding a representative for the class $\alpha_1 + \alpha_2$ amounts to finding a divisor D_3 of degree g such that $[D_1+D_2-2g\infty] = [D_3-g\infty]$. Said otherwise, $[D_3] = [D_1+D_2-g\infty]$ and such a divisor can be computed by finding an element in the Riemann-Roch space $L(D_1 + D_2 - g\infty)$.

During the two decades 1990-2010, a lot of progress was made in the algorithmic framework for computing Riemann-Roch spaces and for computing the group law on the Jacobian of curves, see [69, 64, 67, 119, 77].

1.4.2 Revisiting the Brill-Noether's method

In this work, we focused on the case of nodal curves: this allows us to work with a large family of curves, while keeping the inconvenience caused by singularities to a minimum. In particular, nodes are a special case of *ordinary* singularities, and Brill-Noether's adjoint theory provides us with tools to handle them.

For theoretical purposes, it is convenient to consider a smooth model $\widetilde{\mathscr{C}}$ of $\mathscr{C}$ together with a degree-1 morphism $\widetilde{\mathscr{C}} \to \mathscr{C}$ which is 1-to-1 on nonsingular points on $\mathscr{C}$ and 2-to-1 on singular points. Such a model always exists; we will need it mainly for proofs, and we do not need to compute it explicitly.

This way we can talk about divisors on $\widetilde{\mathscr{C}}$, and we say that a divisor on $\widetilde{\mathscr{C}}$ is *smooth* if it involves only points which do not project to nodes on $\mathscr{C}$. By slight abuse of notation, we say that D is a smooth divisor on $\mathscr{C}$ if it corresponds to a smooth divisor on $\widetilde{\mathscr{C}}$.

We define the *nodal divisor* E on $\widetilde{\mathscr{C}}$ which is the formal sum of the points which project to nodes. Therefore, E is an effective divisor of degree $2r$.

Representation of the curve. Our curve in $\mathbb{P}^2$ will be described by an absolutely irreducible polynomial q in $k[X, Y]$ encoding the intersection of $\mathscr{C}$ with the affine chart $\{(x : y : z) \mid z \neq 0\} \subset \mathbb{P}^2$. For simplicity, we assume that the curve is in *projective Noether position* with respect to Y, i.e. $\deg_Y(q) = \deg(q)$. This can achieved via a generic linear change of variables.

Representation of smooth divisors. Effective smooth divisors correspond to finite algebraic sets included in $\mathscr{C}$. In order to avoid having problems at infinity, we assume that all the points lie in the affine plane. This is a mild restriction as we can always compose with an automorphism of the projective plane $\mathbb{P}^2$ which brings all the points in a given affine chart.

Our representation of effective divisors is inspired by classical representations of 0-dimensional algebraic sets which use a separating function and a parametrization of the algebraic set by the roots of a univariate polynomial, see e.g. [60].

However, in our situation we must also encode multiplicities. A convenient way of doing so is by using a technique which is related to Mumford coordinates for hyperelliptic curves: we use a local expansion of the curve around the points with precision given by the multiplicity of the points. In order to achieve this, we need that the divisors are smooth.

Summing all these ingredients together, we obtained the following data structure:

Definition 1.4.5 (Primitive Element Divisor Representation (PEDR))**.** *A* Primitive Element Divisor Representation *(PEDR) on an affine curve $\mathscr{C}^0$ defined by a polynomial $q \in k[X, Y]$ is the data of*

- *an element $\lambda \in k$;*

- *three univariate polynomials* $\chi, u, v \in k[S]$*;*

such that

(Div-a) $\chi(S)$ *divides* $q(u(S), v(S))$*;*

(Div-b) $\lambda\, u(S) + v(S) = S$*;*

(Div-c) $\gcd(\frac{\partial q}{\partial X}(u(S), v(S)) - \lambda \frac{\partial q}{\partial Y}(u(S), v(S)), \chi(S)) = 1$*;*

(Div-d) χ *is monic;*

(Div-e) $\deg(u) < \deg(\chi)$ *and* $\deg(v) < \deg(\chi)$*.*

A PEDR defines an effective smooth divisor in the following way: given λ, χ, u, v, let $p_1, \ldots, p_\ell$ denote the irreducible divisors of χ. For i in $[\![1, \ell]\!]$, we consider the ideal $\widetilde{\mathfrak{p}_i} \stackrel{\text{def}}{=} \langle p_i(S), X - u(S), Y - v(S)\rangle \subset k[\mathscr{C}^0] \otimes_k k[S]$ and the closed points $\mathfrak{p}_i \stackrel{\text{def}}{=} \widetilde{\mathfrak{p}_i} \cap k[\mathscr{C}^0]$. Note that $\mathfrak{p}_i$ is a prime ideal in $k[\mathscr{C}^0]$ for all i. For the multiplicity, we notice that **(Div-c)** implies that the local ring and its completion are in fact discrete valuation rings. Let $\overline{x}$, $\overline{y}$ be the classes of X, Y in the residue field $\kappa = k[\mathscr{C}^0]/\mathfrak{p}_i$. By Cohen's structure theorem [43, Thm. 7.7], there is an injective morphism from κ to the completed local ring $\widehat{k[\mathscr{C}^0]_{\mathfrak{p}_i}}$, so $\overline{x}$, $\overline{y}$ can be regarded as elements of $\widehat{k[\mathscr{C}^0]_{\mathfrak{p}_i}}$. Then $\lambda(X - \overline{x}) + (Y - \overline{y})$ is a uniformizing element. The multiplicity μ_i associated to $\mathfrak{p}_i$ is the valuation of the function $\chi(\lambda X + Y)$ in this ring. This valuation corresponds to the multiplicity of p_i in the factorization of χ.

Any divisor defined by a PEDR is smooth because of condition **(Div-c)**. Also, it is defined over k by construction. The representation by a PEDR of a smooth divisor is not unique since there is a choice of the primitive element, which is encoded by λ. A natural question is whether any effective smooth divisor D on $\mathscr{C}^0$ can be represented by a PEDR: the answer is yes if k is sufficiently large, namely if $|k| > \binom{\deg(D)+1}{2}$, see [85, Prop. 3.2].

Representation of the nodal divisor. To represent the nodal divisor, we regard it as an algebraic set, without any considerations about multiplicities. Therefore we use a standard *primitive element representation*, which is slightly different from the PEDR above: it is encoded via λ, χ, u, v as in the PEDR, however Condition **(Div-c)** is dropped and **(Div-d)** is replaced by slightly stronger condition to avoid any multiplicity.

Definition 1.4.6 (Primitive Element Representation (PER))**.** *A* Primitive Element Representation *(PER) is the data of*

- *an element* $\lambda \in k$*;*
- *three univariate polynomials* $\chi, u, v \in k[S]$*;*

such that

(Div-a) $\chi(S)$ *divides* $q(u(S), v(S))$*;*

(Div-b) $\lambda\, u(S) + v(S) = S$*;*

(Div-d') χ *is monic and squarefree;*

(Div-e) $\deg(u) < \deg(\chi)$ *and* $\deg(v) < \deg(\chi)$*.*

A PER defines a 0-dimensional algebraic set in $\mathbb{A}^2(k)$. As above, there is a condition between the cardinality of k and that of the algebraic set to be able to represent it, see [85, Prop. 3.3].

Assumptions on the input divisor. Our algorithm requires a few assumptions on the input. First, we need that the support of the input divisor involves only nonsingular points of the curve that lie in the affine chart $Z \neq 0$. The second condition is more technical: it involves that existence of a low-degree function related to the input divisor which does not have zeros at singular points. We emphasize that this assuption can be removed without harming the asymptotic complexity by increasing the degree of the polynomial used as a common denominator for the Riemann-Roch space; this is proved in [3, Sec. 4.3]. However, removing this assumption has a significant impact on the practical timings, since this increases the size of all objects manipulated during the computations by a constant factor.

1.4.3 Algorithms and complexity

Algorithm 4 A bird's eye view of the algorithm.

1: **function** RiemannRochBasis(A curve $\mathscr{C}$ together with its nodal divisor E, and a divisor $D = D_+ - D_-$ on $\mathscr{C}$ such that D_+ and D_- are smooth effective divisors.)
2: ▷ *This function returns a basis of the Riemann-Roch space $L(D)$.*
3: $h \leftarrow$ Interpolate($\deg(\mathscr{C})$, D_+, E)
4: $D_h \leftarrow$ CompPrincDiv($\mathscr{C}$, h, E)
5: $D_{\mathsf{res}} \leftarrow$ SubtractDivisors(D_h, D_+)
6: $D_{\mathsf{num}} \leftarrow$ AddDivisors(D_-, D_{res})
7: $B \leftarrow$ NumeratorBasis($\deg(\mathscr{C})$, D_{num}, $\deg(h)$, E)
8: Return $\{f/h \mid f \in B\}$
9: **end function**

Algorithm 4 gives a general view of the Brill-Noether's framework for computing Riemann-Roch spaces, decomposing the main algorithm into subroutines. The subroutine Interpolate takes as input an effective divisor D_+, and it returns a form h such that $(h) \geqslant D_+ + E$. Geometrically, it computes a nonzero element in $\Gamma(\mathscr{O}_{\mathscr{C}}(-D_+ - E) \otimes \mathscr{O}_{\mathscr{C}}(d))$, in other words a degree-d form on the curve with zeros at the points given in D_+ and at the nodes, for some well-chosen d. This computation is done via linear algebra. Then CompPrincDiv computes from h a convenient representation of the divisor $\mathrm{div}(h) - E$. It essentially boils down to a resultant computation, although some care is required to handle multiplicities. The routines AddDivisors and SubtractDivisors use gcd and lcm computations of univariate polynomials to compute the arithmetic operations on divisors. Again, some care is required to handle multiplicities; in particular, this step might require Hensel lifting. Then, NumeratorBasis takes as input the effective divisor D_{num} and the degree of h, and it returns a basis of the vector space of all forms $f \in k[\mathscr{C}]$ of degree $\deg(h)$ such that $(f) \geqslant D_{\mathsf{num}} + E$. Geometrically, we compute a basis of $\Gamma(\mathscr{O}(-D_{\mathsf{num}} - E) \otimes \mathscr{O}(\deg(h)))$. Finally, we divide this basis by the common denominator h in order to obtain a basis of the Riemann-Roch space.

One of the cornerstones of the correctness of Algorithm 4 is the Brill-Noether's residue theorem. This theorem is one of the foundations of the theory of adjoint curves, and it gives a sufficient condition for a form to be a suitable denominator for all functions in the Riemann-Roch space. In the case of nodal plane curves, an adjoint curve is a curve which goes through all the nodes of $\mathscr{C}$, and E is the adjoint divisor as defined in [54, Sec. 8.1].

Brill-Noether theory is in fact more vast than just nodal curves, and it covers ordinary singularities in general. We shall see later that it can be further generalized via Gorenstein's theory of adjoints.

We won't go into the details of the proof of correctness of the algorithm (see [85] for details), but we mention that it relies on Brill-Noether theorem:

Proposition 1.4.7. *(Brill-Noether theorem for nodal curves)[54, Sec. 8.1] Let D, D' be two linearly equivalent effective divisors on $\mathscr{C}$. Let $h \in k[\mathscr{C}]$ be a form such that $(h) = D + E + A$ for some effective divisor A. Then there exists a form $h' \in k[\mathscr{C}]$ of the same degree as h such that $(h') = D' + E + A$.*

The Brill-Noether is actually a very powerful tool, which roughly speaking explains how the local analysis at singularity gives global information about a curve. The formalism of sheaves is a convenient language to express this relationship between local and global properties.

In what follows, if D is a divisor on $\mathscr{C}$ (more precisely, on its smooth model $\widetilde{\mathscr{C}}$), we let $\mathscr{O}(D)$ denote the sheaf of functions whose valuation at a point P is not less than the multiplicity of P in $-D$. This sheaf $\mathscr{O}(D)$ is a $\mathscr{O}_{\mathscr{C}}$-module, and it is actually a line bundle, whose inverse for the tensor product is $\mathscr{O}(-D)$. Its global sections $\Gamma(\mathscr{O}(D))$ equals the Riemann-Roch space $L(D)$. For $d \in \mathbb{Z}_{\geqslant 0}$, we shall also need the twisting sheaf $\mathscr{O}(d)$ whose global sections are degree-d forms on $\mathscr{C}$, namely $\Gamma(\mathscr{O}(d)) = k[\mathscr{C}]_d$.

The following corollary explains that adjoints measure how far from invertible divisors built on singularities are. In particular, adjoints allow us to reduce the Riemann-Roch problem to the problem of computing a basis of $\Gamma(\mathscr{O}(-D') \otimes \mathscr{O}(d))$ for some effective divisor D' and some positive integer d, as described by the next corollary, which is a direct consequence of Brill-Noether's theorem:

Corollary 1.4.8. *Let d be a positive integer and $D = D_+ - D_-$ be a smooth divisor on $\mathscr{C}$. For any nonzero $h \in \Gamma(\mathscr{O}(-D_+ - E) \otimes \mathscr{O}(d))$,*

$$\Gamma(\mathscr{O}(D)) = \frac{1}{h}\Gamma(\mathscr{O}(D - \operatorname{div}(h)) \otimes \mathscr{O}(d)).$$

Proof. It is a formalization of [85, Thm. 2.2] and of its proof in the language of sheaves. □

Note that $\Gamma(\mathscr{O}(D))$ is not equal to $\Gamma(\mathscr{O}(D - \operatorname{div}(h))) \otimes \frac{\Gamma(\mathscr{O}(d))}{h}$. Theorem 1.4.8 actually describes a general strategy to compute Riemann-Roch spaces: we start by finding a common denominator h of all elements in the Riemann-Roch spaces, then the problem can be reduced to finding a basis of a space of sections associated to an effective divisor. This is the core of the Brill-Noether's approach. Algorithm 4 implements this method. The following result gives the asymptotic complexity of this algorithm:

Theorem 1.4.9. *[85, Thm. 6.8] Algorithm 4 (RiemannRochBasis) requires at most*

$$O(\max(\deg(\mathscr{C})^{2\omega}, \deg(D_+)^{\omega}))$$

arithmetic operations in k.

We will not go into the details of all the subroutines; the main computational tool (which is also the bottleneck of the complexity) is the following interpolation problem:

Problem 1.4.10. *Given a nodal curve $\mathscr{C}$, a positive integer d and an effective divisor D whose support involves only nonsingular points, compute a basis of the space of forms f of degree d such that $\operatorname{div}(f) \geqslant D + E$, i.e. compute a basis of $\Gamma(\mathscr{O}(-D - E) \otimes \mathscr{O}(d))$.*

In this work, we solve this problem by using linear algebra. More precisely, we use the fact that $\Gamma(\mathscr{O}(-D) \otimes \mathscr{O}(d))$ is a sub k-vector space of $\Gamma(\mathscr{O}(d))$ when D is effective. When D has its support in the affine plane $\mathbb{A}^2$, it can be represented via an ideal I_D in the affine coordinate ring $k[\mathscr{C}]$. The space of global sections $\Gamma(\mathscr{O}(-D) \otimes \mathscr{O}(d))$ is precisely the kernel of the evaluation map $\Gamma(\mathscr{O}(d)) \to k[\mathscr{C}]/I_D$, and it can be computed via linear algebra over k.

In follow-up works by Abelard, Berardini, Couvreur and Lecerf, they manage to use an extra structure to decrease the complexity. More precisely, they consider a map $\pi : \mathscr{C} \to \mathbb{P}^1$, which implies that $\mathscr{O}(-D) \otimes \mathscr{O}(d)$ has a structure of a $\pi^*(\mathscr{O}_{\mathbb{P}^1})$-locally free module of rank $\deg(\mathscr{C})$. Under some assumption of the projection map, this makes the space of affine sections of $\mathscr{O}(-D)$ a free $k[X]$-module of finite rank and its elements of degree at most d can be extended to a basis of $\Gamma(\mathscr{O}(-D) \otimes \mathscr{O}(d))$. They use algorithms for computing shifted Popov forms to compute such a basis of the $k[X]$-module of low-degree elements, which provide them with subquadratic complexity.

Assumptions and how to remove them. Our work require some assumptions on the input. The reason why we need those assumptions is that our goal is to provide practical implementation. In particular, we would like that the common denominator h that we compute for our Riemann-Roch space has the smallest possible degree.

Equivalently, we try to choose the smallest possible degre d in Theorem 1.4.8 such that the space of of global sections in which we pick h is nonzero. Direct computations gives us an explicit formula for this degree. However, this strategy may fail in the following exceptional case. By construction, for every h in $\Gamma(\mathscr{O}(-D_+ - E) \otimes \mathscr{O}(d))$, $\operatorname{div}(h) = D_+ + E + A$ for some effective divisor A. It may happen that for all h in $\Gamma(\mathscr{O}(-D_+ - E) \otimes \mathscr{O}(d))$, the support of the corresponding effective divisor A involves singular points. If this situation happens, then our algorithm fails.

In fact, the authors of [3] show that this assumption can be removed without harming the asymptotic complexity, by choosing a larger value for the degree of h. Even though increasing d does not harm the asymptotic complexity, this increases significantly the time of practical computations.

A possible strategy then would be to make the algorithm iterative by detecting if this assumption is not satisfied, and increasing the degree in this case. We did not implement this strategy, but this might be a future development of our `rrspace` software.

In practice, the only examples that I know where the assumptions are not satisfied are those built especially to make this assumption fail.

1.4.4 Implementation and experimental results

We have implemented Algorithm 4 in C++/NTL for $k = \mathbb{Z}/p\mathbb{Z}$. As mentioned in Section 1.4.1, the group law on the Jacobian of a curve can be computed via the discovery of a nonzero function in a Riemann-Roch space. We designed a specific implementation for this task, which can be slightly optimized (for instance, we only need one function in the Riemann-Roch, not the full basis). Our software `rrspace` is available at `https://gitlab.inria.fr/pspaenle/rrspace` and it is distributed under the LGPL-2.1+ license.

Experiments have been run on a Intel(R) Core(TM) i5-6500 CPU@3.20GHz with 16GB RAM.

Figures 1.1 and 1.2 display some experimental results that have been obtained with our implementation and which are compared with the state-of-art implementation of the computation of Riemann-Roch spaces in the Magma computer algebra software.

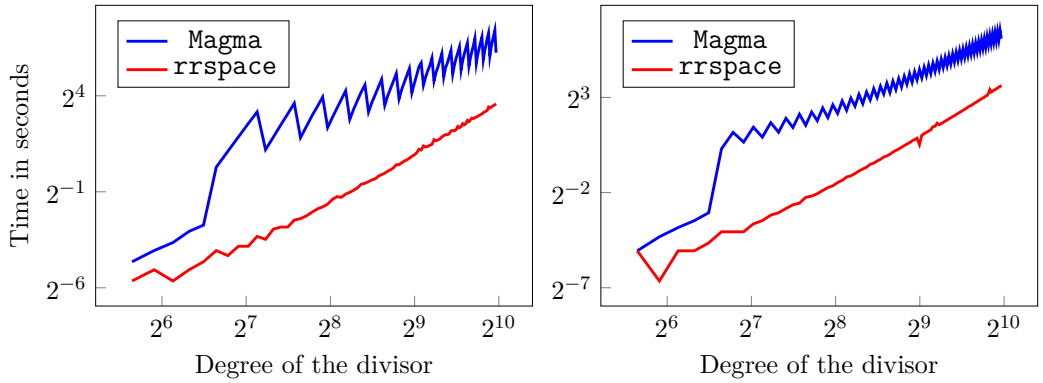


Figure 1.1: Comparison of the time required by `rrspace` and `Magma V2.23-8` to compute a basis of $L(D)$ on a fixed smooth curve of degree 10 over $\mathbb{Z}/65521\mathbb{Z}$. On the left, D is the sum of random irreducible effective divisors of degree 10. On the right, D is a multiple of an irreductible divisor of degree 10. Both axes are in logarithmic scale.

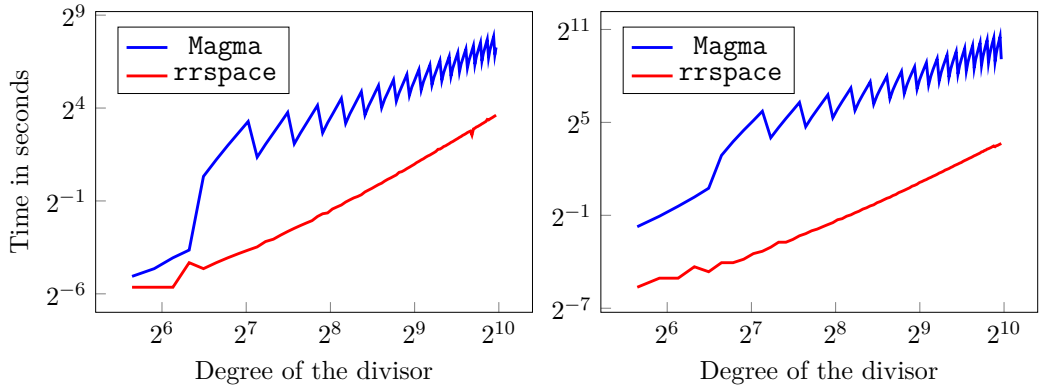


Figure 1.2: Comparison of the time required by `rrspace` and `Magma V2.23-8` to compute a basis of $L(D)$ on a fixed curve of degree 10, where D is the sum of random irreducible effective divisors of degree 10. On the left, the base field is $\mathbb{Z}/65521\mathbb{Z}$ and the curve is nodal. On the right, the base field is $\mathbb{Z}/(2^{32}-5)\mathbb{Z}$ and the curve is smooth. Both axes are in logarithmic scale.

In our experiments on nodal curves, our implementation was faster than Magma's by a factor between 6 and 200. We refer to [85, Sec. 8] for more details on our experiments.

1.4.5 Follow-up works

Our work on Riemann-Roch spaces was followed by a series of papers by other authors. These papers generalized the families of curves that can be handled by the Brill-Noether's approach, or they improved the asymptotic complexity.

- *Sub-quadratic time for Riemann-Roch spaces: case of smooth divisors over nodal plane projective curves* by Abelard, Couvreur and Lecerf [3]. This paper improves the complexity by providing an algorithm that is less than quadratic ($(\omega+1)/2$ to be precise) in the size of the input in the case of curves defined over a fixed finite field $\mathbb{F}_q$. The cornerstone of this improvement is to use specialized algorithms for free finitely-generated $\mathbb{F}_q[X]$-modules (shifted Popov forms);

- *Efficient computation of Riemann-Roch spaces for plane curves with ordinary singularities* by Abelard, Couvreur and Lecerf [4]. This paper deals with a more general class of curves, with the same subquadratic complexity as in the nodal case;

- *Computing Riemann-Roch spaces via Puiseux expansions* by Abelard, Berardini, Couvreur and Lecerf [2]. The aim of this paper is to allow more complicated singularities. This algorithm has exponent complexity ω in terms of the input. They use rational Puiseux series to compute general adjoint curves.

