The point decomposition problem in Jacobian varieties

Alexandre Wallet

SORBONNE UNIVERSITÉS

École Doctorale Informatique, Télécommunications et Électronique (Paris)

THÈSE DE DOCTORAT

pour obtenir le grade de

DOCTEUR EN SCIENCES
DE L'UNIVERSITÉ PIERRE ET MARIE CURIE

Spécialité Informatique

Le problème de décomposition de points dans les Variétés Jacobiennes

présentée et soutenue publiquement par

Alexandre Wallet

le 26 Novembre 2016

Directeur : **Jean Charles Faugère**
Encadrante : **Vanessa Vitse**

Jury

Jean-Charles Faugère,	Directeur de recherche, INRIA	Directeur
Vanessa Vitse,	Maître de conférence, UJF-Grenoble	Encadrante
David Lubicz,	Chercheur associé DGA-MI	Rapporteur
François Morain,	Professeur, École Polytechnique	Rapporteur
Stef Graillat,	Professeur, UPMC	Examinateur
Mohab Safey-El-Din,	Professeur, UPMC	Examinateur

Contents

Part I

Introduction

Curves in Cryptography

The goal of cryptography is to enable two entities to secretely and efficiently exchange data, and to make any non-authorized third party — the attacker — unable to read it. To encrypt/decrypt the data, a *secret key* is used, which must be shared by the sender and the receiver. For this reason, those schemes are called *symmetric*. With the ever growing number of entities willing to securely communicate, the key distribution for symmetric ciphers became a major concern. The advent of *public-key cryptography* and *asymmetric schemes* answered this problem, and secured communication protocols now rely on an asymmetric brick to exchange a key for a symmetric cipher. The security of asymmetric schemes is based on "hard" mathematical problems.

In modern Cryptography, "hard" means that the computation would take a tremendous amount of times (for example, billions of years) with state-of-the-art machines. For nowaday standards, reasonable hardness starts at 2^{128} binary operations, but 2^{80} operations are still accepted for less crucial applications. Complexity classes for algorithms are also used to assess the difficulty of a computational problem. The running time of an algorithm solving a cryptographic problem can usually be expressed as a function of *the size* t of the inputs. With this notion of difficulty, a problem is considered hard when the best known algorithm to solve it belongs runs in a time exponential in t. Improvements are made when a subexponential algorithm that solves the problem is found, and the problem is considered unfit for cryptgraphic purpose — we sometime say it is "broken" — if a (quasi-)polynomial time algorithm can be designed (and implemented).

Asymmetric Cryptography and Discrete Logarithms

Public-key schemes are called asymmetric because the two parties involved do not share the same piece of information for an exchange. Indeed, in an asymmetric protocol, the sender uses the *public key* of the receiver to encrypt his message and send it to the receiver, who can then decrypt it using the *private key* associated to the public key. The sender thus only has access to the public key, while the receiver knows both. It should also be hard for an attacker to decrypt the message, while the owner of the associated private key can do it quickly. In other words, such an encryption scheme can be seen as a *one-way function with a trapdoor*, the trapdoor being the private key. The terminology highlights that it is easy to compute the output of such functions from the input, while it should be hard to recover the input from a given output without the additional information given by the trapdoor.

Officially, Public-key cryptography was born at the end of the 70's, when Diffie and Hellman [DH76] designed a key exchange protocol answering the problem of key-distribution in symmetric cryptography. Diffie-Hellman's solution used the exponentiation in a finite field as a one-way function, the trapdoor being the exponent. The underlying mathematical problem is known as the *Computational Diffie-Hellman Problem* (CDHP):

$$\textit{Given elements } g, g^a, g^b \textit{ in an abelian group } \mathbf{G}\textit{, compute } h = g^{ab}.$$

This problem is really close to the well-known *Discrete Logarithm Problem* (DLP), which can be stated as follows:

$$\textit{Let } g, h \textit{ be elements of an abelian group } \mathbf{G}\textit{. Find, if it exists, an integer } x \textit{ such that } g^x = h.$$

In fact, there are several instances where these problems are equivalent. However it is not known if this equivalence holds in the general case. Another protocol based on the problem of factoring large integers (IF) was proposed by Rivest, Shamir and Adleman [RSA78], and

it is better known by the initials RSA of its authors. Asymmetric cryptography grew rapidly, with new protocols answering several security-related problems. For example, the Digital Algorithm Signature (DSA), based on the DLP, gives a way to guarantee the authentication of the sender.