- *A proof of the Brill-Noether method from scratch* by Berardini, Couvreur and Lecerf [19]. The aim of this paper is to simplify proofs, and to provide a unified framework for describing the recent progress based on the Brill-Noether's method, using only simple mathematical tools.

Chapter 2

Polynomial systems

2.1 Panorama

The study of polynomial systems is one of the foundations of modern applied computer algebra. Perhaps one of the reasons for their ubiquity in applications is their expressiveness: many *inverse* problems in science and engineering can be modeled as polynomial systems. This is the reason why developing general-purpose algorithmic tools is so important. The success of Gröbner basis algorithms is probably related to the fact that they do not need assumptions on the input to return results which provide useful information about the solutions of polynomial systems. However, this flexibility and robustness does not come for free: the efficiency of general-purpose algorithms is not easy to analyze, and worst-case complexity bounds are far too pessimistic for most use cases. Therefore, it is important to investigate specific structures that appear in polynomial systems arising in applications, for which we can try to design specific algorithms and provide usable complexity analyses.

This is the point of view of this chapter, where we will focus on two types of structures. The first structure comes from the monomial support of the polynomial systems. Our main contribution here is to study if some compactifications of the ambient space may be more convenient than the usual projective compactification, which regards affine varieties as dense subsets of projective spaces. The main goal is to find new ways to organise the computations during Gröbner bases algorithms, to increase the efficiency when there are monomial structures. The second structure that we study in this chapter is obtained when we consider polynomial systems encoding *critical points* of maps. This arises typically in applications involving optimization problems, which are quite common.

The main tools to investigate these structures come from algebraic geometry, since indicators of the complexity are often related to numerical invariants which can be computed with tools from intersection theory (dimensions, degrees, etc.).

As Gröbner bases algorithms are versatile, general software implementations are important for applications. In fact, there are already many implementations of Gröbner bases algorithms that have been proposed. We can cite for instance `FGb` by J.-C. Faugère, the implementation within the `Magma` computer algebra software by A. Steel, the recent software `msolve` by J. Berthomieu, C. Eder, V. Neiger, M. Safey El Din, or the `OpenF4` software by V. Vitse. This list is not exhaustive. I have written my own C++/NTL implementation of the $F4$ algorithm in the `tinyGB` software. The goal of this implementation is not to design a general software, but rather to focus on improvements for special families of structured systems. In particular, during the last years, I mainly focused on systems with *monomial structures*, and the endgoal of my

research on this topic is to design fast and efficient software for such polynomial systems.

2.2 Sparse polynomial systems

2.2.1 Preliminaries

Designing efficient algorithms to solve sparse polynomial systems has been a long standing computational problem. Perhaps one of the stepping stone on which this field of research arose was the discovery of the Bernstein-Khovanskii-Kushnirenko theorem during the 70s. This result showed that the number of *toric* solutions — i.e. affine solutions with no zero coordinate — of sparse polynomial systems can be bounded via a value which is much sharper than the classical Bézout bound: It is bounded by the mixed volume of the Newton polytopes of the input polynomials.

The Newton polytope of a Laurent polynomial $f \in \mathbb{C}[X_1^{\pm 1}, \ldots, X_n^{\pm 1}]$ is constructed by considering the set $A \subset \mathbb{Z}^n$ of all exponent vectors of monomials which occur in f with nonzero coefficient: The Newton polytope of f is the convex hull of A in $\mathbb{R}^n$. The mixed volume of n convex and bounded bodies $\mathcal{P}_1, \ldots, \mathcal{P}_n \subset \mathbb{R}^n$ is the real number which is obtained by considering the following map:

$$\begin{array}{ccc} \mathbb{R}^n_{\geqslant 0} & \to & \mathbb{R}_{\geqslant 0} \\ (t_1, \ldots, t_n) & \mapsto & n! \cdot \mathrm{Vol}(t_1 \cdot \mathcal{P}_1 + \cdots + t_n \cdot \mathcal{P}_n), \end{array}$$

where Vol denotes the classical Euclidean volume.

It can be proved that this map is in fact a polynomial map, which is the evaluation of a homogeneous polynomial of degree n in $\mathbb{C}[t_1, \ldots, t_n]$. The coefficient of $t_1 t_2 \cdots t_n$ in this polynomial is the *mixed volume*. Moreover, if the input polytopes are lattice polytopes (i.e. convex hulls of points in $\mathbb{Z}^n$), then the mixed volume is an integer.

Theorem 2.2.1 (BKK theorem)**.** *[20, Thm. A] For generic choice of coefficients in $\mathbb{C}$, the number of solutions in the torus $(\mathbb{C} \setminus \{0\})^n$ of a system of n polynomial equations with fixed monomial supports equals the mixed volume of the Newton polytopes of the polynomials.*

This theorem raises a lot of questions. In particular, it suggests that the classical projective compactification of the affine space might not be particularly relevant in the case of sparse systems, as intersection theory in the projective space cannot take into account the Newton polytopes of the equations. Also, the mixed volume provides a measure of the "algebraic complexity" of the geometric objets. Therefore, a computational goal could be to design algorithms whose complexity is related to this measure.

Before moving on to the general case, we can first focus on multi-homogeneous and weighted homogeneous polynomial systems, which are special cases featuring typical difficulties.

Multi-homogeneity. There are many ways to look at multi-homogeneity. For geometers, multi-homogeneous objects are varieties in products of projective spaces. For algebraists, multi-homogeneity arises from multi-graded rings. In applications, multi-homogeneity arises often when there are several blocks of unknowns which represent coordinates of distinct natures. We will adopt here the point of view of polynomial system solving. We consider a family of ℓ polynomial rings $k[X_0^{(1)}, \ldots, X_{n_1}^{(1)}], \ldots, k[X_0^{(\ell)}, \ldots, X_{n_\ell}^{(\ell)}]$. The tensor product of these polynomial rings is isomorphic to a polynomial ring $k[X_0^{(1)}, \ldots, X_{n_1}^{(1)}, \ldots, X_0^{(\ell)}, \ldots, X_{n_\ell}^{(\ell)}]$. Since each factor

of the tensor product has a $\mathbb{Z}_{\geqslant 0}$-grading, the tensor product inherits a $\mathbb{Z}_{\geqslant 0}^{\ell}$-grading, where the $(d_1, \ldots, d_\ell)$-slice is

$$k[X_0^{(1)}, \ldots, X_{n_1}^{(1)}]_{d_1} \otimes \cdots \otimes k[X_0^{(\ell)}, \ldots, X_{n_\ell}^{(\ell)}]_{d_\ell}.$$

This $\mathbb{Z}_{\geqslant 0}^{\ell}$-grading is a refinement of the usual $\mathbb{Z}_{\geqslant 0}$-grading. The connection with multi-projective geometry comes from the fact that if $(x_0^{(1)}, \ldots, x_{n_1}^{(1)}, \ldots, x_0^{(\ell)}, \ldots, x_{n_\ell}^{(\ell)}) \in \overline{k}^{n_1+\cdots+n_\ell+\ell}$ is a zero of multi-homogeneous polynomials, then for any $(\lambda^{(1)}, \ldots, \lambda^{(\ell)}) \in (\overline{k} \setminus \{0\})^\ell$, the polynomials also vanish at $(\lambda^{(1)} x_0^{(1)}, \ldots, \lambda^{(1)} x_{n_1}^{(1)}, \ldots, \lambda^{(\ell)} x_0^{(\ell)}, \ldots, \lambda^{(\ell)} x_{n_\ell}^{(\ell)})$, therefore solutions of multihomogeneous polynomials correspond to points in the multi-projective space $\mathbb{P}^{n_1} \times \cdots \times \mathbb{P}^{n_\ell}$. A typical difficulty is that multi-homogeneous systems vanish on some high-dimensional coordinate subspaces: when all the variables in a block are set to zero, then all multihomogeneous polynomials with degree at least 1 with respect to this block of variables vanish. In practice, this implies that there are some unwanted high-dimensional parasitic solutions of polynomials systems which create practical algorithmic issues.

Weighted homogeneity. Weighted homogeneous systems arise when the homogenity relation involves some weights. A polynomial $f \in k[X_0, \ldots, X_n]$ is homogeneous with respect to the weight vector $(w_0, \ldots, w_n)$ if $f(X_0^{w_0}, \ldots, X_n^{w_n})$ is homogeneous in the usual sense. The main difficulty for weighted homogeneous systems comes from the fact that the natural ambient space is the weighted projective space, which may have singularities. This implies that some degrees correspond to non-Cartier divisor classes, which is the cause of some subtleties for predicting the behavior of solving algorithms. Another difficulty is that the polynomial algebra is not generated in degree 1, which is also the source of some inconveniences: for instance, generic dense systems of some given degrees might not lead to regular sequences.

Methods for solving multi-homogeneous and weighted homogeneous systems have been studied for a long time. Resultant-based approaches have been designed to build matrices from which some multi-homogenous systems can be solved via linear algebra [37]. Numerical homotopy algorithms can also take profit from the multi-homogeneous structure, by using polyhedral methods [70]. Polyhedral methods were also useful to design symbolic algorithms [71, 44].

Gröbner bases are among the main tools to manipulate symbolically polynomial systems. They were introduced by Buchberger in his PhD thesis [25]. Algorithms for computing Gröbner bases have been improved over the years, and the $F4/F5$ algorithms [49, 50] are now the standard framework for reducing the problem of computing Gröbner bases to linear algebra and efficient row echelon form computations. Gröbner bases algorithms usually proceed degree-by-degree: doing so, they implicitly compute things with respect to completion in a projective space.

The main challenge in order to adapt them to the multi-homogeneous and weighted settings is to understand how the different gradings interact with the structure of the F4 algorithm.

In the multi-homogeneous case, I studied some variants of the F4/F5 frameworks during my PhD thesis [110]. An analysis of the complexity of a variant of Gröbner basis algorithms for zero-dimensional systems was done in [18], see also Matías Bender's PhD thesis [17]. In the weighted setting, Thibaut Verron's PhD thesis investigates algorithmic and complexity aspects of weighted Gröbner bases algorithms, see [47, 46, 117].

The geometric ambient spaces associated to multi-homogeneity and weighted homogeneity are special cases of complete normal toric varieties constructed from polyhedral fans. In the next sections, we try to start answering the difficulties that we encounter in the general context of complete normal toric varieties built from fans.

A first approach is to consider *projective toric varieties*, which correspond to polyhedral fans which arise from normal fans of lattice polytopes. In this case, we start with such a polytope, which provides a closed immersion of the toric variety in a projective space and a $\mathbb{Z}$-gradation of a subgroup of the Picard group of the toric variety (generated by the class of a hyperplane section in the projective space). We show that this provides a framework in which we can try to profit from the monomial structures while using standard techniques from Gröbner bases algorithms. The fact that there is a $\mathbb{Z}$-grading generated in degree 1 enables the use of techniques based on the *Hilbert function and series* to obtain numerical invariants associated to the complexity of the computation.

Another difficulty which arises for instance for weighted projective spaces is that the ambient space might be singular. This is more difficult to handle. In fact, things can get very bad. For instance, there may be sequences of degrees where it is not clear whether generic systems are regular. This problem was asked in [46], since regularity is usually the first step towards an analysis of the complexity of the computations with Gröbner bases.

2.2.2 Algorithms for sparse systems

In this section, the field of definition of the polynomial systems will be the field of complex numbers. Although most results are probably true when $\mathbb{C}$ is replaced with another field of characteristic 0 or p sufficiently large, we will need to invoke several results from toric geometry, which are often stated for toric varieties over $\mathbb{C}$.

2.2.3 Homogeneous coordinate rings for projective toric varieties

This section presents results obtained with Jean-Charles Faugère and Jules Svartz and published in the proceedings of the ISSAC 2014 conference [52].

In this section, our goal is to exploit monomial structures of sparse systems while still using the classical framework for Gröbner bases computations. In order to achieve this, the main tool that we use is a special homogenization process that regards sparse polynomials as homogeneous elements in a $\mathbb{Z}_{\geqslant 0}$-graded ring that is built from the Newton polytopes of the input polynomials. This graded ring is a *polytopal algebra*, and it is a homogeneous coordinate ring for the complete toric variety associated to the normal fan of a polytope.

Before defining the homogeneous ring in which we will perform the computations, we start by defining *semigroup algebras*:

Definition 2.2.2. *An* affine semigroup S *in* $\mathbb{Z}^n$ *is a finitely-generated subsemigroup. The* semigroup algebra $\mathbb{C}[S]$ *is the* $\mathbb{C}$*-algebra of formal sums* $\sum_{s\in S} a_s X^s$ *with* $a_s \in \mathbb{C}$ *with a finite number of nonzero coefficients, with coefficient-wide addition, and multiplication given by*

$$\left(\sum_{s\in S} a_s^{(1)} X^s\right) \cdot \left(\sum_{s\in S} a_s^{(2)} X^s\right) = \sum_{s\in S} \left(\sum_{\substack{s_1,s_2\in S\\ s_1+s_2=s}} a_{s_1}^{(1)} \cdot a_{s_2}^{(2)}\right) X^s.$$

Semigroup algebras play an important role in toric geometry: the building blocks of toric varieties are *affine toric varieties*, which are precisely schemes of the form $\mathrm{Spec}(\mathbb{C}[S])$ for an affine semigroup S.

Definition 2.2.3. *Let* $\mathcal{P}$ *be a lattice polytope in* $\mathbb{R}^n$*, i.e. a convex hull of lattice points. The* polytopal algebra $\mathbb{C}[\mathcal{P}]$ *is the semigroup algebra associated to the semigroup*

$$(\mathbb{R}_{\geqslant 0} \cdot (\mathcal{P} \times \{1\})) \cap \mathbb{Z}^{n+1}$$

in the Euclidean lattice $\mathbb{Z}^{n+1}$.

Polytopal algebras are canonically $\mathbb{Z}_{\geqslant 0}$-graded by the last coordinate of the exponents of monomials: the degree of a monomial $X^{(s,d)}$ is d. The polytope $\mathcal{P}$ is called *normal* if $\mathbb{C}[\mathcal{P}]$ is a *homogeneous algebra*, i.e. it is generated by degree 1 elements, see [32, Lem. 2.2.14].

Polytopal algebras play an important role in toric geometry: they are coordinate rings for projective toric varieties, see [32, Thm. 7.1.13]. Once we have polytopal algebras, we have a way to homogenize polynomials with respect to this polytopal algebra: for a given Laurent polynomial $f \in \mathbb{C}[X_1^{\pm 1}, \ldots, X_n^{\pm 1}]$, we consider the smallest integer $i > 0$ such that the Newton polytope $\mathrm{Newton}(f)$ of f can be included in $i \times \mathcal{P}$ as a lattice polytope. This inclusion gives a way to consider f as a homogeneous element of degree i in $\mathbb{C}[\mathcal{P}]$. This strategy is particularly well-suited in the case where all the Newton polytopes of the input polynomials are actually some multiples of $\mathcal{P}$.

A nice feature of polytopal algebras is that they generalize classical polynomial algebras, and the notion of admissible monomial orderings extends to this setting without any major difficulty. Consequently, we can easily define a notion of *sparse Gröbner bases*, which extends in a natural way the classical notion of Gröbner bases in polynomial algebras, see [52, Def. 3.1]. Then we can use the classical F4 strategy for computing a Gröbner basis: we consider a homogeneous ideal $I \subset \mathbb{C}[\mathcal{P}]$ generated by homogenization of sparse polynomials, and we compute echelon bases of I_d for increasing d by computing the row echelon forms of Macaulay matrices. At some degree d_0, the echelon basis of I_{d_0} can be dehomogenized and it will provide a Gröbner basis of the ideal generated by the input polynomials.

Implementing this strategy can provide large speedups as it takes into account the monomial structures of the input polynomials, see e.g. [52, Table 1]. In the 0-dimensional case, we also provide a variant of the FGLM algorithm, in order to convert a Gröbner basis in the polytopal algebra into a more convenient Gröbner basis for computing the solutions.

The next theoretical step to measure and predict the complexity of such computations is to obtain bounds on d_0. This is often a difficult step. In order to obtain such numerical information, it is often useful to look at its Hilbert function and series:

Definition 2.2.4. *Let* $R = \oplus_{i \geqslant 0} R_i$ *be a finitely-generated* $\mathbb{Z}_{\geqslant 0}$*-graded* k*-algebra. The function*

$$\begin{array}{rccc} \mathrm{HF}_R : & \mathbb{Z}_{\geqslant 0} & \to & \mathbb{Z}_{\geqslant 0} \\ & d & \mapsto & \dim_k(R_d) \end{array}$$

is called the Hilbert function *of* R.

Proposition 2.2.5. *[42, Thm. 1.11][43, Thm. 1.11] If* R *is* homogeneous *(i.e.* $\mathbb{Z}_{\geqslant 0}$*-graded and generated by elements of degree* 1*), then*

- *the generating series* $\mathrm{HS}_R(t) = \sum_{i \geqslant 0} \mathrm{HF}_R(i)\, t^i$ *— called the* Hilbert series *of* R *— is rational and it has the form*
$$\frac{Q(t)}{(1-t)^n},$$
where n *is the Krull dimension of* R *and* Q *is a polynomial such that* $Q(1) \neq 0$.

- *there exists a polynomial* $\mathrm{HP}_R \in \mathbb{Q}[X]$ *— called the* Hilbert polynomial *of* R *— such that* $\mathrm{HF}_R(i) = \mathrm{HP}_R(i)$ *for all sufficiently large* i. *The degree of* HP_R *is the Krull dimension of* R *minus* 1 *(if the Krull dimension is* 0*, then* $\mathrm{HP}_R = 0$*).*

Remark 2.2.6. *The condition of being generated in degree* 1 *is necessary, as shown by the example* $k[X^2]$ *which is generated in degree* 2 *and whose Hilbert series is the expansion at* 0 *of* $\frac{1}{1-t^2}$.

Classical objects for investigating properties of lattice polytopes are their *Ehrhart polynomials and series*, see e.g. [91, Sec. 12]. These algebraic objects contain many numerical information about the combinatorial structure of lattice polytopes. In fact, these objects coincide with the Hilbert function and series of the associated polytopal algebra:

Definition 2.2.7. *Let* $\mathcal{P} \subset \mathbb{R}^n$ *be a lattice polytope. The Hilbert function, polynomial and series of the polytopal algebra* $\mathbb{C}[\mathcal{P}]$ *are called the* Ehrhart function, polynomial and series *of* $\mathcal{P}$.

A typical numerical indicator of the "complexity" of a lattice polytope is the Castelnuovo-Mumford regularity of the associated polytopal algebra, which can be read off from the numerator of the rational function whose expansion around 0 is the Hilbert/Ehrhart series. In particular, this analogy works particularly well for *normal lattice polytopes* [32, Def. 2.2.9]: such polytopes satisfy the property that the semigroup of lattice points in the cone over the polytope are generated by the lattice points in the input polytope. Equivalently, this means that the polytopal algebra is generated by its degree-1 elements.

Proposition 2.2.8. *[52, Prop. 2.7] Let* $\mathcal{P}$ *be a normal lattice polytope in* $\mathbb{R}^n$. *Then the degree of the numerator of its Ehrhart series (seen as the expansion of a rational function) is* $n - \ell$, *where* ℓ *is the largest integer such that the interior of* $\ell \cdot \mathcal{P}$ *does not contain any lattice point. This integer* $n - \ell$ *is the* Castelnuovo-Mumford regularity *of the polytopal algebra.*

Now that we understand the graded structure of polytopal algebras built on normal lattice polytope, we will study the graded structure of quotients of this ring by ideals generated by regular sequences.

Definition 2.2.9. *Let* $R = \oplus_{s \in S} R_s$ *be an algebra graded by an affine semigroup* S. *An element* $x \in R$ *is called* homogeneous of degree s *if* $x \in R_s$ *for some* $s \in S$. *A sequence* $x_1, \ldots, x_\ell$ *of homogeneous elements is called regular if* $\langle x_1, \ldots, x_\ell \rangle \neq R$ *and for all* $i \in [\![1, \ell - 1]\!]$, x_{i+1} *does not divide* 0 *in* $R/\langle x_1, \ldots, x_i \rangle$.

Proposition 2.2.10. *[43, Proof of Thm. 1.1] Let* R *be a (non-necessarily generated in degree* 1*) finitely-generated* $\mathbb{Z}_{\geqslant 0}$*-graded algebra. Let* $x_1, \ldots, x_\ell$ *be a regular sequence of homogeneous elements. Then*

$$\mathrm{HS}_{R/\langle x_1, \ldots, x_\ell \rangle}(t) = \mathrm{HS}_R(t) \cdot \prod_{i=1}^{\ell} (1 - t^{\deg(x_i)}).$$

The reciprocal statement only holds if R is homogeneous:

Proposition 2.2.11. *Let* R *be a finitely-generated homogeneous* $\mathbb{C}$*-algebra. Then* $x_1, \ldots, x_\ell \in R$ *is a regular sequence of homogeneous elements if and only if*

$$\mathrm{HS}_{R/\langle x_1, \ldots, x_\ell \rangle}(t) = \mathrm{HS}_R(t) \cdot \prod_{i=1}^{\ell} (1 - t^{\deg(x_i)}).$$

Proof. This is a consequence of [42, Thm. 1.11]. □

Hilbert series are very interesting because they give the information on the size and rank of the matrices that are built during the Gröbner basis computations, under genericity assumptions on the coefficients of the input system. This provides useful complexity information in order to predict the cost of computing Gröbner bases with such algorithms, in terms of numerical combinatorial data associated to the polytope $\mathcal{P}$.

In particular, a central numerical invariant for estimating the complexity is the *Castelnuovo-Mumford regularity* of $\mathbb{C}[\mathcal{P}]/I$, where I is a homogeneous ideal in $\mathbb{C}[\mathcal{P}]$ constructed from the input polynomials. In particular, it can be shown that if I is generated by $m \leqslant \dim(\mathbb{C}[\mathcal{P}])$ generic homogeneous polynomials of respective degrees $d_1, \ldots, d_m \geqslant 1$, then the Castenuovo-Mumford regularity of $\mathbb{C}[\mathcal{P}]/I$ equals $\operatorname{reg}(\mathbb{C}[\mathcal{P}]) + d_1 + \cdots + d_m - m$. The Castelnuovo-Mumford regularity has a relationship with the maximal degree occurring in the Gröbner basis, although this relationship is not always easy. In the classical setting, there is a special monomial ordering — the *grevlex* ordering — which has the nice property that under some conditions, the largest degree in the Gröbner basis equals the Castelnuovo-Mumford regularity [14]. This is an important tool for the complexity analysis of Gröbner bases algorithms. In polytopal algebras, there is no grevlex ordering and the situation is more complicated, so it is not easy to estimate precisely the cost of Gröbner bases computations. Fortunately, in the overdetermined case, we still get complexity results for sparse systems homogenized via these lattice polytopes:

Theorem 2.2.12. *[52, Lem. 5.2 modified for overdetermined systems, and Thm. 5.3] Let $\mathcal{P}$ be a normal lattice polytope in $\mathbb{R}^n$, and $f_1, \ldots, f_m$ with $m > n$ be a generic system of homogeneous polynomials of respective degrees $d_1 \geqslant \ldots \geqslant d_m \geqslant 1$ in the polytopal algebra $\mathbb{C}[\mathcal{P}]$. Then the maximal degree* dmax *occuring in a sparse Gröbner basis of the ideal $I \subset \mathbb{C}[\mathcal{P}]$ is bounded above by $\operatorname{reg}(\mathbb{C}[\mathcal{P}]) + 1 + \sum_{1 \leqslant i \leqslant n+1}(d_i - 1)$. The complexity of computing this sparse Gröbner basis is bounded above by $O(n \operatorname{HP}_{\mathcal{P}}(\mathsf{dmax})^{\omega})$, where $\operatorname{HP}_{\mathcal{P}}$ is the Ehrhart polynomial of $\mathcal{P}$, and $\omega < 2.373$ is a feasible exponent for matrix multiplication [122].*

When $m > n + 1$, the bound on dmax is not tight, and we provide an analog to the classical Fröberg conjecture in the case of polytopal algebras:

Conjecture 2.2.13. *[52, Conj. 6.3] Let $\mathcal{P}$, $d_1, \ldots, d_m$, $f_1, \ldots, f_m$ be as in Theorem 2.2.12. The Hilbert series of the graded ring $\mathbb{C}[\mathcal{P}]/\langle f_1, \ldots, f_m \rangle$ equals*

$$\left[\operatorname{HS}_{\mathcal{P}}(t) \prod_{1 \leqslant i \leqslant m} (1 - t^{d_i}) \right]_+ , \tag{2.1}$$

where the notation $[\cdot]_+$ means truncating the series at its first nonpositive coefficient, and $\operatorname{HS}_{\mathcal{P}}$ is the Ehrhart series of $\mathcal{P}$. Moreover, the maximal degree occurring in a sparse Gröbner basis of $\langle f_1, \ldots, f_m \rangle$ equals the index of the first nonpositive coefficient in Eq. (2.1).

The non-overdetermined case is more difficult. Initially, we presented complexity bounds for 0-dimensional systems, but there was a flaw in the proof, which was detected by Matías Bender, see `https://members.loria.fr/PJSpaenlehauer/data/FauSpaSva14_erratum.txt`. Fortunately, by using a slightly different method, Bender and Telen managed to design a solving algorithm for sparse systems whose complexity depends directly on the Castelnuovo-Mumford regularity, see [16].

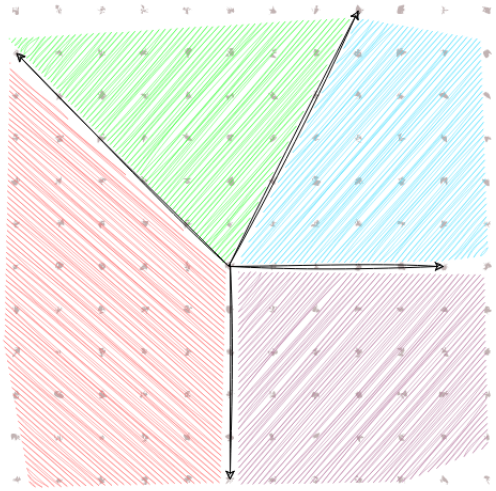


Figure 2.1: A complete polyhedral fan.

2.2.4 Dimensions and regular sequences in complete toric varieties from polyhedral fans

This section presents results obtained with Matías Bender and published in the *Journal of Algebra* [15].

Motivation. As seen in Section 2.2.3, polytopal algebras give a nice framework for computing with polynomials with monomial structures. Such polytopal algebras are in fact coordinate rings for projective toric varieties built from polyhedral fans. However, such coordinate rings lose a bit of flexibility as they encode the embedding of a toric variety inside a projective space. A more general algebraic object to compute with is the *Cox ring* of the toric variety. This is a polynomial ring which has a finer graded structure than polytopal algebras: it is graded by the full divisor class group of the toric variety. This allows more flexibility in the way sparse systems can be homogenized. The aim of the work described in this section is to study what happens when we consider such toric homogenizations, and to detect which homogenizations are well-suited with monomial structures. In particular, we study in which situations these homogenizations introduce unwanted high-dimensional components.

A nice feature of toric varieties built from polyhedral fans (see Fig. 2.1) is that they can be decomposed as a disjoint union of torus orbits isomorphic to $(\mathbb{C}^*)^i$. To prove our main dimension result on toric varieties, we look at what happens on each of these torus orbits since dimension is a local property.

The main tool to understand the dimension of common zeros of sparse polynomials with generic coefficients is the notion of *essentiality*:

Definition 2.2.14 (Essential family)**.** *A family of finite subsets $A_1, \dots, A_k \subset \mathbb{Z}^n$ is* essential *if* $\dim_{\mathbb{R}} \left(\text{AffineSpan}_{\mathbb{R}}(\sum_{i\in E} A_i)\right) \geqslant |E|$ *for every subset $E \subset [\![1, k]\!]$, where* $\text{AffineSpan}_{\mathbb{R}}(\sum_{i\in E} A_i)$ *denotes the smallest affine subspace in $\mathbb{R}^n$ containing $\sum_{i\in E} A_i$.*

The following proposition shows that the extremal-generic dimension of systems of Laurent polynomials can be computed by looking at essentiality properties of their monomial support. Extremal-genericity here means that we only require genericity assumptions on the coefficients

corresponding to the vertices of the Newton polytopes of the Laurent polynomials. The precise definition of extremal-genericity is given in [15, Def. 1.8].

Proposition 2.2.15 (Extremal-generic dimension over the torus)**.** *[15, Prop. 1.12] Let* $\mathbf{A} = (A_1, \dots, A_k)$ *be a family of* k *nonempty finite subsets of* $\mathbb{Z}^n$*. Let* $f_1, \dots, f_k \in \mathbb{C}[X_1^{\pm 1}, \dots, X_n^{\pm 1}]$ *be an extremal-generic system of polynomials with respective supports* $A_1, \dots, A_k$*. Then one of the two following propositions holds true:*

- *The family* $\mathbf{A}$ *is* essential *(Theorem 2.2.14) and* $\dim(\langle f_1, \dots, f_k \rangle) = n - k$*;*
- *The family* $\mathbf{A}$ *is* not essential *and* $\langle f_1, \dots, f_k \rangle = \mathbb{C}[X_1^{\pm 1}, \dots, X_n^{\pm 1}]$.

We start by recalling usual notation for toric geometry. Unless stated otherwise, we use the same notation as in [32].

In what follows, N is a Euclidean lattice, i.e. N is a free abelian group of finite rank equipped with a symmetric positive-definite bilinear form on $N_{\mathbb{R}} := N \otimes_{\mathbb{Z}} \mathbb{R}$ defining a Euclidean norm. We let M denote the dual lattice for the Euclidean norm, and we set $M_{\mathbb{R}} := M \otimes_{\mathbb{Z}} \mathbb{R}$. The notation $\mathbb{C}[M]$ denotes the Laurent polynomial ring with exponents in M and coefficients in $\mathbb{C}$. For $m \in M$, we let χ^m denote the associated character in $\mathbb{C}[M]$. For an affine semigroup $\mathcal{S} \subset M$, we let $\mathbb{C}[\mathcal{S}]$ denote the corresponding subalgebra of $\mathbb{C}[M]$.

The letter X denotes the n-dimensional normal toric variety associated to the lattices (N, M) and to a complete rational fan Σ over $N_{\mathbb{R}}$. We let $\Sigma(1)$ denote the rays of Σ and $S = \mathbb{C}[x_\rho : \rho \in \Sigma(1)]$ denote the Cox ring of X — called the *total coordinate ring* of X in [32, Ch. 5] — with its natural gradation by the *class group* $\mathrm{Cl}(X)$ of X [32, Def. 4.0.13]. By slight abuse of notation, we say that a divisor class $\alpha \in \mathrm{Cl}(X)$ is *Cartier* if it is the class of a Cartier divisor; this corresponds to elements in the Picard group of X [32, Thm. 6.0.20]. Similarly, we say that a divisor class is *effective* if it is the class of an effective divisor. Given a cone $\sigma \in \Sigma$, we denote by $\sigma(1)$ the rays of σ. For a ray $\rho \subset N_{\mathbb{R}}$, we let $u_\rho \in N$ denote its primitive vector, i.e. the only element in N satisfying $\mathbb{Z}_{\geqslant 0} u_\rho = \rho \cap N$. We let $B = \langle \prod_{\rho \notin \sigma(1)} x_\rho : \sigma \text{ cone of } \Sigma \rangle \subset S$ denote the irrelevant ideal. Since Σ is complete, for each $\alpha \in \mathrm{Cl}(X)$, the $\mathbb{C}$-vector space S_α of homogeneous elements of degree α in S is finite-dimensional [32, Prop. 4.3.8]. For simplicity, we call T-divisors the divisors on X which are invariant under the action of the torus [32, Exercise 4.1.1]. The class group $\mathrm{Cl}(X)$ is generated by the T-divisors $\{D_\rho : \rho \in \Sigma(1)\}$, where D_ρ is the divisor defined by the closure of the T-orbit $O(\rho)$ associated to $\rho \in \Sigma(1)$ [32, Thm. 3.2.6].

Given a finite-dimensional vector space $V \subset \mathbb{C}^n$, we say that a property holds generically on V if it holds on a dense subset for the Zariski topology. In this section, V is often a subspace of polynomial systems in $S_{\alpha_1} \times \cdots \times S_{\alpha_r}$, for degrees $\alpha_1, \dots, \alpha_r \in \mathrm{Cl}(X)$. When V is such a finite-dimensional space of polynomials, we say that a property holds for a generic system (in V) if it holds generically on V. Given a degree $\alpha \in \mathrm{Cl}(X)$, a monomial set $\mathcal{A} \in S_\alpha$, and a homogeneous polynomial $f \in S_\alpha$, we say that f has support $\mathcal{A}$ if all monomials with nonzero coefficients belong to $\mathcal{A}$.

In this section, all polytopes are supposed to be convex and bounded. We use the word *family* to denote finite multisets.

Our main technical result is the following dimension formula:

Theorem 2.2.16. *[15, Thm. 1.17] Let* $\alpha_1, \dots, \alpha_r \in \mathrm{Cl}(X)$ *be divisor classes and* $(\mathcal{A}_1, \dots, \mathcal{A}_r) \in S_{\alpha_1} \times \cdots \times S_{\alpha_r}$ *be monomial subsets. For each cone* σ *in* Σ*, let* $E_\sigma = \{i \in [\![1, r]\!] \mid \mathcal{A}_i^\sigma \neq \emptyset\}$*, where* $\mathcal{A}_i^\sigma$ *corresponds to the subset of monomials in* $\mathcal{A}_i$ *which do not vanish identically on the torus orbit* $O(\sigma)$ *(see [15, Notation 1.15] for a more precise definition). By abuse of notation,*

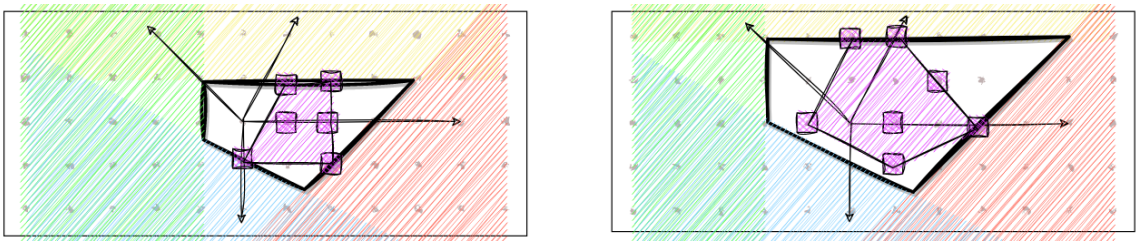


Figure 2.2: In this example, we start with two sparse polynomials $f_1 = \circ + \circ X^2 + \circ XY + \circ X^2Y + \circ XY^2 + \circ X^2Y^2$, $f_2 = \circ Y + \circ X^2 + \circ X^2Y + \circ X^4Y + \circ X^3Y^2 + \circ XY^3 + \circ X^2Y^3$, where the symbols $\circ$ represent generic coefficients in $\mathbb{C}$. We consider the toric variety V defined by the two-dimensional fan whose rays are generated by the four arrows. The black polytopes on the left and right correspond to divisor classes over V. The pink squares correspond to the monomials in the Cox ring of V after homogenization of f_1 (on the left) and f_2 (on the right) with respect to these two divisor classes. The dimension of the subvariety of V defined by the intersection of the two hypersurfaces (under genericity assumptions on the coefficients) corresponding to these two homogenized polynomials can be computed from the incidence properties between the pink and the black polytopes.

we say that E_σ is essential if $\{\mathcal{A}_i^\sigma : i \in E_\sigma\}$ is essential. Let Y a closed subscheme of X defined by a homogeneous ideal $\langle f_1, \dots, f_r\rangle \subset S$, where each $f_i \in S_{\alpha_i}$ is (extremal-)generic of degree α_i with support $\mathcal{A}_i$. Then the dimension of Y is

$$\dim(Y) = \max_{\substack{\sigma\in\Sigma \\ E_\sigma \text{ is essential}}} (n - \dim(\sigma) - |E_\sigma|)\,.$$

If none of the sets E_σ is essential, then Y is empty.

Remark 2.2.17. *Our main result actually holds under slightly weaker genericity assumptions: we only require "extremal-genericity", i.e. under some genericity assumptions on the coefficients corresponding to the vertices of the Newton polytopes of the polynomials in the system.*

Figure 2.2 provides a picture of Theorem 2.2.16 on an example.

The general Theorem 2.2.16 can also help us understand the behavior of (extremal-)generic systems in polytopal algebras. A nice feature is that this setting includes polytopal algebra over polytopes whose vertices are not lattice points. This is especially important for studying weighted homogeneous systems. The following result provides a combinatorial criterion which decides whether generic systems with prescribed support in a polytopal algebra are regular sequences:

Theorem 2.2.18. *[15, Thm. 3.3] Let $d_1, \dots, d_r \in \mathbb{Z}_{\geqslant 0}$ be degrees, and $A_1, \dots, A_r$ be monomial subsets in $k[\mathcal{P}]_{d_1}, \dots, k[\mathcal{P}]_{d_r}$ i.e. $A_i \subset (d_i \cdot P) \cap M$, and $f_1, \dots, f_r$ be a generic system with support $A_1, \dots, A_r$. Let Y be the closed subscheme of $\mathrm{Proj}(\mathbb{C}[\mathcal{P}])$ defined by the ideal $\langle f_1, \dots, f_r\rangle$. Then:*

1. *The subscheme Y is not empty.*
2. *The sequence $f_1, \dots, f_r$ is regular in $\mathbb{C}[\mathcal{P}]$ if and only if for any $F \in \mathrm{Faces}(P) \cup \{P\}$, $|\{i \in [\![1, r]\!] \mid A_i \cap (d_i \cdot F) \neq \emptyset\}| \geqslant \dim(F) + r - n$.*

In fact, this theorem provides an interesting statement when it is specialized to classical homogeneous systems, when the support of each polynomial is not the full set of monomials of some degree d:

Corollary 2.2.19. *[15, Coro. 3.7] Let $r \leqslant n$, $d_1, \dots, d_r > 0$ be positive integers and $A_1, \dots, A_r \subset \mathbb{C}[X_0, \dots, X_n]$ be subsets of monomials of respective degrees $d_1, \dots, d_r$, i.e. $A_i \subset \mathbb{C}[X_0, \dots, X_n]_{d_i}$. Then a generic homogeneous system $f_1, \dots, f_r \in \mathbb{C}[X_0, \dots, X_n]$ with respective monomial supports $A_1, \dots, A_r$ is regular if and only if for every subset $I \subset [\![0, n]\!]$,*

$$|\{j \in [\![1, r]\!] : A_j \cap \mathbb{C}[X_i : i \in I] \neq \emptyset\}| \geqslant |I| - 1 + r - n.$$

Another interesting case occurs when we specialize Theorem 2.2.18 to polytopal algebras where all polynomials involve all the monomials of some given degree:

Corollary 2.2.20. *[15, Coro. 3.4] Let $r \leqslant n$, $d_1, \dots, d_r \in \mathbb{Z}_{\geqslant 0}$ be positive integers, and $f_1, \dots, f_r$ be a system of generic homogeneous polynomials in $\mathbb{C}[\mathcal{P}]$ of respective degrees $d_1, \dots, d_r$. The sequence $f_1, \dots, f_r$ is regular if and only if for every $F \in \mathrm{Faces}(\mathcal{P}) \cup \{\mathcal{P}\}$,*

$$|\{i \in [\![1, r]\!] \mid (d_i \cdot F) \cap M \neq \emptyset\}| \geqslant \dim(F) + r - n.$$

The specialization to weighted homogeneous systems provides a general criterion to identify families of weighted degree at which regular sequences can be found. This answers an open question which was asked in [46].

Theorem 2.2.21. *[15, Thm. 3.8] Let $(a_0, \dots, a_n)$ be positive weights, $r \leqslant n$ and $(d_1, \dots, d_r)$ be positive integers. Then an extremal-generic sequence $f_1, \dots, f_r \in \mathbb{C}[X_0, \dots, X_n]$ of weighted homogeneous polynomials of respective weighted degrees $d_1, \dots, d_r$ is regular if and only if for any subset $J \subset [\![0, n]\!]$, the inequality $|\{i \in [\![1, r]\!] : d_i \in \sum_{j \in J} a_j \mathbb{Z}_{\geqslant 0}\}| \geqslant |J| + r - n - 1$ holds true.*

Complexity. Our criterions allows us to compute combinatorially the generic dimension of the intersection of the closures of hypersurfaces in the torus. A natural question is what is the theoretical complexity of computing these dimensions.

It appears that this corresponds to NP-hard problems. A first NP-hard reduction comes from the fact that the problem of deciding if there exists a monomial of weighted degree d in the weighted polynomial ring with weights $(w_0, \dots, w_n)$ is NP-hard, as it is an instance of a knapsack problem. However, even if we have as input the monomial support, then it is still NP-hard, because we proved that there is a reduction to the hitting set problem, which is known to be NP-hard. We refer to [15, Sec. 4] for more details on these reductions.

2.2.5 Quadratic fewnomials

This section presents joint work with Jules Svartz et Jean-Charles Faugère, and it has been published in the proceedings of the ISSAC 2016 conference [48].

In this work, we study a much less structured setting for sparse systems: systems where the set of monomials is sparse, but there is no clear structure coming from a polyhedron. More formally, we are working in a semigroup algebra $\mathbb{C}[S]$, where S is an affine semigroup which has no particular structure. In particular, we assume that it is far from normal, which destroys a lot of the nice structure of toric varieties built from fans.

In this section, we study the following setting: we set $\mathbf{M}$ a subset of quadratic monomials in $\mathbb{C}[X_1, \dots, X_n]$ of small cardinality, and we assume that it is chosen at random. The main parameter that has a strong impact on the structure of the problem in this setting is the number of square monomials in the support.

As often in polynomial system solving, it is not even easy to define what we mean by "solving polynomial systems". Therefore, our first goal is to define formally the algorithmic problem that we want to solve.

We fix a monomial subset $\mathbf{M} \subset \mathbb{C}[X_1, \ldots, X_n]$ which contains 1, and we consider its linear span $\mathrm{Span}_{\mathbb{C}}(\mathbf{M})$ which is a linear space of finite dimension.

Problem 2.2.22 (Effective fewnomial Nullstellensatz)**.** *Given $r \in \mathbb{Z}_{>0}$ and $(f_1, \ldots, f_r) \in \mathrm{Span}_{\mathbb{C}}(\mathbf{M})^r$ such that $\langle f_1, \ldots, f_r \rangle = \mathbb{C}[X_1, \ldots, X_n]$, compute $(h_1, \ldots, h_r) \in \mathbb{C}[X_1, \ldots, X_n]^r$ such that $\sum_{i=1}^r f_i \cdot h_i = 1$.*

A nice property is that we can entirely work in the semigroup algebra $\mathbb{C}[\mathbf{M}]$:

Proposition 2.2.23. *[48, Prop. 2.1] With the same notation and assumptions as in Theorem 2.2.22, there exists $(h_1, \ldots, h_r) \in \mathbb{C}[\mathbf{M}]^r$ such that $\sum_{i=1}^r f_i \cdot h_i = 1$.*

In the case of systems having finitely-many solutions, we define the following algorithmic problem, from which we can represent the solutions as roots of a univariate polynomial.

Problem 2.2.24 (Partial 0-dimensional fewnomial system solving)**.** *Given $r \in \mathbb{Z}_{>0}$, a system of polynomials $(f_1, \ldots, f_r) \in \mathrm{Span}_{\mathbb{C}}(\mathbf{M})^r$ which has finitely-many common zeros, and a monomial $\mu \in \mathbf{M}$, compute a univariate polynomial $P_\mu \in \mathbb{C}[\mu]$ which vanishes at all the common zeros of the system.*

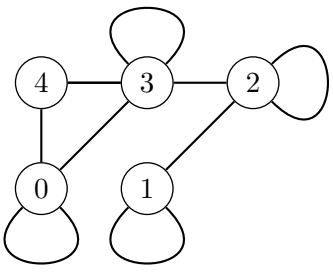
As often, we are interested in the situation when the coefficients of the polynomials are generic. Although it might look as if restricting the search in $\mathbb{C}[\mathbf{M}]$ seems to add more constraints, it also adds structure which will be convenient for computations. We focus on the simplest non-linear case: we assume that $\mathbf{M}$ contains monomials of degree $\leqslant 2$. Up to homogenizing by a new variable X_0, we then get that the homogenized $\widetilde{\mathbf{M}}$ is a subset of quadratic monomials in $\mathbb{C}[X_0, \ldots, X_n]$.