However, even the best things must come to an end. Both DLP and RSA-based cryptosystems are broken by fast *quantum* algorithms. Hence, the advent of the *quantum computer* should, in time, seal their doom. A big part of the community's efforts is now dedicated to the development of new mathematical foundations for cryptographic protocols (Lattices, Correcting codes, ...), gathered into the kingdom of *Post-Quantum Cryptography*. Nevertheless it is too soon to bury the Discrete Logarithm Problem and Integer Factorization, as a practical quantum computer is still nowhere to be seen, and is not expected in the next 20 years. Moreover, the majority of the current cryptosystems have bricks that rely on DLP or IF. It is fundamental to continue their study to ensure that the transition to the post-quantum world is smooth, and that our current notion of security does not crumble down suddenly.

Both IF and DLP can be used for key exchange and signature, but which one should we use in practice? The answer depends on the situation, but the question can mainly be reduced to a problem of the *size* of the key.

Key size and Elliptic Curves

The theoretical security of an assymetric cryptographic protocol is measured by the minimal known number of binary operations to recover the key. An immediate way to recover the key is by trying all the possibilities until the key is found — the *brute force* attack. If a given key can be represented using t bits, then at worst 2^t operations are needed to find the key. This gives an upper bound on any algorithm for attacking a cryptosystem, and in this way, it can be seen that the hardness of an assymetric scheme depends on the key size. While keys in symmetric cryptography are rarely larger than 256 bits, the same cannot be said for asymmetric cryptography. It is recommended by the National Institute of Standards and Technology (NIST) that any RSA key should at least be of 2048 bits to achieve a minimal security[1], and it is considered that a Finite Field Diffie-Hellman key exchange has the same security as a RSA.

In contexts where the key size is critical, such as embedded systems, how can one design a strong asymmetric scheme with rather small keys? This problem was adressed when Miller [Mil86a] and Koblitz [Kob87] proposed to use *the group of rational points of an elliptic curve* defined over a finite field as a basis for a DLP based cryptosystem. This marked the birth of *Curve Based Cryptography*, which has various developments and problematics. For DLP based cryptosystems, it is accepted that a 160 bits key using elliptic curve achieves the weakest accepted security, and that less than 256 bits are enough to achieve a 2048 bits RSA key security. In particular, elliptic curve-based cryptosystems should always be preferred when key size matters.

As we have seen already, the Discrete Logarithm Problem can be stated over any abelian group; the reader may wonder at first how to link elliptic curves and abelian groups. In fact, any algebraic curve can be related to an abelian group, and we now dive into more details on this topic.

[1]The NIST claims [BBB+12] that 15360 bits RSA key achieve the same security as a 256 symmetric AES key.

Discrete Logarithms over Algebraic Curves

The development of algebraic geometry showed that it is a natural association by introducing the *Jacobian variety* of a curve. Far more than just an abelian group, they enjoy the structure of *Abelian varieties*, that is to say projective varieties endowed with a group law that can be expressed using rational functions of the coordinates. Rational functions are close to polynomial functions, so that computational methods and therefore cryptosystems can be designed on Abelian Varieties. The dimension and the structure of the Jacobian variety depends on its underlying curve.

The family of Algebraic Curves is a vast and fascinating yet wild jungle. A classification of curves can be done by considering their *genus*, which also determine the dimension of the Jacobian variety. A genus 0 curve is more or less a projective line, with a trivial Jacobian variety. Thus, we are not interested in them in this work. Non-trivial Jacobian varieties start to appear with genus 1 curves, which are exactly the elliptic curves. Genus 2 curves are *Hyperelliptic curves*, and can be thought as their bigger sisters. When the genus is at least 3, a distinction appears between hyperelliptic curves and other curves.

Theoretically, all algebraic curves could be used to design a DLP-based cryptosystem. However, from a practical and computational point of view, only elliptic curves and hyperelliptic curves of genus 2, which means Abelian Varieties of dimension 1 and 2, are considered. This is due to two reasons: first, their arithmetics are the most efficient speed-wise and size-wise, which is obviously crucial for practical implementations. Second, the work of the cryptanalysts during the 20 last years [ADH99, Eng02, EG02, Die06, Gau00, EGT11] showed relative weaknesses for the other types. In term of raw security, it also happens that no efficient attack is known for most elliptic curves and hyperelliptic curves of genus 2, making them *a priori* stronger. While the reader now has an idea of which curves are strong, he might want to learn more about the attacks they are strong against and proceed to the next paragraph.