It appears that understanding the structure of this homogenized semigroup algebra is quite complicated. One way to get a handle on numerical data associated to this semigroup would be to compute the following series:

$$\sum_{i \geqslant 0} |\mathbf{M}^i| t^i,$$

where $\mathbf{M}^i = \{m_1 \cdots m_i \mid m_1, \ldots, m_i \in \mathbf{M}\}$. Such a computation can be approached by considering a graph with $n+1$ vertices constructed from $\mathbf{M}$: each monomial $X_i X_j$ in the homogenized support $\widetilde{\mathbf{M}}$ provides an edge between vertices i and j (loops are allowed).

Example 2.2.25. *Let us consider $\mathbf{M} = \{1, X_1^2, X_2^2, X_3^2, X_3, X_4, X_1X_2, X_2X_3, X_3X_4\}$. This monomial set can be represented by the following graph. Loops correspond to squares in $\mathbf{M}$.*



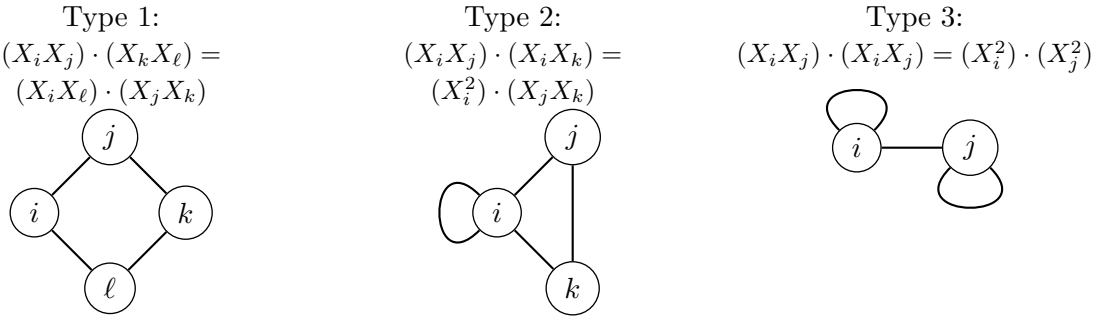


Figure 2.3: The three types of quadratic relations

Combinatorial information about $\mathbf{M}$ can be read off from the graph associated to $\mathbf{M}$. In particular, there are relations between elements in $\mathbf{M}$. In particular, quadratic relations of the form $\mu_1\mu_2 = \mu_3\mu_4$ correspond to special subgraphs in the graph associated to $\mathbf{M}$.

Proposition 2.2.26. *[48, Prop. 3.2] Let $\mathbf{M}$ be a set of monomials of degree $\leqslant 2$, and let G be the graph associated to it. The cardinality of $\mathbf{M}^2$ is $\binom{|\mathbf{M}|+1}{2} - \lambda(G) + \mathrm{clique}_4(G)$, where $\lambda(G)$ is the number of subgraphs of G isomorphic to any of the three graphs in Figure 2.3 and $\mathrm{clique}_4(G)$ is the number of 4-cliques in G.*

Using this structure we can identify some cases where the system has generically no solutions and where there are small certificates of inconsistency:

Theorem 2.2.27. *[48, Thm. 3.4] Let $(f_1, \ldots, f_r) \in \mathrm{Span}_{\mathbb{C}}(\mathbf{M})^r$ be a system with generic coefficients. Let G be the graph associated to $\mathbf{M}$, and $\nu(\mathbf{M})$ be the matching number of the subgraph of vertices in G with a loop. If $r \geqslant |\mathbf{M}| - \frac{\sqrt{1+8\nu(\mathbf{M})}-1}{2}$, then there exist polynomials $h_1, \ldots, h_r \in \mathrm{Span}_{\mathbb{C}}(\mathbf{M})^r$ such that $\sum_{i=1}^r f_i h_i = 1$.*

Corollary 2.2.28. *[48, Coro. 3.5] With the notation and under the assumptions of Theorem 2.2.27, there is an algorithm which solves Problem 2.2.22 within*

$$O\left(r|\mathbf{M}|\left(\binom{|\mathbf{M}|+1}{2} - \lambda(G) + \mathrm{clique}_4(G)\right)^{\omega-1}\right)$$

arithmetic operations, where ω is a feasible exponent for matrix multiplication.

The main point of this corollary is that the complexity is polynomial in the size of the input, which has $m|\mathbf{M}|$ coefficients in $\mathbb{C}$.

The next step in our analysis is to study what happens when the set of monomials is a random variable. Our main result in this setting is:

Theorem 2.2.29. *[48, Thm. 4.4] Let k be a fixed integer, $a_n, b_n \in \mathbb{Z}_{>0}$ be such that $a_n + b_n = n + k + 1$, and $\mathbf{M}$ be a subset of monomials of degree at most 2 in $\mathbb{C}[X_1, \ldots, X_n]$ distributed uniformly among those which contain 1, a_n nonsquare monomials, and b_n nonconstant square monomials. If $b_n = \Omega(n^{1/2+\varepsilon})$ for $\varepsilon > 0$, then the probability that the assumptions of Theorem 2.2.27 are satisfied tends to 1 as n grows.*

The proof Theorem 2.2.29 relies on the study of random graphs in the Erdös-Renyi model, since the premises of Theorem 2.2.27 can be read off from a graph constructed from $\mathbf{M}$.

Finally, we finish by studying the cases of fewnomial systems with all the squares. In this case, we can expect generic systems of n equations in n variables to have 2^n solutions. An interesting fact is that when the number of nonsquares is fixed, we can represent these solutions with a data structure which has polynomial size, and this representation can be computed in polynomial time:

Proposition 2.2.30. *[48, Cor. 5.2] Let k be a fixed positive integer, and $\mathbf{M}$ be a set of monomials containing 1, all the square monomials X_i^2, and k nonsquare monomials of degree at most 2. Let $(f_1, \dots, f_n) \in \mathrm{Span}_{\mathbb{C}}(\mathbf{M})$ be a 0-dimensional system with support $\mathbf{M}$. Then for any nonconstant $\mu \in \mathbf{M}$, Problem 2.2.24 with input $(f_1, \dots, f_n)$ and μ can be solved within $O(n^\omega)$ arithmetic operations in $\mathbb{C}$ as n grows, ω is a feasible exponent for matrix multiplication.*

2.3 Real polynomial systems with many positive solutions

The work presented in this section is joint work with Frédéric Bihan and Francisco Santos and it has been published in the *SIAM Journal on Applied Algebra and Geometry* [22].

In this paper, our main object of interest is the maximal number $\Xi_{d,k}$ of non-degenerate solutions in $\mathbb{R}^d_{>0}$ of a polynomial system $f_1 = \cdots = f_d = 0$ where $f_1, \dots, f_d$ are multivariate polynomials in $\mathbb{R}[X_1^{\pm 1}, \dots, X_d^{\pm 1}]$ which involve at most k distinct monomials. We focus on a method of tropical flavor, by constructing families of systems for which the number of real roots is reached asymptotically.

2.3.1 The number of positive solutions of fewnomials

Descartes' rule of signs is a classical tool which bounds the number of solutions of a polynomial systems in terms of the signs of the coefficients and the number of monomials.

Theorem 2.3.1 (Descartes' rule of signs)**.** *The number of positive real roots of a Laurent polynomial*

$$f(X) = \textstyle\sum_{i \in [\![1, k+2]\!]} a_i X^{u_i} \in \mathbb{R}[X^{\pm 1}]$$
$$u_1, \dots, u_{k+2} \in \mathbb{Z}, u_1 < u_2 < \cdots < u_{k+2}$$

is bounded above by the number of changes of signs betweeen consecutive coefficients:

$$|\{x : x \in \mathbb{R}_{>0}, f(x) = 0\}| \leqslant |i : i \in [\![1, k+1]\!], a_i\, a_{i+1} < 0|.$$

In particular, $|\{x : x \in \mathbb{R}_{>0}, f(x) = 0\}| \leqslant k + 1$.

A natural question is whether this bound can always be reached for fewnomials, i.e. polynomials with few monomials.

Question 2.3.2. *Given integers $u_1 < \cdots < u_{k+2}$, does there exist real numbers $a_1, \dots, a_{k+2} \in \mathbb{R}$ such that $|\{x : x \in \mathbb{R}_{>0}, \sum a_i x^{u_i} = 0\}| = k + 1$? If yes, how to construct such integers?*

This question can be solved by using asymptotic methods:

Proposition 2.3.3. *For $\varepsilon > 0$ sufficiently small, the polynomial $f_\varepsilon(X) = \sum_{i \in [\![1, k+2]\!]} (-1)^i \varepsilon^{u_i^2} X^{u_i}$ has exactly $k + 1$ positive roots. All of them are non-degenerate, i.e. the derivative does not vanish at the roots.*

Sketch of the proof. Fix $j \in [\![1, k+1]\!]$, and let us consider the map $X \mapsto \varepsilon^{-u_{j+1}-u_j} X$. Then set

$$g_\varepsilon(X) \stackrel{\text{def}}{=} f_\varepsilon(\varepsilon^{-u_{j+1}-u_j} X) \cdot \varepsilon^{u_j u_{j+1}}.$$

The convexity of the map $x \mapsto x^2$ implies that $g_\varepsilon(X) = \sum_{i \in [\![1,k+2]\!]} (-1)^i \varepsilon^{a_i} X^{u_i}$, where $a_j = a_{j+1} = 0$ and $a_i > 0$ for all $i \notin \{j, j+1\}$. This implies that for sufficiently small ε, g_ε has a non-degenerate root close to 1. Therefore, f_ε has a non-degenerate root of magnitude $\approx \varepsilon^{-u_j-u_{j+1}}$ for each $j \in [\![1, k+1]\!]$ as ε tends to zero. Hence f_ε has at least $k+1$ non-degenerate positive roots. Descartes' rule of sign concludes the proof: f_ε has at most $k+1$ positive roots. □

The convexity of the function $x \mapsto x^2$ is one of the keys of the proof, and other convex functions could be used. Viro's method generalizes this approach to higher-dimensional situations [118]: the *combinatorial patchworking method* allows us to define parametrized polynomials such that the real topology of the variety is asymptotically dictated by the combinatorial properties of a polyhedral construction.

The aim of our work is to consider the case of families of multivariate polynomials defining 0-dimensional sets. Our goal is to construct systems with few monomials but many positive solutions. The systems that we want to construct have the following form:

Definition 2.3.4. *Let $d > 0, k \geqslant 0$ be two integers. A* real fewnomial system *is the data of two matrices $A \in \mathrm{Mat}_{d,d+k+1}(\mathbb{R}), U \in \mathrm{Mat}_{d,d+k+1}(\mathbb{Z})$, which encodes the polynomial system $f_1^{(A,U)}, \dots, f_d^{(A,U)} \in \mathbb{R}[X_1, \dots, X_d]$ where*

$$f_i^{(A,U)}(X_1, \dots, X_d) = \sum_{j \in [\![1,d+k+1]\!]} \left(A_{i,j} \prod_{\ell \in [\![1,d]\!]} X_\ell^{U_{\ell,j}} \right).$$

Definition 2.3.5. *A point $x = (x_1, \dots, x_d) \in \mathbb{R}^d_{\geqslant 0}$ is called a* non-degenerate positive root *of the fewnomial system (A, U) if $f_1^{(A,U)}(x) = \cdots = f_d^{(A,U)}(x) = 0$ and if the Jacobian matrix of $f_1^{(A,U)}, \dots, f_d^{(A,U)}$ has full rank at x.*

We now state the main problem that we want to study:

Problem 2.3.6. *Given $k, d \in \mathbb{Z}_{\geqslant 0}$, what is the maximal number $\Xi_{d,k}$ of non-degenerate positive roots over all real fewnomial systems of type (d, k)?*

We emphasize that a natural question is whether $\Xi_{d,k}$ is indeed finite. Indeed, as we do not bound the degree of the equations, the number of complex solutions is not bounded. Surprisingly, the number of solutions of real fewnomial systems is actually finite: a major theorem by Khovanskii establishes the finiteness of $\Xi_{d,k}$.

Theorem 2.3.7. *[76]*

$$\Xi_{d,k} \leqslant 2^d 2^{\binom{d+k}{2}} (d+1)^{d+k}.$$

Since Khovanskii's theorem, there has been progress on the problem of finding tight upper bounds for $\Xi_{d,k}$. The state-of-the-art before our work can be summarized as follows:

- $\max((\lceil k/d \rceil + 1)^d, (\lceil d/k \rceil + 1)^k) \leqslant \Xi_{d,k} \leqslant (e^2+3) 2^{\binom{k}{2}} d^k/4$ [21, 23];
- $\Xi_{2,2} \leqslant 7$ [45].

We also define a function $\xi : [0,1] \to [1,\infty]$ which encodes the asymptotic behavior of $\Xi_{n,d}$:

Proposition 2.3.8. *[22, Sec. 2] For $d,k>0$, the limit $\lim_{n\to\infty} \Xi_{nd,nk}^{1/(dn+kn)} \in [1,\infty]$ exists. Its value depends only on d/k and it is bounded below by $(\Xi_{d,k})^{1/(d+k)}$.*

We define a function $\xi : \mathbb{Q}\cap]0,1[\to [\infty]$ which sends $d/(d+k)$ to $\lim_{n\to\infty} \Xi_{nd,nk}^{1/(dn+kn)} \in [1,\infty]$. This function is log-concave [22, Prop. 2.5], which implies that it can be extended by log-concavity to a function $\xi : [0,1] \to [1,\infty]$. This function is motivated by the fact that all lower bounds that we know for ξ are finite. Note that by log-concavity, $\xi(\alpha)$ is either finite for all $\alpha \in [0,1]$ or for none.

This raises several thought-provoking questions:

Question 2.3.9.
- *Is $\xi(\alpha)$ finite for some (equivalently for all) $\alpha \in]0,1[$?*
- *Does $\xi(1-\alpha) = \xi(\alpha)$ for all $\alpha \in [0,1]$?*
- *More generally, does $\Xi_{d,k} = \Xi_{k,d}$ for all $d,k>0$?*

Our main result is a new lower bound on ξ, which is obtained via a polyhedral construction:

Theorem 2.3.10. *[22, Thm. D] For every $\alpha \in]0,1[$,*

$$\xi(\alpha) \geqslant \left(\frac{\sqrt{\alpha^2+(1-\alpha)^2}+1-\alpha}{\alpha}\right)^{\frac{\alpha}{2}} \left(\frac{\sqrt{\alpha^2+(1-\alpha)^2}+\alpha}{1-\alpha}\right)^{\frac{1-\alpha}{2}}.$$

2.3.2 A polyhedral construction

The objective of our work is to provide a combinatorial method to construct fewnomial systems with many nondegenerate positive roots. Our construction mimics in the multivariate setting the technique for univariate polynomials described in Theorem 2.3.3.

In Theorem 2.3.3, the construction relies on the fact that up to scaling the variable, we can transform the polynomial in a small perturbation of a binomial $X^n - X^m$ which has a root at 1.

The d-dimensional analog of the univariate binomials $X^n - X^m$ are systems of d polynomials involving $d+1$-monomials and whose coefficients satisfy a condition similar to the change of sign in the univariate case. This condition can be expressed by the fact that the $d \times (d+1)$ matrix recording the coefficients of this system must have a positive kernel vector:

Definition 2.3.11. *A $d \times (d+1)$ matrix with real entries is* positively spanning *if the origin is in the interior of the convex hull of its column vectors. Equivalently, a matrix is positively spanning if and only if it has a vector with positive entries in its kernel.*

Real solutions of polynomial systems with d polynomials in d variables involving exactly $d+1$ monomials are handled by the following statement:

Proposition 2.3.12. *[22, Prop. 3.3] Let $\mathcal{A} = \{w_1,\dots,w_{d+1}\} \subset \mathbb{Z}^d$ be the set of vertices of a d-simplex in $\mathbb{R}^d$. For $i \in [\![1,d]\!], j \in [\![1,d+1]\!]$, fix real numbers $C_{ij} \in \mathbb{R}$. Set*

$$f_i = \sum_{1\leqslant j\leqslant d+1} C_{ij} X^{w_j}, \quad i \in [\![1,d]\!].$$

The system $f_1(X) = \cdots = f_d(X) = 0$ has at most one nondegenerate positive solution; it has one if and only if the matrix (C_{ij}) is positively spanning.

Next, our goal is to glue these building blocks to construct a system which can be reduced to a small perturbation of basic systems of d polynomials supported on $d+1$ monomials whose coefficient matrix is positively spanning.

In order to do so, we use convexity as in Theorem 2.3.3. We shall start with a d-dimensional regular simplicial complex (i.e. a simplicial complex which can be lifted via a convex function) where vertices are lattice points. The coordinates of the vertices will provide us with the matrix U. The regularity of the simplicial complex allows us (via a change of variables) to restrict to each simplex and therefore to reduce to the case where $k = 0$, i.e. there is exactly $d+1$ monomials.

We propose a variant of Viro's method which constructs a 1-dimensional parametric system such that for asymptotically small values of the parameter, each simplex will contribute to a nondegenerate solution if and only if the restriction of the system to this simplex corresponds to a positively spanning matrix.

We start with a finite set $\mathcal{A} = \{w_1, \dots, w_n\} \subset \mathbb{Z}^d$ which represents the monomial support of our system. In order to ensure that we are not in a degenerate case, we ask that the convex hull Q of $\mathcal{A}$ is full dimensional. Then we fix a *regular triangulation* Γ of Q with vertices in $\mathcal{A}$. A triangulation is regular if there exists a lifting function $\nu : Q \to \mathbb{R}$ which is convex and affine on each simplex of Γ but not affine on the union of two adjacent simplices. Next, we fix a coefficient matrix C_{ij} of dimension $d \times n$.

From $\mathcal{A}$, C and ν, we build the following parametric polynomial system, using the variable t as a positive real 1-dimensional parameter:

$$f_{i,t} = \sum_{j=1}^{n} C_{ij} t^{\nu(w_j)} X^{w_j}, \quad i \in [\![1, d]\!].$$

For each value of $t \in \mathbb{R}_{\geqslant 0}$, this defines a system of d polynomials in d variables and real coefficients. We are interested at what happens when t tends to 0. Our main point is that the number of nondegenerate solutions can be obtained asymptotically by looking at what happens at each simplex. For this, we say that a simplex Δ in Γ is positively decorated by C if the $d \times (d+1)$ matrix of C formed by the columns corresponding to the vertices of Δ is positively spanning.

The following theorem describes precisely the asymptotic number of non-degenerate solutions of such systems:

Theorem 2.3.13. *[22, Thm. 3.4] There exists $t_0 \in \mathbb{R}_{\geqslant 0}$ such that for all $t \in]0, t_0[$, the number of nondegenerate positive solutions of $f_{1,t}(X) = \cdots = f_{d,t}(X) = 0$ is bounded from below by the number of d-simplices in Γ which are positively decorated by C.*

If we want to obtain good lower bounds on the function ξ, we now have to construct positively decorated complexes with many facets but few vertices. In other words, we have reduced our problem to a combinatorial problem about simplicial complexes. In particular, our problem has links with a few other classical properties of simplicial complexes.

Balanced simplicial complexes are simplicial complexes whose 1-dimensional skeleton can be colored with $d+1$ colors. Another subfamily of simplicial complexes are bipartite simplicial complexes, which are complexes for which the adjacency graph of the $(d+1)$-dimensional simplices is bipartite (two $(d+1)$-dimensional simplices are adjacent if they share a common d-dimensional face). Our objective is to construct positively decorable simplical complexes, i.e. simplicial complexes together with a map which assigns a vector in $\mathbb{R}^d$ to each simplex so that

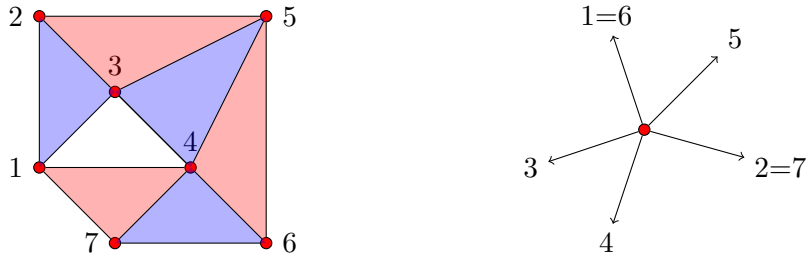


Figure 2.4: A two-dimensional simplicial complex whose adjacency graph is bipartite (left) and which is positively decorable (right) but not balanced. The white triangle 134 is not part of the complex.

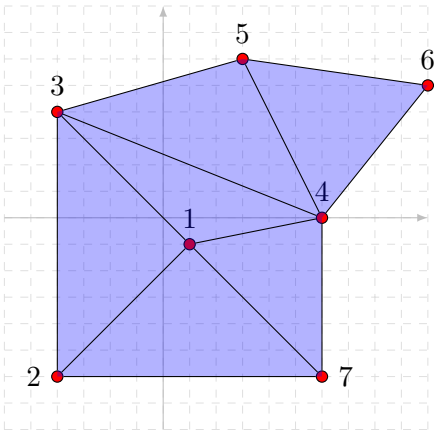


Figure 2.5: A balanced simplicial complex.

the $d+1$ vectors corresponding to the vertices of a $(d+1)$-simplex provides us with a positively spanning matrix.

Balanced, positively decorable, and bipartite simplicial complexes are related by the following statement:

Theorem 2.3.14. *[22, Thm. 5.5] Let Γ be a pure orientable complex. If Γ is balanced, then it is positively orientable. If Γ is positively orientable, then it is bipartite. If Γ is a triangulation of a simply connected manifold, then it is balanced if and only if it is bipartite.*

The cyclic polytope is an important geometric object since it is the polytope which maximizes the number of faces of each dimension for a given number of vertices. Therefore this is a good candidate to build a balanced simplicial complex on its boundary. We managed to build a simplicial complex with many positively decorated simplices. I will not detail the construction here (details can be found at [22, Sec. 6]), but the interesting fact is that we can count the number of positively decorated simplices as a number of matchings in a graph. Enumerating them gives a formula involving Delannoy numbers, and asymptotic analysis leads to the proof of our main result Theorem 2.3.10.

Figure 2.6 describes how our new lower bound improves on the best previously known lower bounds on the number of nondegenerate solutions of fewnomial systems.

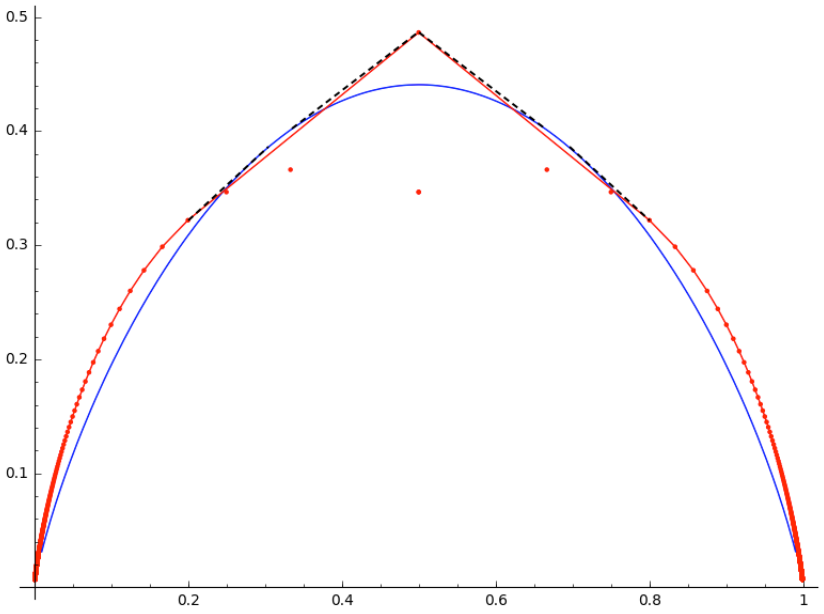


Figure 2.6: The different lower bounds for $\log \xi(\alpha)$, $\alpha \in (0,1)$. The red line is the best previously known lower bound. Our lower bound (blue curve) is above the previously known ones for $\alpha \in [0.2434, 0.3659]$. This range can be extended to $\alpha \in [0.2, 0.8]$ using log-concavity (dashed lines).

2.4 Computations of critical points

This section describes the work on the computation of critical points with Gröbner bases algorithms which I have done since the end of my PhD thesis. The results have been published in two articles: [111] in the *SIAM Journal on Optimization* and a joint work with Mohab Safey El Din [104] in the proceedings of the *ISSAC 2016 conference*.

Motivation. Critical points of regular maps are central objects in algebraic geometry, and in particular they play a crucial role in the study of real solutions of polynomial equations. A typical and representative example is learned in high-shool where the sign of $b^2 - 4\,a\,c$ controls the number of real solutions of the equation $a\,X^2 + b\,X + c = 0$. If we let $H \subset \mathbb{C}^4$ denote the hypersurface of tuples (X, a, b, c) satisfying $a\,X^2 + b\,X + c = 0$, then the complex numbers $(a, b, c) \in \mathbb{C}^3$ satisfying $b^2 - 4\,a\,c = 0$ are precisely the *complex critical values* of the projection map

$$\begin{array}{rccc} \pi : & H & \to & \mathbb{C}^3 \\ & (X, a, b, c) & \mapsto & (a, b, c) \end{array}$$

The critical values are the images of the *critical points* (also called *ramification points*) of the projection map: in our example, the critical points are the tuples $(X, a, b, c) \in \mathbb{C}^4$ which satisfy simultaneously $a\,X^2 + b\,X + c = 0$ and $\frac{\partial}{\partial X}(a\,X^2 + b\,X + c) = 0$. Real critical values describe the locus where complex conjugate points in the fiber $\pi^{-1}(a, b, c)$ of a point $(a, b, c) \in \mathbb{C}^3$ may coincide and give rise to two distinct real solutions, or vice versa. This is why the study of critical points is especially important for investigating the topology of real algebraic varieties: roughly speaking, the topology (for a real Euclidean metric) of the fiber of a regular map between real algebraic variety is "locally constant" away from singularities and critical values.

From a computational viewpoint, a nice property of critical points is that they can be computed over the complex numbers, as computational tools for polynomial systems are often designed for handling computations over algebraically closed fields. This observation is the starting point for the notion of *discriminant variety* [83], which has been studied in the context of computational real algebraic geometry.

In this section, we study what happens when the domain of the polynomial map is a smooth affine variety. More precisely, let $V \subset \mathbb{A}^n$ be a smooth variety, and we consider a polynomial function q on V. The critical locus Z of q is the subvariety $Z \subset V$ of points $\mathbf{x} \in \mathbb{A}^{\mathbf{n}}$ where the gradient of q is normal to the tangent space $T_{\mathbf{x}}V$.

If V is a complete intersection defined by r polynomials $f_1, \ldots, f_r \in k[X_1, \ldots, X_n]$ and q is encoded as a polynomial in the same polynomial ring, then the critical locus Z is formed by the points $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{A}^n$ such that

$$f_1(\mathbf{x}) = \cdots = f_r(\mathbf{x}) = 0 \quad \text{and} \quad \operatorname{rank} \left(\begin{bmatrix} \frac{\partial f_1}{\partial X_1}(\mathbf{x}) & \cdots & \frac{\partial f_1}{\partial X_n}(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_r}{\partial X_1}(\mathbf{x}) & \cdots & \frac{\partial f_r}{\partial X_n}(\mathbf{x}) \\ \frac{\partial q}{\partial X_1}(\mathbf{x}) & \cdots & \frac{\partial q}{\partial X_n}(\mathbf{x}) \end{bmatrix} \right) \leqslant r$$

The matrix occurring in the inequality is the *Jacobian matrix* of $(f_1, \ldots, f_r, q)$. The ideal in $k[X_1, \ldots, X_n]$ defining the critical locus is therefore generated by $f_1, \ldots, f_r$ and by the $(r+1) \times (r+1)$ minors of the Jacobian matrix. This raises some computational questions. For instance, the minors of a polynomial matrix typically have a very high degree, and they may be complicated to compute with. A classical way to handle this problem computationally is to use *Lagrange multipliers*, i.e. to encode the rank condition in new variables representing kernel vectors of the matrix.

In practice, this leads to the following polynomial system over $k[X_1, \ldots, X_n] \otimes k[Y_1, \ldots, Y_{r+1}]$:

$$\begin{cases} f_1(X_1, \ldots, X_n) = \cdots = f_r(X_1, \ldots, X_n) = 0, \\ \begin{bmatrix} Y_1 & \ldots & Y_{r+1} \end{bmatrix} \cdot \begin{bmatrix} \frac{\partial f_1}{\partial X_1}(X_1, \ldots, X_n) & \cdots & \frac{\partial f_1}{\partial X_n}(X_1, \ldots, X_n) \\ \vdots & \ddots & \vdots \\ \frac{\partial f_r}{\partial X_1}(X_1, \ldots, X_n) & \cdots & \frac{\partial f_r}{\partial X_n}(X_1, \ldots, X_n) \\ \frac{\partial q}{\partial X_1}(X_1, \ldots, X_n) & \cdots & \frac{\partial q}{\partial X_n}(X_1, \ldots, X_n) \end{bmatrix} = 0. \end{cases}$$

Since these equations are homogeneous with respect to $Y_1, \ldots, Y_{r+1}$, the ideal they generate defines a variety in $\mathbb{A}^n \times \mathbb{P}^r$ whose projection to $\mathbb{A}^n$ is the critical locus. This formulation avoids the problem of the high degrees of the minors. However, this algebraic modeling features another problem: the system is bihomogeneous, and this is more difficult to handle computationally. Even more problems arise when the variety V is not a complete intersection, or if V is not smooth. A typical example is described in Section 2.5.2.

To estimate the complexity of the computation of critical points by using Gröbner bases, two numerical invariants give meaningful information: the *regularity* and the *degree* of the ideal. The path to obtain complexity estimates often require to be able to estimate the values of these numerical indicators of the complexity.

Sometimes, these indicators are not sufficient, and we require more numerical information of the topology of the geometrical objects. The degrees of the cycles associated to the critical locus of the projection of a projective variety to a linear space provide more information for projective varieties: these numbers are called the *polar degrees* of the variety. These degrees can

be related to the degrees of the Chern classes, and to the coefficients of the Hilbert polynomial of the variety.

2.4.1 Varieties defined by generic polynomials

The work presented in this section has been published in [111].

This work investigates the complexity of computing critical points with Gröbner bases under genericity assumptions on the coefficients of the input. As usual, we consider the computational model where we count at unit cost arithmetic operations in the base field, and we neglect all other operations.

The main result (see Theorem 2.4.1 below) states that computing critical points with Gröbner bases algorithms is "almost polynomial" in the size of the output, under genericity assumptions.

To measure the complexity of computing the Gröbner basis of a 0-dimensional ideal $I \subset k[X_1, \ldots, X_n]$, it is often relevant to compare it to the degree $\dim_k(k[X_1, \ldots, X_n]/I)$ of the ideal, as this measures the geometric complexity of the object. In particular, the number of coefficients in a reduced Gröbner basis of a zero-dimensional ideal I in $k[X_1, \ldots, X_n]$ is bounded below by $\dim_k(k[X_1, \ldots, X_n]/I)$ and it cannot exceed $n \dim_k(k[X_1, \ldots, X_n]/I)^2$.

The degree of generic critical points is known, see [93, Thm. 2.2]. If I is the ideal generated by $f_1, \ldots, f_r$ (of respective degrees $d_1, \ldots, d_r$) and by the $(r+1)$-minors of the Jacobian matrix of $(f_1, \ldots, f_r, q)$, under genericity assumptions the degree of I is

$$\delta := \dim_k(k[X_1, \ldots, X_n]/I) = \left(\prod_{1 \leqslant i \leqslant p} d_i \right) \sum_{i_0 + \cdots + i_p = n-p} (d_0 - 1)^{i_0} \cdots (d_p - 1)^{i_p}. \tag{2.1}$$

An interesting feature of this formula is that it is much simpler when all degrees are 2: this is the problem of quadratic programming, which is well-known by practitioners to be much easier than the general case. In this case, the degree specializes to $2^p \binom{n}{p}$. We assume in the sequel that $\max(d_0, \ldots d_p) \geqslant 2$, since the problem degenerates if all polynomials are linear.

Theorem 2.4.1. *[111, Thm. 1.1] Let $d_0, \ldots, d_p$ be positive integers. Set $D = \max_{[\![0,p]\!]}(d_i)$ and let $q, f_1, \ldots, f_p \in \mathbb{Q}[X_1, \ldots, X_n]$ be polynomials of degrees $d_0, d_1, \ldots, d_p$ with generic coefficients. Let A (resp. G) be the arithmetic (resp. geometric) average of the multiset*

$$\{d_1, \ldots, d_p, \underbrace{D-1, \ldots, D-1}_{n-p \text{ times}}\}.$$

Then a lexicographical Gröbner basis of the ideal vanishing on the critical points of the map q restricted to the variety $f_1 = \cdots = f_p = 0$ can be computed within $\delta^{O(\log A / \log G)}$ arithmetic operations in $\mathbb{Q}$.

Theorem 2.4.1 states in particular that the complexity is polynomial in the degree for families of tuples $d_0, \ldots, d_p$ such that $\log A / \log G$ is bounded. Note that to construct families of degrees such that $\log A / \log G$ is unbounded, we have to consider examples where the degrees are very badly balanced.

Theorem 2.4.1 has a few interesting consequences, which can be obtained directly by expanding δ using Equation (2.1) and by doing classical asymptotic analysis.

Corollary 2.4.2. *[111, Coro. 1.2] The complexity bound in Theorem 2.4.1 never exceeds the best previously known bound $D^{O(n)}$ [38, Sec. 10.3].*

Even though the bound in Theorem 2.4.1 does not improve the previously known best bound in the worst case, there are several subfamilies where it provides substantial improvements. Two examples are given below.

Corollary 2.4.3. *[111, Coro. 3.9] If $d_0 = \cdots = d_p = 2$, then the complexity bound in Theorem 2.4.1 specializes to $n^{O(p)}$.*

Theorem 2.4.3 states that for quadratic programming of fixed codimension p, the complexity is actually polynomial in n, whereas the previous best bounds were exponential.

Corollary 2.4.4. *[111, Coro. 3.8] If $\max(d_0, \ldots, d_p) \geqslant 3$, then the complexity bound in Theorem 2.4.1 specializes to $\delta^{O(n/(n-p))}$.*

Theorem 2.4.4 together with Theorem 2.4.3 show that if codimension p of V is bounded, then the complexity is polynomial in the number of critical points.

The Eagon-Northcott complex. As mentioned above, in order to obtain the complexity estimate in Theorem 2.4.1, we need to estimate the degree of regularity of the problem. In this work, in order to obtain such an estimate, we used a tool from commutative algebra called the Eagon-Northcott complex. Under some conditions, this algebraic object describes the structure of ideals generated by the maximal minors of a matrix. This provides the information required to understand the contribution of the maximal minors of the Jacobian matrix to the regularity.

If R is a graded ring, F and G are free R-modules of respective ranks f and g with $g < f$, we consider a morphism $\alpha : F \to G$ of graded R-modules. We can consider the associated morphism $\wedge^g\alpha : \wedge^g\alpha : \wedge^g F \to \wedge^g G$. Note that $\wedge^g G$ is a free module of rank 1, and it is therefore isomorphic to R. We emphasize that this isomorphism is canonical once bases of F and G have been fixed. In this case, the image of $\wedge^g\alpha$ is the ideal in R generated by the maximal minors of the matrix representing α with respect to the fixed bases.

The Eagon-Northcott complex associated to α is a sequence of maps

$$0 \to (\mathrm{Sym}_{f-g} G)^* \otimes \wedge^f F \to (\mathrm{Sym}_{f-g-1} G)^* \otimes \wedge^{f-1} F \to (\mathrm{Sym}_{f-g-2} G)^* \otimes \wedge^{f-2} F \to \cdots \to \\ \to (\mathrm{Sym}_1 G)^* \otimes \wedge^{g+1} F \to \wedge^g F \xrightarrow{\wedge^g\alpha} \wedge^g G.$$

We will not describe the details of the maps in this thesis, but we state a few important properties:

- the successive composition of maps is zero;
- all objects are free R-modules of finite rank;
- if R, F, G are graded, then all maps are compatible with the grading.

For more details on the Eagon-Northcott complex, we refer to [43, Sec. A.2.6].

Moreover, if $R = k[X_{11}, \ldots, X_{fg}]$, $(e_1, \ldots, e_f)$ (resp. $(e'_1, \ldots, e'_g)$) is a basis of F (resp. G), and α is the map defined on the bases by $\alpha(e_i) = \sum_{1 \leqslant j \leqslant g} x_{ij} e'_j$, then the Eagon-Northcott complex is exact.

Roughly speaking, this means that provided that the entries of the matrix defining the map α are "generic enough", then the Eagon-Northcott complex is a free resolution of R/I, where I is the ideal generated by the maximal minors of the matrix representing α.

In our study of critical points, we apply this strategy to the Jacobian matrix of the input system: the genericity assumptions on the coefficients of the input system $(f_1, \ldots, f_r)$ and the function q will be sufficient so that the Eagon-Northcott complex associated to the map

$$\begin{array}{rccc} \alpha: & k[X_1, \ldots, X_n]^n & \to & k[X_1, \ldots, X_n]^{r+1} \\ & e_i & \mapsto & \frac{\partial q}{\partial x_i} e'_{r+1} + \sum_{1 \leqslant j \leqslant r} \frac{\partial f_j}{\partial x_i} e'_j \end{array}$$

is exact.

To conclude our analysis, we need to tensor the Eagon-Northcott complex with the Koszul complex associated to $(f_1, \ldots, f_r)$, which yields a complex of free $k[X_1, \ldots, X_n]$-modules for which the image of the last map is $k[X_1, \ldots, X_n]/(I + \langle f_1, \ldots, f_r \rangle)$. Finally we add the grading into the picture, using the natural degree of the input polynomials. This provides us with exact sequences of vector spaces, leading to Hilbert series, and ultimately this provides us with a value for the degree of regularity.

This is summed up in the following theorem:

Theorem 2.4.5. *[111, Coro. 3.2] For generic homogeneous polynomials $(q, f_1, \ldots, f_p)$ in the ring $k[X_1, \ldots, X_n]$ of respective degrees $(d_0, d_1, \ldots, d_r)$, let us consider the ideal generated by $f_1, \ldots, f_p$ and by the maximal minors of* $\mathrm{Jac}(f_1, \ldots, f_p, q)$*. Then there exists and positive integer $d \in \mathbb{Z}_{>0}$ such that $I_d = R_d$; the* degree of regularity *is the smallest such integer, and it equals:*

$$\mathrm{d_{reg}} = (n - p - 1) \max_{0 \leqslant i \leqslant p} \{d_i - 1\} - n - p + d_0 + 2 \sum_{1 \leqslant i \leqslant p} d_i.$$

Once we have access to the degree of regularity, it automatically translates into complexity bounds for Gröbner basis computations. There is still a last issue remaining: for applications, we actually want complexity results for affine systems.

A way to handle this issue is to check that nothing bad happens at infinity. This is quite technical, but it can be done methodically, leading to the following complexity estimate, which uses the fact that the grevlex ordering behaves nicely with respect to homogeneization and dehomogenization:

Theorem 2.4.6. *[111, Thm. 3.4 and its proof] For generic non-homogeneous $(q, f_1, \ldots, f_p)$ in $k[X_1, \ldots, X_n]$ of fixed degrees $(d_0, d_1, \ldots, d_p)$, let I be the ideal generated by $f_1, \ldots, f_p$ and by the maximal minors of* $\mathrm{Jac}(f_1, \ldots, f_p, q)$*. Then a grevlex Gröbner basis of I can be obtained by computing the row echelon form of the Macaulay matrix in degree*

$$(n - p - 1) \max_{0 \leqslant i \leqslant p} \{d_i - 1\} - n - p + d_0 + 2 \sum_{1 \leqslant i \leqslant p} d_i.$$

As the complexity of computing a row echelon form is well-known, this allows us to prove the complexity bounds in Theorem 2.4.1, Theorem 2.4.3 and Theorem 2.4.4. We emphasize that the solving process usually requires two steps: the first one is the computation of a Gröbner basis with respect to the grevlex ordering which amounts to row echelon form computations, the second one is the change of ordering via the FGLM algorithm whose complexity is at most cubic in the degree of the ideal.

Using the Eagon-Northcott complex to compute numerical invariants of determinantal ideals is a classical idea, which is for instance featured in [26]. Another approach is to use combinatorial properties of the numerical invariants associated to determinantal ideals and varieties. For instance, *Grothendieck polynomials* provide combinatorial methods to access these numerical invariants, see e.g. [79, Thm. A] and [111, Sec. 3.3].

2.4.2 Smooth varieties

The results presented are joint work with Mohab Safey El Din and they are published in [104].

The aim of the work presented in this section is to be able to work with more general families of polynomials than those satisfying the strong genericity assumptions in Section 2.4.1. Our complexity results will depend on some numerical indicators of the input polynomial system which are more precise than the degree of the input. More precisely, we study the problem of computing the critical points of a map f on a smooth algebraic variety V (which is given as a set of generators of a defining ideal). The main numerical indicators of the "complexity" of the input are the *polar degrees* of V. The i-th polar degree $\delta_i(V)$ is the degree of critical locus of a generic linear projection to $\mathbb{C}^{i+1}$.

The main result provide a formula for the degree of the critical locus for a generic polynomial of degree d on a smooth projective variety V in terms of d and the polar degrees of V.

Theorem 2.4.7. *[104, Thm. 1] The degree of the critical locus of a generic polynomial g (with $\deg(g) > 1$) on a smooth equidimensional variety V (whose projective closure is smooth) of dimension d is*

$$\sum_{j \in [\![0,d]\!]} \delta_{j+1}(V)(\deg(g)-1)^j.$$

This degree is interesting in practice for two reasons: it encodes the size of the output (i.e. the number of critical points). Moreover, there are algorithms whose complexity relies on this value. In particular, we show how to use a geometric resolution algorithm [11] in order to compute the critical points with a complexity which is essentially quadratic in the value in Theorem 2.4.7, under genericity assumptions on the function g [104, Thm. 16].

2.5 Structured low-rank approximation

Structured Low-Rank Approximation (abbreviated SLRA) is a general framework that can encode several computational problems. This problem can be stated informally as follows: given a linear space of structured matrix $E \subset \mathrm{Mat}_{p,q}(\mathbb{R})$ and a matrix $M \in \mathrm{Mat}_{p,q}(\mathbb{R})$, we want to find a nearby matrix $M^\star$ such that $M^\star$ belongs to E and $M^\star$ has a low rank $r \leqslant \min(p,q)$. A typical case of application appears when M is a perturbation of a low-rank structured matrix, so we know that it is close to some low-rank solution $M^\star$. A typical difficulty is that the solution set of structured low-rank approximation usually has positive dimension, so the problem does not admit a unique solution.

To handle this problem of non-unicity, we can use two approaches:

- we may just want a nearby solution of given rank r, focussing on fast convergence.
- to restore unicity, we may search for "the" closest solution, which is unique in most cases. This approach is particularly well-suited for symbolic and algebraic methods, because this can be expressed algebraically.

It is important to note that in applications, the closest solution may not have any relevant meaning. Often the problem comes from a perturbation of a given low-rank structured matrix, and the closest solution of the corresponding is usually distinct from the matrix we started from.

Example 2.5.1 (Approximate univariate GCD)**.** *Let us consider the polynomials: $f = X^2 - 2$, $g = X^2 - (\sqrt{2}+\sqrt{3})X + \sqrt{6}$ in $\mathbb{R}[X]$. These two polynomials share a common GCD:* $\mathrm{GCD}(f,g) = X - \sqrt{2}$.

However, if we have access to these polynomials only with floating point coefficients, we see that $f \approx 1.00 \cdot X^2 + 0.00 \cdot X - 2.00$, $g \approx 1.00 \cdot X^2 - 3.14 \cdot X + 2.45$. *Now it is much less obvious that these two polynomials share a common GCD. One way to try to formalize this problem is to build the Sylvester matrix.*

$$\mathrm{Syl}(f,g) = \begin{bmatrix} 1.00 & 0.00 & -2.00 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 1.00 & 0.00 & -2.00 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1.00 & 0.00 & -2.00 \\ 1.00 & -3.14 & 2.45 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & 1.00 & -3.14 & 2.45 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1.00 & -3.14 & 2.45 \end{bmatrix}.$$

Bold zeros represent formal zeros here, whereas 0.00 *are approximate. The approximate determinant of* $\mathrm{Syl}(f,g)$ *is* 0.08, *which seems to indicate the pair* (f,g) *is close to a pair* $(f^\star, g^\star)$ *which actually has a nontrivial GCD in* $\mathbb{R}[X]$. *Finding such a pair is equivalent to finding a Sylvester matrix*

$$\mathrm{Syl}(f^\star,g^\star) = \begin{bmatrix} a_2 & a_1 & a_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & a_2 & a_1 & a_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & a_2 & a_1 & a_0 \\ b_2 & b_1 & b_0 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & b_2 & b_1 & b_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & b_2 & b_1 & b_0 \end{bmatrix}$$

which is close to $\mathrm{Syl}(f,g)$ *and which is not full rank. The set of possible Sylvester matrices form a linear space in* $\mathrm{Mat}_{6,6}(\mathbb{R})$. *Of course the closest such matrix need not correspond to the initial exact pair* $(X^2-2, X^2-(\sqrt{2}+\sqrt{3})X+\sqrt{6})$.