Computations of Discrete Logarithms

Not counting brute force, there are mainly three approaches to compute a discrete logarithm in a finite abelian group:

- *generic algorithms* only rely on the group law, and make no use of additional structures, *e.g* a ring structure or a geometrical structure;
- *Index-Calculus algorithms:* while countless variants exists, the main outline of the algorithm is always the same:
  1. select some elements in the target group; they form the *factor base*.
  2. Find enough linear combinations (*relations*) between the discrete logarithms of those elements. This phase is called *the harvesting*, and make intensive usage of additional structures (field structure, geometric or number theoretic properties, ...).
  3. Solve the linear system given by the relations and deduce the wanted discrete logarithm from the solution.
- *transfer attacks*, as the name suggests, transfer an instance of the DLP from a group to another — the additional structures may even be different — where an efficient method among the previous two may exist.

There is no hope to achieve an efficient attack using generic algorithms, as showed by Nechaev [Nec94], then Shoup [Sho97]: such methods are at best exponential in the size of the biggest

prime factor of the target group's order. The family of Index-Calculus algorithms is particularly rich[2], and lead to the first subexponential [ADH99, EGT11] and quasipolynomial [BGJT14] algorithms for certain classes of curves and finite fields, respectively. Transfer attacks target very specific curves: supersingular elliptic curves [MOV93, FMR94] and anomalous curves [Sma99, Sem98], curves defined over extension fields [GHS02, Die03]. It turns out that in this thesis, we are particularly interested in the latter curves.

Decomposition attacks and Algebraic Cryptanalysis

For curves defined over extension fields, a variant of Index-Calculus called *Decomposition attack* leads to promising asymptotic complexities [Gau09, Nag10]. There, one relation translates to a geometric property, which is modelled by a polynomial system using the concept of *Weil Descent*. In particular, these attacks belong to the realm of *Algebraic Cryptanalysis*. Their main concern is therefore the difficulty to solve polynomial systems, and a practical Decomposition attack first needs an efficient way of solving the systems arising from the harvesting. While solving polynomial systems over finite fields is known to be hard in general [GJ90], system arising from practical applications such as algebraic cryptanalysis are not random. This non-randomness sometimes hides algebraic properties, which may be used to simplify the solving process. It remains to clarify what we mean by "solving a system" in our applications.

Polynomial System Solving (PoSSo)

Polynomial systems appear in several practical applications, and "solving" can have a different meaning depending on the context. In algebraic cryptanalysis, solving means finding a way to represent all the solutions of the system in (an algebraic closure of) the base field. When the number of solutions is finite, we also want to list them all. The usual strategy rely on *Gröbner bases* methods. Informally, the idea is to transform the defining equations of the system into other equations admitting the same space of solutions, but with a shape that enables the solving. Introduced in Buchberger's thesis [Buc65], efficient algorithms to compute Gröbner bases using linear algebra have been designed since [Laz83, Fau99, Fau02], and are contained in several computer algebra language (Magma [BCP97], Maple, Sage, Singular, ...). While other methods exist, using Gröbner bases allows for a better understanding of the complexity of the resolution and therefore the efficiency of an algebraic cryptanalysis. Besides, the additional properties of the system can be read from a Gröbner basis, and exploited to speed up the computation: symmetries of the equations (physics-related [Sva14], cryptography-related [Spa12, Huo13, UDP15]), multi or weighted homogeneity [Spa12, FSEDV16]...

Motivations and objectives of this thesis

This thesis focuses on the algebraic cryptanalysis of the Discrete Logarithm Problem in the Jacobian variety of algebraic curves of genus greater than 1. A first reason is that genus 2 curves have been suggested as potential standards [BD04], and that recent works on their arithmetic have reached practical competitivity with elliptic curves (genus 1 curves) [BCHL16, CCS15, LR16, RSSB16]. Moreover, because of transfer attacks, a Discrete Logarithm Problem over an elliptic curve may be transferred over a curve of higher genus, where more efficient attacks may be known. Our main topic is the harvesting phase of the Index-Calculus algorithm, where efficiency can only be achieved by a deep understanding of the algebraic modelling, depending on the variant.

[2]Saying that there is a dedicated Index-Calculus algorithm for each curve or each finite field might not be an overstatement!