Let us formalize the problem of Structured Low-Rank Approximation:

Problem 2.5.2 (Structured Low-Rank Approximation (SLRA))**.** *Let* E *be an linear (or affine) subspace in* $\mathrm{Mat}_{p,q}(\mathbb{R})$, $M \in E$ *be a matrix, and* $r < \min\{p,q\}$ *be a positive integer. Find a matrix* $M^\star \in E$ *such that* $\mathrm{rank}(M^\star) \leqslant r$ *and* $\|M - M^\star\|$ *is "small".*

2.5.1 Numerical approach

The work presented in this section is joint work with Éric Schost and it has been published in the journal *Foundations of Computational Mathematics* [107].

The main goal of this section is to design a Newton-like iterative numerical algorithm in order to achieve quadratic convergence. We start by describing one of the main ingredients of (unstructured) low-rank approximation over the reals: the *Singular Value Decomposition*, which is featured in Eckart-Young theorem.

Definition 2.5.3. *A* Singular Value Decomposition *(SVD) of a real matrix* $M \in \mathrm{Mat}_{p,q}(\mathbb{R})$ *is given by three matrices* $U \in \mathrm{Mat}_{p,p}(\mathbb{R})$, $V \in \mathrm{Mat}_{q,q}(\mathbb{R})$, $\Sigma \in \mathrm{Mat}_{p,q}(\mathbb{R})$ *such that* U *and* V *are orthonormal matrices,* $M = U \cdot \Sigma \cdot V^\intercal$, Σ *is diagonal and its diagonal entries are nonnegative and sorted in non-increasing order. The diagonal entries of* Σ *are called the* singular values *of* M.

The space $\mathrm{Mat}_{p,q}(\mathbb{R})$ of real matrices has a canonical Euclidean scalar product, i.e. a canonical positive definite bilinear form:

$$\begin{array}{rccc} \langle \_, \_ \rangle : & \mathrm{Mat}_{p,q}(\mathbb{R}) \times \mathrm{Mat}_{p,q}(\mathbb{R}) & \to & \mathbb{R} \\ & (M,N) & \mapsto & \mathrm{Trace}(M \cdot N^\intercal). \end{array}$$

The associated Euclidean norm $\|M\| \stackrel{\text{def}}{=} \sqrt{\langle M, M\rangle}$ is often called the *Frobenius norm* on $\mathrm{Mat}_{p,q}(\mathbb{R})$.

Theorem 2.5.4 (Eckart-Young theorem)**.** *Let $M \in \mathrm{Mat}_{p,q}(\mathbb{R})$ be a real matrix, (U, V, Σ) be an SVD of M, and $r > 0$ be a positive integer. Let $\Sigma^\star$ be the matrix obtained by setting the $\min(p, q) - r$ smallest singular values to zero in Σ, and set $M^\star = U \cdot \Sigma^\star \cdot V$. Then $M^\star$ minimizes the Frobenius distance to M among all matrices of rank at most r in $\mathrm{Mat}_{p,q}(\mathbb{R})$.*

The Eckart-Young theorem reduces the problem of *Unstructured Low-Rank Approximation* (i.e. $E = \mathrm{Mat}_{p,q}(\mathbb{R})$ in Theorem 2.5.2) to the computation of a SVD.

When we deal with Structured Low-Rank Approximation, the problem is that (except in a few very special case), the Unstructured Low-Rank Approximation of a matrix $M \in E$ given by the SVD need not belong to E. Nevertheless, by using Euclidean scalar product on matrices, we can project back on E. Iterating this process leads to an algorithm called *Cadzow's algorithm* or *lift-and-project* and which is one of the main tools studied for Structured Low-Rank Approximation. In particular, its convergence properties have been investigated. It converges linearly: provided that the starting matrix is close enough to a valid solution, the sequence of matrices $(M_i)_{i \geqslant 0}$ constructed via this iterative approach converges to a matrix $M^\star \in E$ which has rank at most r, and there exists $0 < \gamma < 1$ such that $\|M_{i+1} - M^\star\| \leqslant \gamma \|M_i - M^\star\|$ for all sufficiently large i.

The goal of the algorithm that we propose is to reach *quadratic convergence* under similar assumptions (in particular that the initial matrix for the iteration is sufficiently close to an admissible solution), i.e. we propose an iterative process such that $\|M_{i+1} - M^\star\| \leqslant \gamma \|M_i - M^\star\|^2$.

We emphasize that quadratic convergence leads to an exponential speedup, because reaching a matrix at distance less than $\varepsilon > 0$ from the limit matrix requires $O((\log_2 \varepsilon)^{-1})$ iterations, whereas this requires $O(\varepsilon^{-1})$ iterations with an algorithm having linear convergence.

Our main result is the following:

Theorem 2.5.5 (Sketch)**.** *[107, Thm. 4.1] We provide an algorithm — called NewtonSLRA — which computes a map $\varphi : E \to E$. Let $\mathcal{D}_r \subset \mathrm{Mat}_{p,q}(\mathbb{R})$ be the set of matrices of rank at most r. For any matrix $M \in E$ sufficiently close to a point where E and $\mathcal{D}_r$ intersect transversely, the sequence $(M_i)_{i \in \mathbb{Z}_{\geqslant 0}}$ defined by $M_0 = M$ and $M_{i+1} = \varphi(M_i)$ converges towards a matrix $M^\star \in E \cap \mathcal{D}_r$. The rate of convergence of this iterative process is quadratic. Computing φ involves the computation of a SVD, and a number of arithmetic operations in $\mathbb{R}$ which is polynomial in $p, q, r, \dim(E)$.*

Some problems of convergence and conditioning can occur when the initial matrix is close to a singular point of $\mathcal{D}_r$ or to a point where E and $\mathcal{D}_r$ do not intersect transversely. Roughly speaking, in order to converge, the closer we are to a singularity, the closer to an admissible solution the initial point of the iteration should be.

A way to formalize this is via the limit operator Φ which sends a matrix M to the limit of the sequence $(M_i)_{i \in \mathbb{Z}_{\geqslant 0}}$.

Theorem 2.5.6 (Sketch)**.** *[107, Thm. 4.2] The limit operator Φ is well-defined and continuous around any matrix $M \in E \cap \mathcal{D}_r$ at which the intersection is transverse. Moreover, Φ is differentiable at M and the derivative of Φ is the projection to the tangent space of $E \cap \mathcal{D}_r$ at M.*

This theorem mainly states that the limit operator is locally a projection on the tangent space of the set of admissible low-rank matrices, provided that the initial point is close enough to a valid solution.

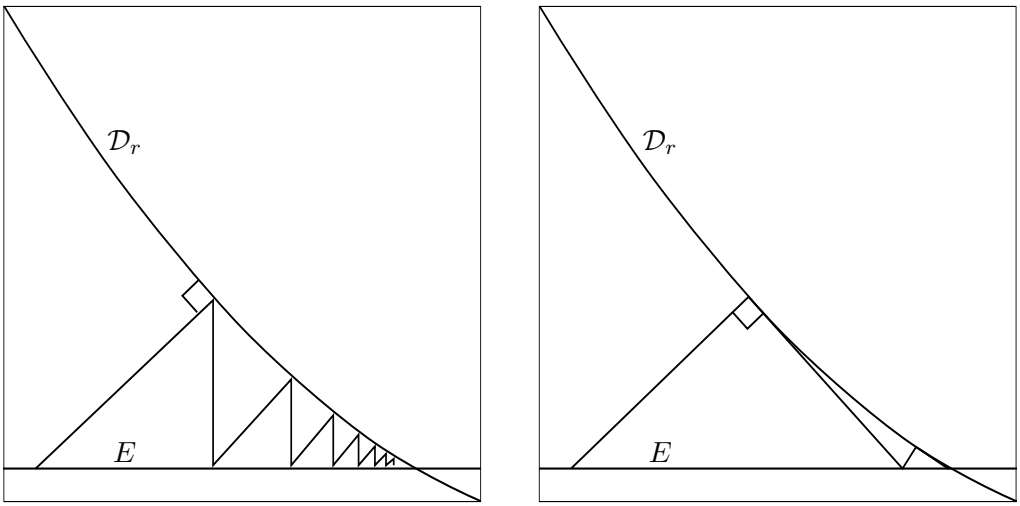


Figure 2.7: Cadzow's algorithm (left) and NewtonSLRA (right)

Description of the algorithms

The iterative process is a variant of Newton's method. Its main idea is illustrated in Figure 2.7. The graph on the left pictures Cadzow's algorithm. The orthogonal projection on $\mathcal{D}_r$ is obtained via the SVD computation and Eckart-Young theorem. The core idea of NewtonSLRA is depicted on the picture on the right: in fact, the SVD contains more information than what is exploited in Cadzow's algorithm; it also contains a description of the tangent space to $\mathcal{D}_r$ via rows in U and V which correspond to the nonzero singular values. Therefore, instead of projecting back on E orthogonally, we use the tangent space to project in the spirit of Newton's method. This is the main change that allows us to obtain a quadratic convergence.

We give in Algorithm 5 and Algorithm 6 two descriptions of NewtonSLRA, which would be fonctionally equivalent if computations where done exactly with real numbers. Since we use floating-point numbers, these two variants have distinct behaviors, and the choice of which variant should be used depends on the parameters. In particular, the first one is more efficient when r is small compared to $\dim(E)$, whereas the second one performs better when r is large and $\dim(E)$ is small. The main difference lie in the dimensions of an intermediate matrix which is constructed to compute the projection of E via the tangent space to $\mathcal{D}_r$.

In these algorithms, E may be an affine space, and we let E^0 denote the underlying vector space.

Applications and experiments

We studied the behavior of our algorithms on typical cases of SLRA occuring in applications or in other areas of computational mathematics. The algorithms have been implemented in a Maple package freely available at the following url: `https://members.loria.fr/PJSpaenlehauer/data/software/NewtonSLRA_notes.html`

The first instance of SLRA that we looked at is the problem of the approximate univariate gcd: here our linear space of matrices is a space of Sylvester matrices. We compared our software with state-of-the-art software dedicated to the approximate gcd problem: `GPGCD` and `uvGCD`. We observed that our implementation has a quicker convergence rate than `GPGCD` and `uvGCD`. This

Algorithm 5 one iteration of `NewtonSLRA/1` algorithm

procedure `NewtonSLRA/1`($M \in E$, $(E_1, \dots, E_d)$ an orthonormal basis of E^0, $r \in \mathbb{Z}_{\geqslant 0}$)
  $(U, S, V) \leftarrow \mathrm{SVD}(M)$
  $S_r \leftarrow r \times r$ top-left sub-matrix of S
  $U_r \leftarrow$ first r columns of U
  $V_r \leftarrow$ first r columns of V
  $\widetilde{M} \leftarrow U_r \cdot S_r \cdot V_r^\intercal$
  $\widetilde{u_1}, \dots, \widetilde{u_{p-r}} \leftarrow$ last $p - r$ columns of U
  $\widetilde{v_1}, \dots, \widetilde{v_{q-r}} \leftarrow$ last $q - r$ columns of V
  for $i \in \{1, \dots, p-r\}, j \in \{1, \dots, q-r\}$ **do**
    $N_{(i-1)(q-r)+j} \leftarrow \widetilde{u_i} \cdot \widetilde{v_j}^\intercal$
  end for
  $A \leftarrow (\langle N_k, E_\ell \rangle)_{k,\ell} \in \mathrm{Mat}_{(p-r)(q-r),d}(\mathbb{R})$
  $b \leftarrow (\langle N_k, \widetilde{M} - M \rangle)_k \in \mathrm{Mat}_{(p-r)(q-r),1}(\mathbb{R})$
  return $M + \sum_{\ell=1}^{d} \left(A^\dagger \cdot b\right)_\ell E_\ell$, where $A^\dagger$ is the Moore-Penrose pseudo-inverse of A.
end procedure

Algorithm 6 one iteration of `NewtonSLRA/2` algorithm

procedure `NewtonSLRA/2`($M \in E$, $(E'_1, \dots, E'_{pq-d})$ an orthonormal basis of $(E^0)^\perp$, $r \in \mathbb{Z}_{\geqslant 0}$)
  $(U, S, V) \leftarrow \mathrm{SVD}(M)$
  $S_r \leftarrow r \times r$ top-left sub-matrix of S
  $U_r \leftarrow$ first r columns of U
  $V_r \leftarrow$ first r columns of V
  $\widetilde{M} \leftarrow U_r \cdot S_r \cdot V_r^\intercal$
  $u_1, \dots, u_p \leftarrow$ columns of U
  $v_1, \dots, v_q \leftarrow$ columns of V
  $(T_\ell)_{1 \leqslant \ell \leqslant (p+q-r)r} \leftarrow$ list of all matrices of the form $u_i \cdot v_j^\intercal$, where $i \leqslant r$ or $j \leqslant r$
  $A' \leftarrow (\langle E'_k, T_\ell \rangle)_{k,\ell} \in \mathrm{Mat}_{pq-d,(p+q-r)r}(\mathbb{R})$
  $b' \leftarrow (\langle E'_k, M - \widetilde{M} \rangle)_k \in \mathrm{Mat}_{pq-d,1}(\mathbb{R})$
  return $\widetilde{M} + \sum_{\ell=1}^{(p+q-r)r} \left(A'^\dagger \cdot b'\right)_\ell T_\ell$, where $A'^\dagger$ is the Moore-Penrose pseudo-inverse of A'.
end procedure

is due to the quadratic convergence of the algorithm, which is observed in practice. `GPGCD` and `uvGCD` have a linear rate of convergence, but they converge towards the optimal solution of the problem, whereas `NewtonSLRA` converges only to a "good" solution. We refer to [107, Tables 1, 2, 3] for numerical data about these expriments.

The second particular case of SLRA that we have studied is the problem of *matrix completion*. In this problem, the input is a matrix with some entries missing, and the goal is to fill these missing entries such that the matrix has some given rank. This problem has many applications, and it became particularly famous at the beginning of the 21th century due to its connections with the Netflix prize about algorithms for predicting user ratings. Matrix completion is a special case of SLRA where the affine space of matrices is the set of possible matrices when some entries are fixed and the other ones are unknowns. We compared our software with results obtained via convex optimization [100]. Our results show that there is a range of parameters where NewtonSLRA converges to a valid solution whereas methods from convex optimization fail. We refer to [107, Fig. 4] for numerical data about these expriments. However, a limitation of NewtonSLRA is that it is mostly efficient on matrices of small size. For large matrices, we compared NewtonSLRA with a Riemannian optimization method [116], and it appeared that NewtonSLRA is not competitive for such parameters.

Finally, we investigate Hankel matrices. These matrices occur in many applications, as they are for instance connected to tensors: Hankel matrices of rank r correspond to symmetric $(2 \times 2 \times \cdots \times 2)$-tensors of rank r. We compare the experimental results obtained with our implementation of `NewtonSLRA` with the Structured Total Least Norm (STLN) approach in [97]. Our experiments suggest that the convergence of `NewtonSLRA` is faster for larger amounts of noise in the input. We also run experiments that suggest that `NewtonSLRA` behaves quite well when there are outliers, i.e. when some entries of the matrix are much noisier than the rest of the input. We refer to [107, Tables 4 and 5] for numerical data about these expriments.

2.5.2 Algebraic and symbolic approach

The work presented in this section is joint work with Bernd Sturmfels and Giorgio Ottaviani and it has been published in [95].

In this section, we focus on symbolic approaches to solve SLRA. A first question is how to define the problem. Given some matrix known exactly (for instance via rational numbers), the goal is to find a nearby matrix in $E \cap \mathcal{D}_r$. One way to achieve this is to compute the minimizer of the Frobenius distance to M on the smooth locus of $E \cap \mathcal{D}_r$. Under genericity assumptions on the matrix M, this minimizer is well-defined and unique. This minimizer must be algebraic since it is a critical point of the distance function to M, hence it is defined by polynomial equations.

We can already notice that if we are not interested in the closest solution but on some nearby function, we may perturb the Euclidean distance function by adding weights.

For $\Lambda = (\lambda_{ij}) \in \mathrm{Mat}_{p,q}(\mathbb{R}_{>0})$, we let

$$\|M\|_\Lambda = \sqrt{\sum_{\substack{1 \leqslant i \leqslant p \\ 1 \leqslant j \leqslant q}} \lambda_{ij} M_{ij}^2}$$

denote the weighted Frobenius norm.

When Λ is the all-one matrix, then $\|\cdot\|_\Lambda$ is the classical Frobenius norm.

Given a matrix with real entries, exact unstructured weighted low-rank approximation refers to the problem of finding the closest matrices of bounded rank for this weighted Euclidean distance.

Problem 2.5.7 (Exact Weighted Low-Rank Approximation)**.** *Given a matrix with real entries* $M \in \mathrm{Mat}_{p,q}$*, a positive integer* $r \in \mathbb{Z}_{>0}$*, a weight matrix* $\Lambda \in \mathrm{Mat}_{p,q}$*, find the set of matrices* $M^\star$ *of rank* $\leqslant r$ *which minimize* $\|M - M^\star\|_\Lambda$*.*

This problem can be extended to its structured version by adding a linear or affine constraint on the solution space:

Problem 2.5.8 (Exact Weighted Structured Low-Rank Approximation)**.** *Given a matrix with real entries* $M \in \mathrm{Mat}_{p,q}$*, a positive integer* $r \in \mathbb{Z}_{>0}$*, a weight matrix* $\Lambda \in \mathrm{Mat}_{p,q}$*, and a linear or affine subspace of matrices* $E \subset \mathrm{Mat}_{p,q}(\mathbb{R})$ *find the set of matrices* $M^\star$ *in* E *of rank* $\leqslant r$ *which minimize* $\|M - M^\star\|_\Lambda$*.*

We note that the set of minimizers is a semi-algebraic set, since minimizing the norm is equivalent to minimizing its square, which is a polynomial function in the entries of $M^\star$.

Moreover, under genericity assumptions on M, the minimizer is actually unique, and its coordinates are algebraic: there are solutions of a multivariate polynomial system whose coefficients are polynomial in the entries of M and Λ. In particular, if the entries of M and Λ are algebraic numbers, then so are the entries of the minimizer $M^\star$ if it is unique.

This implies that the entries of $M^\star$ can be represented and computed exactly. An important numerical indicator of the difficulty of this computation is the algebraic degree of the problem. This indicator controls— roughly speaking —the maximal possible degree of the extension (with respect to the field where the coefficients of M, Λ belong) where the coordinates of $M^\star$ live.

The degree of this problem has huge importance for symbolic methods such as Gröbner bases, because it is related to the size of the output of these algorithms. Symbolic-numeric methods such as homotopy continuation are also governed by this degree since this controls the maximal number of paths that must be followed in order to find the solution.

Critical points and polynomial systems

Global minimization problems cannot easily be retranscribed algebraically. An easier notion which can be expressed algebraically is the notion of critical points of a map, which can be described as a rank deficiency condition on a Jacobian matrix; Hence we can obtain a polynomial system from the vanishing condition on some minors of a matrix.

Critical points of a map contain in particular local minimizers, and therefore also global minimizers. In fact, under some genericity assumptions on M, we expect the number of complex critical points to be finite, and their number corresponds to the algebraic degree that we are looking for. This algebraic degree actually contains some information about the "algebraic difficulty" of the problem, but this data is not easily accessible. Bounds on such degrees can sometimes be obtained via tools from intersection theory and algebraic geometry. However, there are also many cases where it is not clear how to estimate them. Therefore, estimating these degrees often imply a large part of numerical and algebraic experiments. However, such computations to compute this algebraic degree are often far from trivial, and this is where efficient computer algebra is needed.

As a motivating example, we studied a conjecture that was stated by William Rey in [101] and which claims that the number of local minimizers of a weighted (unstructured) low-rank approximation problem is bounded above by $\min(p, q)$. The following example (which was found via experiments with Gröbner bases) disproves this conjecture.

Example 2.5.9. *[95, Example 2] Set $r = 1$, $p = q = 3$, so we consider the problem of exact weighted low-rank approximation with the following data and weight matrices:*

$$M = \begin{bmatrix} -59 & 11 & 59 \\ 11 & 59 & -59 \\ 59 & -59 & 11 \end{bmatrix} \qquad \Lambda = \begin{bmatrix} 9 & 6 & 1 \\ 6 & 1 & 9 \\ 1 & 9 & 6 \end{bmatrix}.$$

The map $X \mapsto \|M - X\|_\Lambda$ restricted to rank 1 matrices has 39 complex critical points, which can be computed exactly using Gröbner bases. These 39 points are grouped in 6 maximal ideals over $\mathbb{Q}$, of respective degrees $1, 2, 6, 10, 10, 10$. Among those 39 critical points, 19 are real, 7 are local minimizers, and 3 of them are global minimizers. The three global minimizers are algebraic numbers of degree 10.

Since $7 > \min(3, 3)$, this example disproves Rey's conjecture.

However, things are not that simple in general as we encounter several computational problems:

- computing minors of a Jacobian matrix creates possibly high-degree multivariate polynomials, which might be difficult to handle;
- if the variety on which we are minimizing is singular, the singularizities will also make the minors of Jacobian matrix vanish. Therefore, if the singular locus has positive dimension, then this process creates high-dimensional components where our polynomial system vanishes, and which is a problem for computing the algebraic degree. A solution to this problem would be to restrict the minimization problem to the smooth locus; this can be achieved computationally via a process called *saturation*. However, this does not come freely: it implies adding new variables and high-degree multivariate polynomials to an already quite large system. In general this problem arises as soon as $r > 1$.

We also run computations on an applicative example from magnetic resonance imagery which was given to us by Thomas Schultz [108]. This case involves a noisy tensor of format $3 \times 3 \times 3 \times 3$. The goal is to find the closest rank-2 tensor. This problem can be model by finding the closest rank-2 matrix in a space of 6×6 catalecticant matrices, which is a subspace of the space of Hankel matrices. See [95, Example 7] for more details.

Our main objective is to design methods and formulas to compute this algebraic degree, using all the tools at our disposal. These tools involves computational tools from computer algebra, but also theoretical techniques from intersection theory. The general machinery to compute the degrees of critical points of Euclidean distance functions to varieties has been developed in [40], with the notion of *Euclidean Distance degree* (abbreviated "ED degree"). The EDdegree of an algebraic variety V is the number of complex critical points of the Euclidean distance function on the smooth locus of V to a generic point outside the variety. Our aim is to study what happens specifically when V is a linear section of the variety of low-rank matrices. A nice property of the ED degree is that it can be computed via the *polar degrees*, assuming that some condition is satisfied. The polar degrees are the same numerical objects as those used in Section 2.4.2.

Theorem 2.5.10. *[40, Thm. 5.4] If some transversality condition is satisfied, then the ED degree of an algebraic variety is the sum of its polar degrees.*

This theorem relates the ED degree of an algebraic variety to the numerical indicator of its complexity. This can also be expressed in terms of the degrees of the *Chern classes*, see [40, Thm. 5.8]. Note that Theorem 2.4.7 can be regarded as a variant of Theorem 2.5.10.

A nice property of generic linear sections of algebraic varieties is that we have a relationship between the polar classes: a generic linear section kills off the polar classes of highest dimensions. Another nice feature of the weighted Euclidean distance is that if we consider generic weights, then the transversality condition in Theorem 2.5.10 is satisfied. Therefore, in order to compute the degree of exact weighted structured low-rank approximation under genericity assumptions on Λ, E and M, we only need to compute the polar degrees of the varieties of low-rank matrices.

For $r = 1$ and $r = \min(p, q) - 1$, there is a closed formula for the polar degrees. For intermediate r, computing these values is more complicated and requires some Schubert calculus, see e.g. [95, Sec. 3].

Chapter 3

Research project

3.1 Point counting for general curves of large characteristic

In Section 1.2.2, we proved a complexity result for computing the zeta functions of hyperelliptic curves of large genus. A natural question is to ask whether this generalizes to all curves.

Problem 3.1.1. *Given a genus-g curve (let us say: planar, nodal, projective) over $\mathbb{F}_q$, does there exist a Schoof-like algorithm to compute its zeta function with complexity $O_g(\log(q)^{cg})$ for some $c > 0$? Here, the notation $O_g(\cdot)$ means that the constant hidden in the $O(\cdot)$ may depend on g.*

This would provide in particular an algorithm of complexity $O_d(\log(q)^{cd^2})$ for computing the number of solutions in $\mathbb{F}_q^2$ of an equation of the form $f(X, Y) = 0$, where $f \in \mathbb{F}_q[X, Y]$ is a polynomial of degree $\leqslant d$ with only nodal singularities. If this goal is reached, then this can probably be extended to curves with non-degenerate singularities by using the techniques that I plan to work on for this family of curves, see Section 3.4.

The main advantage of hyperelliptic curves is that elements in its Jacobian variety are easy to describe and to compute with, thanks to the Mumford coordinates and to Cantor's algorithm. For more general curves, the situation is more complicated: there is no easy and convenient representation of divisors, and the arithmetic must be done via Riemann-Roch space computations. Therefore, we cannot use Cantor's polynomials as in [6] to compute the ℓ-torsion. However, the recent developments of algorithmic tools might be sufficient to be able to build suitable polynomial systems encoding the ℓ-torsion, provided that the singularities of the curve are nice enough.

If this is too hard, there are some intermediate cases between hyperelliptic curves and general plane curves, for instance superelliptic curves or C_{ab} curves. For such families, the arithmetic in the Jacobian variety is easier to describe, and it may lead to interesting polynomial systems for modelling the ℓ-torsion.

As a side note, we can remark that the evolution of characteristic-p methods followed a similar path: they were first designed for hyperelliptic curves in odd characteristic [75], then they were generalized to the even characteristic [36], to superelliptic curves [55], to C_{ab} curves [35], then finally to general curves [115].

Another extension of the Schoof-like point-counting methods for large characteristic and large genus in Section 1.2.2 might be the computation of the zeta functions of higher-dimensional varieties defined over finite fields, although this would require much more technical machinery.

Another direction would be to try to optimize the constant c in the exponent of the complexity in the hyperelliptic setting, see Theorem 1.2.9. Indeed, it is a bit unsatisfactory that the proven bounds on the degrees of the coefficients of Cantor's polynomials are not tight. There are clear conjectures about these degrees which are stated in [6, Remark at the end of Sec. 6], and proving them would lead to improvements of the constant c.

In terms of practical computations, this project might interact with my plans of designing software to compute Riemann-Roch spaces for curves with non-degenerate singularities, see Section 3.4.

3.2 Isogenies of products of isogenous elliptic curves with complex multiplication

This part of the research project is already started via the PhD thesis of Julien Soumier started in October 2023 on this topic.

Isogenies of maximal abelian varieties

Isogenies of elliptic curves have received a lot of attention from cryptographers during the last decade, since they feature hard mathematical problems which can be used to design cryptographic primitives and which are resistant to quantum computers.

Until 2022, the flagship cryptosystem featuring isogenies of elliptic curves was SIDH [34], a fast quantum-resistant key-exchange protocol. However, in 2022, a groundbreaking discovery showed that it was possible to compute an isogeny between two elliptic curves from the knowledge of their action on a sufficiently large subgroup of points [29, 90, 102]. The key ingredient in these methods is to use isogenies of higher-dimensional abelian varieties.

A few papers later, there were clean complexity results and the isogeny toolbox got greatly expanded with new tools. These tools have since been used to design new cryptosystems, and this area of research is very active.

In the cryptographic world of isogenies, the family of elliptic curves which is mostly used are *supersingular elliptic curves* defined over $\mathbb{F}_{p^2}$. Moreover, for applications, it is often required that these curves have a lot of rational points. In fact, for technical reasons they are often *maximal*, i.e. their number of $\mathbb{F}_{p^2}$-rational points equals $(p+1)^2$.

The toolbox for isogenies that was designed after the attacks on SIDH uses results by Kani about *isogeny diamonds*, see e.g. [74]. These objects are constructed from products of abelian varieties. If we start with maximal elliptic curves, all the abelian varieties constructed by using Kani's method are maximal. Maximal abelian varieties have strong structural properties, which are probably not yet fully exploited in the cryptographic and algorithmic setting, see e.g. [72]. Current methods do not exploit that extra structure, since they rely mainly on the classical machinery of theta functions for representing principally polarized abelian varieties.

We plan on investigating if structural properties of maximal varieties have an impact on cryptographic properties.

Cryptographic applications

A consequence of the recent developments of the toolbox for isogenies is that researchers have tried to find new hard isogeny problems on which cryptosystems may be built. A related line of work is to try to patch broken systems. A typical example is the M-SIDH cryptosystem [53], proposed by Fouotsa, Moriya and Petit. This cryptosystem relies on the following modification

of the SIDH protocol: instead of publishing the action of a secret A-isogeny on the B-torsion, the action is published only up to a scalar multiplication by a square root of unity in $(\mathbb{Z}/B\mathbb{Z})^\times$. Surprisingly, this modification seems sufficient to avoid the recent attacks which use higher-dimensional isogeny computations. The hardness of this modified problem is still unclear, which calls for more research in this direction.

3.3 Drinfeld modules

The algorithmic framework for Drinfeld modules has been considerably developed during the last decades. Still, the range of applications of this toolbox is not completely clear. As Drinfeld modules are analogs to elliptic curves, it is quite natural to look for applications in *public-key cryptography*, since this is a typical application domain for elliptic curves (and more generally for algorithmic arithmetic geometry). However, all tries to build cryptosystems on Drinfeld modules have failed so far. This is due to the fact that the analogs of the hard problems for elliptic curves that are used as a foundation for cryptography are often computationally easy in the world of Drinfeld modules. For instance, computing isogenies between Drinfeld modules or endomorphism rings can be done in polynomial time, whereas no such fast algorithm in the world of abelian varieties is known.

Still, this is probably not the end of the story. There are many techniques in the science of communication that need less requirements than the most usual cryptographic building blocks such as encryption, signatures, etc. I plan to continue investigating whether there may be applications of Drinfeld modules for the construction of cryptographic protocols.

Applications of Drinfeld modules in cryptography and science of communication can also take indirect paths. For instance, works by Niederreiter and Xing in the 90s [94] showed how Drinfeld modules can be used to construct algebraic curves with many rational points; such objects are especially important in coding theory, for the construction of good algebraico-geometric codes. It would be interesting to revisit those methods by using the algorithmic tools that have been recently developed.

A third way Drinfeld modules can be used in cryptography could be by revisiting algorithms in which the Frobenius endomorphism plays a central role. For instance, a typical algorithm in this setting is the quasi-polynomial algorithm for computing discrete logarithm in the multiplicative group of a finite field of small characteristic [12]. It is an algorithm which is known to be difficult to analyze in a formal way (see e.g. [78]), and perhaps it can be reinterpreted with Drinfeld modules.

On top of applications to the science of communication, Drinfeld modules have applications for computing with univariate polynomials and algebraic curves, in the same way abelian varieties can be used for studying integers and number fields. A typical computational application of Drinfeld modules is the factorization of univariate polynomials [39]. I am planning to investigate if there are more such applications of Drinfeld modules in computer algebra.

3.4 Plane curves with non-degenerate singularities

In this section, I describe research directions which are continuations of the results on the computational aspects of Riemann-Roch spaces in Section 1.4. The general context is computational birational geometry for algebraic curves.

The main observation that leads to the research directions in this section is the schism in the complexity of computing Riemann-Roch spaces in terms of the structure of the singularities

of the input plane curve. The general motto is that when we understand the structure of the singularities, then we can compute efficiently objects related to the birational geometry of the curve. Riemann-Roch spaces are typical objects where the computational difficulty is strongly related to the way we understand and deal with the singularities. The other classical object that falls in this category is the computation of integral bases, whose formal definition is given below.

Definition 3.4.1. *Let $A \hookrightarrow B$ be an extension of Noetherian domains. The integral closure $\overline{A}$ of A in B is the set $\{b \in B \mid \exists f \in A[X]$ monic s.t. $f(b) = 0\}$. There are natural addition and multiplicative laws on $\overline{A}$, and hence $\overline{A}$ is a domain. If A is principal and $B = A[Y]/g(y)$ for a monic polynomial $g \in A[Y]$, then $\overline{A}$ is a free A-module of rank* $\deg(g)$.

A typical instanciation occurs when A is either a polynomial ring $A = k[X]$ or when A is the ring of integers $A = \mathbb{Z}$. In this section, we focus on the former case: $A = k[X]$.

Problem 3.4.2. *(Computation of integral bases) Given a monic polynomial $f \in k[X][Y]$, compute a $k[X]$-basis of the integral closure of $k[X,Y]/(f)$ with respect to the field extension $k(X) \hookrightarrow \mathrm{Frac}(k[X,Y]/(f))$.*

This problem has a long and rich history. Perhaps one of its first appearance from the viewpoint of modern computer algebra is in the Ph.D. thesis of Trager [114], in the context of algorithms for integrating algebraic functions. The problem of computing integral bases also plays an important role in the arithmetic algorithms for computing Riemann-Roch spaces designed in Hess' Ph.D. thesis [67]. Several improvements on this problem were obtained during the last decade, see e.g. [1] and references therein.

The problem of computing integral bases is strongly linked with the computational analysis of singularities (in fact the problem is trivial for non-singular curves), and we might even see it as a computational way of desingularizing curves.

Local analysis and Newton polygon

Plane curves are standard objects in computer algebra. They arise as soon as we consider pairs of field elements $(x, y) \in k^2$ which satisfy an algebraic relation given by a bivariate polynomial $Q \in k[X,Y]$. When we study such objects, some obstructions in the analysis arise from the *singularities* of the curve, which can be thought of as points where the some information about the curve is locally compressed. Studying what happens at the singularities is a key ingredient of algorithms that give information about the curve.

The formal way of describing a singularity of an affine plane curve described by a ring R of Krull dimension 1 is a maximal ideal $\mathfrak{m} \subset k[X,Y]$ such that the local ring $R_{\mathfrak{m}}$ is not a discrete valuation ring. Said otherwise, the vector space $\mathfrak{m}/\mathfrak{m}^2$ has dimension more than 1. One way to obtain a computational handle on singularities is to consider two functions α, β which generate $\mathfrak{m}$ as an R-module[2]. By Cohen's structure theorem [43, Thm. 7.7], there is an isomorphism between the completion $\widehat{k[X,Y]_{\mathfrak{m}}}$ and the power series ring $\kappa[[S,T]]$, where the isomorphism sends α to S and β to T. This isomorphism is not unique, however the Newton polygon of the image of the polynomial q defining the curve is uniquely defined since all isomorphisms give the same Newton polygon.

As a side note, we can notice that if the ideal $\mathfrak{m}$ encodes a rational closed point with affine coordinates (a, b), then $\alpha = X - a$, $\beta = Y - b$ generates the local ring at $\mathfrak{m}$.

[2] more generally we can consider a pair (α, β) which forms a *regular system of parameters* of the local ring.

The Newton polygon encodes a lot of information about the singularities. The notion of *non-degenerate singularity* can be read off from the Newton polygon: it correspond to cases where the restriction to the faces of the polygon provide separable univariate Laurent polynomials. This is formalized in the following definition:

Definition 3.4.3. *Let α, β be generators of $\mathfrak{m} \subset k[X,Y]$. There exists an isomorphism ϕ between the completion $\widehat{k[X,Y]_{\mathfrak{m}}}$ and the power series ring $\kappa[[S,T]]$ sending α to S and β to T. This isomorphism is unique up to composition by the Galois group* $\mathrm{Gal}(\kappa/k)$.

The Newton polygon *of an element $f \in k[X,Y]_{\mathfrak{m}}$ w.r.t. (α, β) is the Newton polygon of its image in the power series ring, namely it is the lower convex hull of the exponent vectors of the monomials occurring in $\phi(f)$ with nonzero coefficient. For each rational number $q = a/b \in \mathbb{Q}$ (with a, b positive coprime integers), define the* slope polynomial $S_f^q \in \kappa[T^{\pm 1}]$ *as a Laurent polynomial corresponding to the coefficients of the monomials in $\phi(f)$ whose exponent vectors minimize the linear functional $(u,v) \mapsto au + bv$. The polynomial $S_f^{(q)} \in \kappa[T^{\pm 1}]$ is uniquely defined up to the natural action of* $\mathrm{Gal}(\kappa/k)$ *and multiplication by Laurent monomials.*

The singularity is called non-degenerate *(w.r.t. α, β) if $S_f^{(q)}$ is separable for all $q \in \mathbb{Q}$.*

It can be noticed that $S_f^{(q)} = 1$ for all but finitely-many q: $S_f^{(q)} \neq 1$ if and only if there is a 1-dimensional face of the Newton polytope with slope $-q^{-1}$. *Ordinary singularities* are a special case of non-degenerate singularities: they correspond to the case where the only non trivial slope polynomial is $S_f^{(1)}$ and it is separable. Therefore, non-degenerate are more general than ordinary singularities, but we still get a lot of information on the structure of the singularity from its Newton polygon. For instance, there is a old theorem due to Baker which quantifies the negative contribution to the genus of a non-degenerate singularity from its Newton polygon [10].

Gorenstein's theory of adjoints

The classical way to deal with ordinary singularities relies on the notion of *adjoint curves*. This method goes back to Brill and Noether, and is for instance described in Severi's monograph [109]. Brill and Noether's adjoint theory has a generalization due to Gorenstein [62]. The core idea of this generalization is to notice that the problem of computing in the presence of a singularity comes from the gap between the localized coordinate ring and its normalization. This normalization is to intersection of all discrete valuation rings centered at the singularity. A computational tool to have access of this normalization is the *conductor* between these two rings.

Gorenstein's conductor at the singularity is defined as follows:

$$\mathfrak{C}_{\mathfrak{m}} = \{f \in k[C_0]_{\mathfrak{m}} : f \cdot \overline{k[C_0]_{\mathfrak{m}}} \subset k[C_0]_{\mathfrak{m}}\}.$$

Therefore $\mathfrak{C}_{\mathfrak{m}}$ is an ideal in the localized coordinate ring $k[C_0]_{\mathfrak{m}}$ and knowing it gives a lot of information about the singularity. For ordinary singularities, there is a combinatorial description of this conductor, and we can hope that such nice combinatorial formulas could also be found for non-degenerate singularities.

Explicit computations of integral bases and Riemann-Roch spaces

There are several algorithms for the computation of integral bases in function fields. These algorithms work by computing local bases at each singularity, using the most general tools available (e.g. Puiseux series) to perform the local computations. I believe that in the case of

non-degenerate singularities, these computations can probably be made more efficient in theory and in practice, by using Gorenstein's theory of ajoints.

It is worth noticing that most of the singularities that we encounter in applications are non-degenerate, so such algorithmic improvements might have concrete impacts on applications.

I believe that Gorenstein's theory of adjoints could be made explicit for curves for which we know a system of parameters that make each singularity non-degenerate. Hopefully, this will extend the class of curves for which we have efficient algorithms for the computation of Riemann-Roch spaces and integral bases. For Riemann-Roch spaces, for instance, algorithms with subquadratic complexities are for now restricted to curves with ordinary singularities [4], and we may hope to extend them to curves with non-degenerate singularities.

Hopefully this can also be made very efficient in practice and lead to practical software. An important computational tool in this setting are fast algorithms for polynomial matrices. This is a crucial tool for achieving subquadratic complexities for the computation of Riemann-Roch spaces. The final objective of this work line would be to provide the computer algebra community with fast software computing integral bases and Riemann-Roch bases. In particular, such a software might build upon software for fast methods for polynomial matrices which is currently developed by Vincent Neiger and Éric Schost, who are two leading experts in this domain, see `https://github.com/vneiger/pml`.

3.5 Gröbner basis engineering and technology

In this section, I present some selected topics for Gröbner bases and polynomial systems, which are related to some difficulties that I encountered in the study of polynomial systems coming from applications. I believe that these directions could lead to the development of interesting practical computational tools.

Practical implementation of Gröbner bases for applications

Early termination. In some applications, most of the time required by Gröbner bases computations is actually spent reducing high-degree S-polynomials to zero. In other words, at some point, the computation is finished but the algorithm cannot prove that the output is correct since we do not have any proof that all these S-polynomials will reduce to zero. Such a phenomenon appeared for instance during the computation of the 7-torsion of genus-3 hyperelliptic curves with explicit real multiplication in [5]. It would be very interesting to have some ways to stop the computation and admit heuristically that the result is correct. Of course, in this situation we must be prepared and we should know what happens in the case where the result is actually incomplete. In many applicative situations (for instance in cryptography), often the computation of the Gröbner basis is only a step in a longer computation, and the final result can be checked *a posteriori*. A related idea is to investigate how to extract useful information from partial Gröbner bases.

Partial FGLM. The FGLM algorithm [51] is a computational technique to transform a 0-dimensional Gröbner basis for a monomial ordering into a Gröbner basis for another monomial ordering. It is a crucial step in the classical solving process for solving 0-dimensional systems with Gröbner bases. I plan to investigate if some variants of the FGLM algorithm can be designed when the input is not a full Gröbner basis but only a partial one, and how using such a process introduce parasitic solutions which can be filtered *a posteriori*.

Bestiary of polynomial systems. In order to test the robustness of Gröbner bases algorithms (and other methods), it is useful to have access to a large database of polynomial systems which exhibit some structures and some specificities. Such databases already exist, for instance the database designed by Dario Andrea Bini and Bernard Mourrain[3]. I believe that it would also be interesting to populate such databases with examples coming from arithmetic geometry and from cryptography. I would like to build my own database of examples, to complement the databases already existing with new families of structured polynomial systems.

TinyGB. During the last years, I have written a C++/NTL implementation of the F4 algorithm to compute Gröbner bases. The main aim of this implementation is to serve as a vehicle to test new algorithmic ideas for Gröbner bases computations. The algorithmic tools described in the previous paragraph may be implemented within this software, which is freely available at the URL `https://gitlab.inria.fr/pspaenle/tinygb`.

Gröbner bases and global sections of line bundles

This section presents a few theoretical views on Gröbner bases, which may be useful for extending Gröbner bases computations to practical situations where the classical degree-driven algorithms do not work well. The way classical Gröbner bases are usually presented often focuses strongly on the notion of monomial ordering, and on the fact that the initial monomial ideal is a degeneration that preserves a lot of the structure of the input ideal. While this is indeed an important topic for Gröbner bases, it often hides the fact that this property is not always needed for polynomial system solving. Behind this remark lies the fact that a lot of good properties of monomial orderings in the affine and projective setting do not extend well in other settings — for instance, when we want to perform computations in the Cox ring of a complete toric variety. There are other general symbolic methods for polynomial system solving which do not require the notion of monomial ordering: for instance, resultants focus mainly on linear algebra, and less on monomial orderings, and their objective is to compute determinants of well-suited linear maps.

I believe that it is quite interesting to understand how these linear maps for polynomial system solving — which arise in almost all methods for solving polynomial systems — can be interpreted geometrically. In fact, these linear spaces can be thought of spaces of functions which arise as global sections of coherent sheaves on complete varieties.

Given a coherent sheaf $\mathcal{F}$ on a (normal, integral, Noetherian) $\mathrm{Spec}(k)$-scheme X and for each class $\alpha \in \mathrm{Cl}(X)$, a useful information to compute is an explicit description of all line bundles $O(D)$ on X for Weil divisors D and bases of the corresponding linear spaces of global sections of the sheaves $O(D) \otimes \mathcal{F}$.

In the (weighted) projective case, when $\mathcal{F}$ is an ideal sheaf (which defines a closed subscheme) this data is actually produced by a Gröbner basis, which is a consequence of the following proposition:

Proposition 3.5.1. *Let $I \subset k[\mathbf{X}]$ be an ideal. A finite subset of $\{g_1, \dots, g_r\} \subset I$ is a* Gröbner basis *with respect to a monomial ordering $\prec$ if and only if for any $d \in \mathbb{Z}_{>0}$, there exists $d' \in \mathbb{Z}_{>0}$ such that the set $\sum_{i=1}^{r} g_i \cdot k[\mathbf{X}]_{\leqslant d' - \deg(g_i)}$ (with the convention that $k[\mathbf{X}]_{\leqslant m} = \{0\}$ for $m < 0$) contains an echelonized k-basis (with respect to $\prec$) of $\sum_{i=1}^{r} f_i \cdot k[\mathbf{X}]_{\leqslant d - \deg(f_i)}$.*

In the case of a complete toric variety X built from a polyhedral fan (see Section 2.2.4), the analog of degrees in the classical setting are divisor classes, and for any torus-invariant divisor D

[3] https://www-sop.inria.fr/saga/POL/

on X, we can build a coherent sheaf $O(D)$ whose global sections $\Gamma(O(D))$ are finite-dimensional vector spaces generated by monomials. This is the analog of the linear spaces of homogeneous polynomials of some given degree in the classical case, and therefore Gröbner bases should deal with the problem of finding a convenient description of these linear spaces. More precisely, if $\mathcal{I}$ is an ideal sheaf on X defining a closed subscheme, we want to compute a data structure which provide us with a way to compute with the vector spaces $\Gamma(O(D))/\Gamma(\mathcal{I} \otimes O(D))$. Although there are infinitely-many divisor classes on X, echelonized bases for a finite subset of them is sufficient to describe all of them: this is the analog of the finiteness of Gröbner bases in the classical case.