Motivations

Standardization of Elliptic Curves Several families of curves have been excluded from practical DLP-based applications after the work of the community of both cryptanalists and mathematicians (supersingular [FMR94], anomalous [Sma99, Sem98], non-hyperelliptic $\mathscr{C}_{ab}$ curves of small genus [Die06], ...). Nowadays, Elliptic Curves Cryptosystems are already widely developed. However, it is always possible that weaknesses of certain parameters may have been overlooked. For example, transfer attacks have been successfully combined with Decomposition attacks in [JV12] to threaten elliptic curves previously unreachable. The idea was to first transfer a DLP instance to an hyperelliptic curve of small genus (2 or 3) defined over a "small" extension field[3]. Then a Decomposition attack was run over the hyperelliptic curve to compute the target Discrete Logarithm. While the overall computation remained experimental, the total running time vastly improved over the estimations using methods dedicated to elliptic curves. This is a reason why the standardization of elliptic curves cannot be done without a good understanding on the hardness of the DLP for higher genus curves.

Index-Calculus for higher genus curves The first subexponential Index-Calculus algorithm for curves was first heuristic [ADH99], and it took additional works [Eng02, EG02] to propose a rigorous analysis together with a (theoretical) algorithm. Several other variants are also theoretically well-understood (Gaudry's [Gau00], Diem's [Die06], Decomposition attacks [Gau09, Die11, Nag10]). For all those algorithms, the question of their practical efficiency and thus their implementation can be asked. A first example of an answer is the optimized graph implementation of [LL15], but designed only for genus 3 non-hyperelliptic curves. Is it possible to design efficient implementations for other algebraic curves ? Other examples are given by the Decomposition attacks over elliptic curves of [FGHR14, FHJ+14, GG14], where *symmetries* of the equations were exploited to sharpen the complexity bounds on the polynomial systems' solving. As stated before, no such improvements were known for Decomposition attacks over hyperelliptic curves, while they share many properties with elliptic curves. This lack of knowledge is particularly concerning for genus 2 curves, which have been suggested as potential alternatives to elliptic curves — and even became competitive [Gau07, GL09, LR16, BCHL16, CCS15, RSSB16] — until post-Quantum cryptography becomes mandatory. Improvements on both asymptotical and practical algorithmic of algebraic curves may reveal rule out new families of curves. They may even enable a complete computation of a discrete logarithm.

Computations of Discrete Logarithms The best way to highlight weaknesses of curves is to estimate the running time to compute Discrete Logarithms in its Jacobian Variety, or even to sucessfully compute one. From the academic point of view, running a complete Index-Calculus on parameters achieving realistic security is also a proof-of-concept for improvements on the algorithmic aspect and weaknesses of a given family of curves. This can only be done if efficient methods are designed. While Decomposition attacks looked promising over elliptic curves, they reached a stalemate with [FHJ+14]. On the other hand, over higher genus curves, no practical Decomposition attacks were known at the beginning of this thesis for extension degree greater than 2. Nagao's approach [Nag10] remained mostly theoretical, the only conceivable cases being when $g = 2, 3$ and $n = 2, 3$: these were the only situations where estimations could be done in [JV12], and the running time to find a single relations still prevented any practicality. However there were no real in-depth analysis of Nagao's approach. It is natural to ask whether an efficient harvesting could be designed for particular curves or if the attack would stay theoretical. In the first case, computation of Discrete Logarithms or at least estimations on the necessary time are mandatory.

[3] By small, we mean here that the extension degree admits a small factor (between 2 and 6).

Objectives and Tools

First, this thesis aims at giving a clear framework on the various harvesting phases for Index-Calculus over Algebaic Curves of genus $g \geq 2$ by defining new concepts or generalizing existing notions, enabling a systematic study and a rigorous analysis of the complexity of the harvesting. Another goal is to propose new theoretic and algorithmic methods for the resolution of polynomial systems with particular algebraic properties over finite fields. As polynomial systems are involved in Decomposition attacks, the analysis of the new approaches is done in order to improve the complexity bounds on the harvesting phase. This way, a better insight on the strength of certain curves is obtained, which in turn refines which curves can be made practical or not.

On the theoretical side of the necessary tools, basic algebraic geometry, and particularly the theory of algebraic curve and their Jacobian Varieties, is needed to rigorously describe the various modellings of the harvesting phase. Special properties of the sytems are modelled using Invariant theory, Group theory and Field theory. From the computational point of view, powerful computer algebra and efficient Gröbner basis libraries are needed to perform experiments and give proof-of-concept for the new notions we develop. We mainly used Magma 2.19 [BCP97], which contains an efficient implementation of F4 algorithm [Fau99], to run our experiments. On some occasions, the FGb package [Fau10] in the Maple computer algebra system is used.