Finding which finite subsets provide sufficient information is related to the classical notion of regularity, and from a theoretical perspective this is probably driven by local cohomology. For 0-dimensional subschemes, the situation is simpler as we may not need the full information: usually it is sufficient to compute the characteristic polynomial of a multiplication map by a function f, as the eigenvalues contain the information of the values of f at the points of the closed subscheme.

Let us now see how this translates algorithmically. Most classical Gröbner basis algorithms proceed by considering increasing degrees $d \in \mathbb{Z}_{\geqslant 0}$, and by computing an echelonized basis of polynomials of degree at most d. In a more general setting of a complete variety X, we do not have anymore a canonical ordering on the divisor classes. A general algorithm for Gröbner bases would proceed as follows:

- We start by having as input a description of a complete variety X, a description of an ideal sheaf $\mathcal{I}$ on X, and a description of the class group of X.
- Then we would compute echelonized bases for $\Gamma(\mathcal{I} \otimes O(D))$ for a finite family of divisors D with distinct classes, potentially reusing previous computations: if $D_1 + D_2 = D_3$, then the image of the echelonized basis of $\Gamma(\mathcal{I} \otimes O(D_1))$ via the map $\Gamma(\mathcal{I} \otimes O(D_1)) \otimes \Gamma(O(D_2)) \to \Gamma(\mathcal{I} \otimes O(D_3))$ could provide a partial computation of an echelonized basis for $\Gamma(\mathcal{I} \otimes O(D_3))$.

This pattern would probably be particularly suited in situations where X is related to a multiplicative structure and the global sections of $O(D)$ are generated by monomials, which is the case for instance over toric varieties built from complete polyhedral fans.

Gröbner bases over toric varieties built from complete polyhedral fans

Seeing polynomials as global sections of sheaves might seem theoretical, but this language is very convenient for generalizations on other complete varieties than projective spaces. My main focus in this area in the future is to continue the investigation of Gröbner basis algorithms for sparse systems from the point of view of toric compactifications. Our first step in this direction was to understand how dimension and intersection behaves for sparse systems over these toric compactifications, and we got a complete criterion in [15], see Section 2.2.4.

A nice feature of toric varieties built from complete polyhedral fans is that we have a combinatorial description of the divisors and the associated sheaves. The next step is to understand how the classical Gröbner basis approach can be adapted to this case.

At the end, the main objective would be to integrate this work in the software tinyGB, in order to provide a Gröbner bases software based on efficient linear algebra which uses toric compactifications in order to speed-up computations for systems with monomial structures.

Bibliography

[1] S. Abelard. On the complexity of computing integral bases of function fields. In *International Workshop on Computer Algebra in Scientific Computing*, pages 42–62. Springer, 2020.

[2] S. Abelard, E. Berardini, A. Couvreur, and G. Lecerf. Computing Riemann–Roch spaces via puiseux expansions. *Journal of Complexity*, 73:101666, 2022.

[3] S. Abelard, A. Couvreur, and G. Lecerf. Sub-quadratic time for Riemann-Roch spaces: case of smooth divisors over nodal plane projective curves. In *Proceedings of the 45th International Symposium on Symbolic and Algebraic Computation*, pages 14–21, 2020.

[4] S. Abelard, A. Couvreur, and G. Lecerf. Efficient computation of Riemann–Roch spaces for plane curves with ordinary singularities. *Applicable Algebra in Engineering, Communication and Computing*, pages 1–66, 2022.

[5] S. Abelard, P. Gaudry, and P.-J. Spaenlehauer. Counting points on genus-3 hyperelliptic curves with explicit real multiplication. *The Open Book Series*, 2(1):1–19, 2019.

[6] S. Abelard, P. Gaudry, and P.-J. Spaenlehauer. Improved complexity bounds for counting points on hyperelliptic curves. *Foundations of Computational Mathematics*, 19:591–621, 2019.

[7] L. M. Adleman and M.-D. Huang. Counting points on curves and abelian varieties over finite fields. *Journal of Symbolic Computation*, 32(3):171–189, 2001.

[8] E. Arbarello, M. Cornalba, P. Griffiths, and J. D. Harris. *Geometry of Algebraic Curves: Volume I.* Springer-Verlag, 1985.

[9] S. Bae and J. K. Koo. On the singular Drinfeld modules of rank 2. *Mathematische Zeitschrift*, 210:267–275, 1992.

[10] H. F. Baker. Examples of the application of Newton's polygon to the theory of singular points of algebraic functions. *Transactions of the Cambridge Philosophical Society*, 15:403–450, 1894.

[11] B. Bank, M. Giusti, J. Heintz, G. Lecerf, G. Matera, and P. Solernó. Degeneracy loci and polynomial equation solving. *Foundations of Computational Mathematics*, 15(1):159–184, 2015.

[12] R. Barbulescu, P. Gaudry, A. Joux, and E. Thomé. A heuristic quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques – EUROCRYPT 2014*, pages 1–16. Springer, 2014.

[13] M. Bardet, M. Bros, D. Cabarcas, P. Gaborit, R. Perlner, D. Smith-Tone, J.-P. Tillich, and J. Verbel. Improvements of algebraic attacks for solving the rank decoding and MinRank problems. In *Advances in Cryptology–ASIACRYPT 2020: 26th International Conference on the Theory and Application of Cryptology and Information Security*, pages 507–536. Springer, 2020.

[14] D. Bayer and M. Stillman. A criterion for detecting m-regularity. *Inventiones mathematicae*, 87(1):1–11, 1987.

[15] M. Bender and P.-J. Spaenlehauer. Dimension results for extremal-generic polynomial systems over complete toric varieties. *Journal of Algebra*, 646:156–182, 2024.

[16] M. Bender and S. Telen. Toric eigenvalue methods for solving sparse polynomial systems. *Mathematics of Computation*, 91(337):2397–2429, 2022.

[17] M. R. Bender. *Algorithms for sparse polynomial systems: Gröbner bases and resultants*. PhD thesis, Sorbonne université, 2019.

[18] M. R. Bender, J.-C. Faugère, and E. Tsigaridas. Towards mixed Gröbner basis algorithms: the multihomogeneous and sparse case. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, pages 71–78, 2018.

[19] E. Berardini, A. Couvreur, and G. Lecerf. A proof of the Brill-Noether method from scratch. *ACM Communications in Computer Algebra*, 57(4):200–229, 2024.

[20] D. N. Bernshtein. The number of roots of a system of equations. *Functional Analysis and its applications*, 9(3):183–185, 1975.

[21] F. Bihan, J. M. Rojast, and F. Sottile. On the sharpness of fewnomial bounds and the number of components of fewnomial hypersurfaces. In *Algorithms in algebraic geometry*, pages 15–20. Springer, 2008.

[22] F. Bihan, F. Santos, and P.-J. Spaenlehauer. A polyhedral method for sparse systems with many positive solutions. *SIAM Journal on Applied Algebra and Geometry*, 2(4):620–645, 2018.

[23] F. Bihan and F. Sottile. New fewnomial upper bounds from Gale dual polynomial systems. *Moscow mathematical journal*, 7(3), 2007.

[24] S. R. Blackburn, C. Cid, and S. D. Galbraith. Cryptanalysis of a cryptosystem based on Drinfeld modules. *Cryptology ePrint Archive*, 2003.

[25] B. Buchberger. Bruno Buchberger's PhD thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal. *Journal of symbolic computation*, 41(3-4):475–511, 2006.

[26] N. Budur, M. Casanellas, and E. Gorla. Hilbert functions of irreducible arithmetically Gorenstein schemes. *Journal of Algebra*, 272(1):292–310, 2004.

[27] D. G. Cantor. Computing in the Jacobian of a hyperelliptic curve. *Mathematics of computation*, 48(177):95–101, 1987.

[28] D. G. Cantor. On the analogue of the division polynomials for hyperelliptic curves. *Journal für die reine und angewandte Mathematik*, 447, 1994.

[29] W. Castryck and T. Decru. An efficient key recovery attack on SIDH. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques – EUROCRYPT 2023*, pages 423–447. Springer, 2023.

[30] H. Cohen, G. Frey, R. Avanzi, C. Doche, T. Lange, K. Nguyen, and F. Vercauteren. *Handbook of elliptic and hyperelliptic curve cryptography*. CRC press, 2005.

[31] J.-M. Couveignes. Hard homogeneous spaces. *Cryptology ePrint Archive*, 2006.

[32] D. A. Cox, J. B. Little, and H. K. Schenck. *Toric Varieties*. AMS, 2011.

[33] P. Dartois, A. Leroux, D. Robert, and B. Wesolowski. SQISignHD: new dimensions in cryptography. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques – EUROCRYPT 2024*, pages 3–32. Springer, 2024.

[34] L. De Feo, D. Jao, and J. Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *Journal of Mathematical Cryptology*, 8(3):209–247, 2014.

[35] J. Denef and F. Vercauteren. Counting points on Cab curves using Monsky-Washnitzer cohomology. *Finite Fields and Their Applications*, 12(1):78–102, 2006.

[36] J. Denef and F. Vercauteren. An extension of Kedlaya's algorithm to hyperelliptic curves in characteristic 2. *Journal of cryptology*, 19(1):1–25, 2006.

[37] A. Dickenstein and I. Z. Emiris. Multihomogeneous resultant formulae by means of complexes. *Journal of Symbolic Computation*, 36(3-4):317–342, 2003.

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

[39] J. Doliskani, A. K. Narayanan, and É. Schost. Drinfeld modules with complex multiplication, Hasse invariants and factoring polynomials over finite fields. *Journal of Symbolic Computation*, 105:199–213, 2021.

[40] J. Draisma, E. Horobeţ, G. Ottaviani, B. Sturmfels, and R. R. Thomas. The Euclidean distance degree of an algebraic variety. *Foundations of computational mathematics*, 16:99–149, 2016.

[41] V. G. Drinfel'd. Elliptic modules. *Mathematics of the USSR-Sbornik*, 23(4):561, 1974.

[42] D. Eisenbud. *The geometry of syzygies*. Springer, 2005.

[43] D. Eisenbud. *Commutative algebra: with a view toward algebraic geometry*. Springer, 2013.

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

[45] B. El Hilany. Constructing polynomial systems with many positive solutions using tropical geometry. *Revista Matemática Complutense*, 31:525–544, 2018.

[46] J.-C. Faugère, M. S. El Din, and T. Verron. On the complexity of computing Gröbner bases for weighted homogeneous systems. *Journal of Symbolic Computation*, 76:107–141, 2016.

[47] J.-C. Faugère, M. Safey El Din, and T. Verron. On the complexity of computing Gröbner bases for quasi-homogeneous systems. In *Proceedings of the 38th International Symposium on Symbolic and Algebraic Computation*, pages 189–196, 2013.

[48] J.-C. Faugère, P.-J. Spaenlehauer, and J. Svartz. Computing small certificates of inconsistency of quadratic fewnomial systems. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, pages 223–230, 2016.

[49] J.-C. Faugère. A new efficient algorithm for computing Gröbner bases (f4). *Journal of pure and applied algebra*, 139(1-3):61–88, 1999.

[50] J. C. Faugère. A new efficient algorithm for computing Gröbner bases without reduction to zero (F5). In *Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation*, pages 75–83, 2002.

[51] J.-C. Faugère, P. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *Journal of Symbolic Computation*, 16(4):329–344, 1993.

[52] J.-C. Faugère, P.-J. Spaenlehauer, and J. Svartz. Sparse Gröbner bases: the unmixed case. In *Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, pages 178–185, 2014.

[53] T. B. Fouotsa, T. Moriya, and C. Petit. M-SIDH and MD-SIDH: countering SIDH attacks by masking information. In *EUROCRYPT 2023*, pages 282–309. Springer, 2023.

[54] W. Fulton. Algebraic curves, 2008.

[55] P. Gaudry and N. Gürel. An extension of Kedlaya's point-counting algorithm to superelliptic curves. In *Advances in Cryptology-ASIACRYPT 2001: 7th International Conference on the Theory and Application of Cryptology and Information Security*, pages 480–494. Springer, 2001.

[56] P. Gaudry, D. Kohel, and B. Smith. Counting points on genus 2 curves with real multiplication. In *International Conference on the Theory and Application of Cryptology and Information Security – ASIACRYPT 2011*, pages 504–519. Springer, 2011.

[57] P. Gaudry and É. Schost. A low-memory parallel version of Matsuo, Chao, and Tsujii's algorithm. In *International Algorithmic Number Theory Symposium*, pages 208–222. Springer, 2004.

[58] C. F. Gauss. Disquisitiones Arithmeticae, 1801.

[59] E.-U. Gekeler. On finite Drinfeld modules. *Journal of Algebra*, 141(1):187–203, 1989.

[60] M. Giusti, G. Lecerf, and B. Salvy. A Gröbner free alternative for polynomial system solving. *Journal of complexity*, 17(1):154–211, 2001.

[61] V. D. Goppa. Algebraico-geometric codes. *Izvestiya Rossiiskoi Akademii Nauk. Seriya Matematicheskaya*, 46(4):762–781, 1982.

[62] D. Gorenstein. An arithmetic theory of adjoint plane curves. *Transactions of the American Mathematical Society*, 72(3):414–436, 1952.

[63] D. Goss. *Basic structures of function field arithmetic*. Springer Science & Business Media, 2012.

[64] G. Haché. Computation in algebraic function fields for effective construction of algebraic-geometric codes. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes: 11th International Symposium*, pages 262–278. Springer, 1995.

[65] R. Hartshorne. *Algebraic geometry*. Springer, 2013.

[66] D. R. Hayes. A brief introduction to Drinfeld modules. *The arithmetic of function fields*, 2:1–32, 1992.

[67] F. Hess. Computing Riemann–Roch spaces in algebraic function fields and related topics. *Journal of Symbolic Computation*, 33(4):425–445, 2002.

[68] E. W. Howe and K. E. Lauter. Improved upper bounds for the number of points on curves over finite fields. *Annales de l'Institut Fourier*, 53(6):1677–1737, 2003.

[69] M.-D. Huang and D. Ierardi. Efficient algorithms for the Riemann-Roch problem and for addition in the Jacobian of a curve. *Journal of Symbolic Computation*, 18(6):519–539, 1994.

[70] B. Huber and B. Sturmfels. A polyhedral method for solving sparse polynomial systems. *Mathematics of computation*, 64(212):1541–1555, 1995.

[71] G. Jeronimo, G. Matera, P. Solerno, and A. Waissbein. Deformation techniques for sparse systems. *Foundations of Computational Mathematics*, 9(1):1–50, 2009.

[72] B. W. Jordan, A. G. Keeton, B. Poonen, E. M. Rains, N. Shepherd-Barron, and J. T. Tate. Abelian varieties isogenous to a power of an elliptic curve. *Compositio Mathematica*, 154(5):934–959, 2018.

[73] A. Joux and A. K. Narayanan. Drinfeld modules may not be for isogeny based cryptography. *Cryptology ePrint Archive*, 2019.

[74] E. Kani. The number of curves of genus two with elliptic differentials. *Journal für die reine und angewandte Mathematik*, 485:93–121, 1997.

[75] K. S. Kedlaya. Counting points on hyperelliptic curves using Monsky-Washnitzer cohomology. *Journal of the Ramanujan Mathematical Society*, 16:323–338, 2001.

[76] A. G. Khovanskii. A class of systems of transcendental equations. *Doklady Akademii Nauk*, 255(4):804–807, 1980.

[77] K. Khuri-Makdisi. Asymptotically fast group operations on Jacobians of general curves. *Mathematics of Computation*, 76(260):2213–2239, 2007.

[78] T. Kleinjung and B. Wesolowski. Discrete logarithms in quasi-polynomial time in finite fields of fixed characteristic. *Journal of the American Mathematical Society*, 35(2):581–624, 2022.

[79] A. Knutson and E. Miller. Gröbner geometry of Schubert polynomials. *Annals of Mathematics*, pages 1245–1318, 2005.

[80] N. Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177):203–209, 1987.

[81] D. Kohel and B. Smith. Efficiently computable endomorphisms for hyperelliptic curves. In *Algorithmic Number Theory: 7th International Symposium, ANTS-VII*, pages 495–509. Springer, 2006.

[82] D. Lazard. Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In *European Conference on Computer Algebra*, pages 146–156. Springer, 1983.

[83] D. Lazard and F. Rouillier. Solving parametric polynomial systems. *Journal of Symbolic Computation*, 42(6):636–667, 2007.

[84] A. Le Gluher. *Symbolic Computation and Complexity Analyses for Number Theory and Cryptography*. PhD thesis, Université de Lorraine, 2021.

[85] A. Le Gluher and P.-J. Spaenlehauer. A fast randomized geometric algorithm for computing Riemann-Roch spaces. *Mathematics of Computation*, 89(325):2399–2433, 2020.

[86] A. Le Gluher, P.-J. Spaenlehauer, and E. Thomé. Refined analysis of the asymptotic complexity of the Number Field Sieve. *Mathematical Cryptology*, 1(1):71–88, 2021.

[87] A. Leudière and P.-J. Spaenlehauer. Computing a group action from the class field theory of imaginary hyperelliptic function fields. *Journal of Symbolic Computation*, page 102311, 2024.

[88] A. Leudière and P.-J. Spaenlehauer. Hard homogeneous spaces from the class field theory of imaginary hyperelliptic function fields. Cryptology ePrint Archive, Paper 2022/349, 2022. `https://eprint.iacr.org/2022/349`.

[89] D. Lorenzini. *An Invitation to Arithmetic Geometry*. American Mathematical Society, 1996.

[90] L. Maino, C. Martindale, L. Panny, G. Pope, and B. Wesolowski. A direct key recovery attack on SIDH. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques – EUROCRYPT 2023*, pages 448–471. Springer, 2023.

[91] E. Miller and B. Sturmfels. *Combinatorial commutative algebra*, volume 227. Springer Science & Business Media, 2005.

[92] V. S. Miller. Use of elliptic curves in cryptography. In *Conference on the theory and application of cryptographic techniques*, pages 417–426. Springer, 1985.

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

[94] H. Niederreiter and C. Xing. Drinfeld modules of rank 1 and algebraic curves with many rational points. ii. *Acta Arithmetica*, 81(1):81–100, 1997.

[95] G. Ottaviani, P.-J. Spaenlehauer, and B. Sturmfels. Exact solutions in structured low-rank approximation. *SIAM Journal on Matrix Analysis and Applications*, 35(4):1521–1542, 2014.

[96] M. Papikian. *Drinfeld Modules*. Springer, 2023.

[97] H. Park, L. Zhang, and J. B. Rosen. Low rank approximation of a Hankel matrix by structured total least norm. *BIT Numerical Mathematics*, 39:757–779, 1999.

[98] J. Pila. Frobenius maps of abelian varieties and finding roots of unity in finite fields. *Mathematics of Computation*, 55(192):745–763, 1990.

[99] B. Poonen. Introduction to Drinfeld modules. In *Contemporary Mathematics*, volume 779, pages 167–186. AMS, 2022.

[100] B. Recht, W. Xu, and B. Hassibi. Necessary and sufficient conditions for success of the nuclear norm heuristic for rank minimization. In *47th IEEE Conference on Decision and Control*, pages 3065–3070. IEEE, 2008.

[101] W. Rey. On weighted low-rank approximation. arXiv:1302.0360, 2013.

[102] D. Robert. Breaking SIDH in polynomial time. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques – EUROCRYPT 2023*, pages 472–503. Springer, 2023.

[103] A. Rostovtsev and A. Stolbunov. Public-key cryptosystem based on isogenies. *Cryptology ePrint Archive*, 2006.

[104] M. Safey El Din and P.-J. Spaenlehauer. Critical point computations on smooth varieties: degree and complexity bounds. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, pages 183–190, 2016.

[105] T. Scanlon. Public key cryptosystems based on Drinfeld modules are insecure. *Journal of cryptology*, 14:225–230, 2001.

[106] R. Schoof. Elliptic curves over finite fields and the computation of square roots mod p. *Mathematics of computation*, 44(170):483–494, 1985.

[107] É. Schost and P.-J. Spaenlehauer. A quadratically convergent algorithm for structured low-rank approximation. *Foundations of Computational Mathematics*, 16(2):457–492, 2016.

[108] T. Schultz, A. Fuster, A. Ghosh, R. Deriche, L. Florack, and L.-H. Lim. Higher-order tensors in diffusion imaging. In *Visualization and Processing of Tensors and Higher Order Descriptors for Multi-Valued Data*, pages 129–161. Springer, 2014.

[109] F. Severi. *Vorlesungen über Algebraische Geometrie*. Springer-Verlag, 1921.

[110] P.-J. Spaenlehauer. *Solving multi-homogeneous and determinantal systems: algorithms, complexity, applications.* PhD thesis, Université Pierre et Marie Curie (Univ. Paris 6), 2012.

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

[112] A. Sutherland. A generic approach to searching for Jacobians. *Mathematics of Computation*, 78(265):485–507, 2009.

[113] W. Tautz, J. Top, and A. Verberkmoes. Explicit hyperelliptic curves with real multiplication and permutation polynomials. *Canadian Journal of Mathematics*, 43(5):1055–1064, 1991.

[114] B. M. Trager. *Integration of algebraic functions.* PhD thesis, MIT, 1984.

[115] J. Tuitman. Counting points on curves using a map to $\mathbb{P}^1$. *Mathematics of Computation*, 85(298):961–981, 2016.

[116] B. Vandereycken. Low-rank matrix completion by Riemannian optimization. *SIAM Journal on Optimization*, 23(2):1214–1236, 2013.

[117] T. Verron. *Regularisation of Gröbner basis computations for weighted and determinantal systems, and application to medical imagery.* PhD thesis, Université Pierre et Marie Curie-Paris VI, 2016.

[118] O. Viro. Patchworking real algebraic varieties. *arXiv preprint math/0611382*, 2006.

[119] E. J. Volcheck. Computing in the Jacobian of a plane algebraic curve. In *International Algorithmic Number Theory Symposium*, pages 221–233. Springer, 1994.

[120] A. Weil. Numbers of solutions of equations in finite fields. *Bull. Amer. Math. Soc.*, 55:497–508, 1949.

[121] B. Wesolowski. Computing isogenies between finite Drinfeld modules. *IACR Communications in Cryptology*, 1(1), 2024.

[122] V. V. Williams. Multiplying matrices faster than Coppersmith-Winograd. In *Proceedings of the 44th annual ACM symposium on theory of computing*, pages 887–898, 2012.