Contributions

Harvesting by Sieving

We design a Sieving approach to the harvesting phase in the classic Index-Calculus. Overall, such approaches are time-memory tradeoffs: cheap computations are stored instead of repeating expensive ones. We show how the reformulation of Sarkar-Singh's sieving approach to hyperelliptic harvesting [SS14] can be generalized to all algebraic curves. Our approach can also be adapted to any variants of Index-Calculus (Large Primes [GTTD07], Singularity based [DK13]). Our simpler reformulation also allows for an implementation that does not rely on sorting lists and can even prevent factorization of polynomials. Experiments and timings confirm that our approach perform better than both classical works by factors ranging from 3 to 7 (Tables 4.1 and 4.2). Those results can be found in Chapter 4 and lead to a publication at *LatinCrypt 2015* [VW15], co-authored with Vanessa Vitse.

Generalization of Semaev's Summation Polynomials

Before this thesis, Decomposition attacks were separated into two distinct worlds: the world of elliptic curves, whith Gaudry and Diem's usage of Summation Polynomial; and the world of other curves, where only Nagao's algorithm was known, using a more geometrical approach involving bases of Riemann-Roch spaces. Both ideas modelled the harvesting into solving polynomial systems with finite number of solutions. In Chapter 5 we present a new approach to the hyperelliptic decomposition attacks, by generalizing the notion of Summation Polynomials to *Summation Ideals* (Definition 5.4). In order to do this, we introduce new objects. For $\mathscr{H}$ a genus g imaginary hyperelliptic curve, let (P) stands for the canonical embedding of $\mathscr{H}$ into $\mathrm{Jac}(\mathscr{H})$, and let $m \geq 2g+1$. Let the m-summation map be $\Sigma_m : \mathscr{H}^m \longrightarrow \mathrm{Jac}(\mathscr{H})$, defined by $\Sigma_m(P_1,\ldots,P_m) = \sum_{i=1}^{m}(P_i)$. We then define the m-Summation Variety of $\mathscr{H}$ as the set $\mathscr{V}_m = \Sigma_m^{-1}(\mathscr{O}) = \{(P_1,\ldots,P_m) \ : \ \sum_{i=1}^{m}(P_i) = \mathscr{O}\}$, where $\mathscr{O}$ denotes the neutral element of $\mathrm{Jac}(\mathscr{H})$, and obtain the following result.

Theorem 0.1. *There exist an embedding of $\mathscr{V}_m$ into the affine space $\mathbb{A}^{2m-g}$ given by polynomial equations.*

The embedding is exhibited with a polynomial ideal $\mathscr{I}_m \subset \mathbb{F}[a_1,\dots,a_{m-g},X_1,\dots,X_m]$ (Theorem 5.3). The m^{th} Summation Ideal is then defined as the elimination ideal $\mathscr{I}_m \cap \mathbb{F}[X_1,\dots,X_m]$, equivalently as the ideal describing the projection of $\mathscr{V}_m$ over the m last coordinates, and we define Summation polynomials as any set of generators for the m^{th} Summation Ideal (Definition 5.4). The geometric study of the projection shows that the codimension equals the genus of the curve. In other words, as soon as $g \geq 2$, it can only be generated by *sets* of polynomials (of cardinality at least g) instead of one single polynomial as in Semaev's approach [Sem04]. Indeed, assuming the base field $\mathbb{F}$ is algebraically closed, we obtain the next results, which show that our new notion generalizes Semaev's:

Proposition 0.2. *For any $m \geq 2g+1$, a set $\mathbb{S}_m$ of m^{th} summation polynomials associated to $\mathscr{H}$ exists, and it verifies:*

$$\mathbb{S}_m(\mathbf{x}) = 0 \Leftrightarrow \exists\, y_1,\dots,y_n \in \mathbb{F} \text{ such that } P_i = (x_i,y_i) \in \mathscr{H}, 1 \leqslant i \leqslant m, \text{ and } (P_1)+\dots+(P_m) = \mathscr{O}.$$

The m^{th} Summation Ideal is globally invariant under the action of the m^{th} symmetric group $\mathfrak{S}_m$, and a set of symmetric generators exists and can be algorithmically computed.

In particular, the elliptic ($g = 1$) Summation Polynomial is recovered this way, together with its known properties (symmetry, irreducibility). Although we focus on hyperelliptic curves, everything extends to any algebraic curve. We formulate a conjecture (Conjecture 5.8) for the degree of the Summation variety:

Conjecture 0.3. *Let $\mathfrak{S}_m$ be the symmetric group on m elements, and $\mathscr{J}_m$ be the ideal generated by a symmetric set of Summation Polynomials. Then we have:*

$$\deg \mathscr{V}_m/\mathfrak{S}_m = 2^{m-g}, \quad \deg \mathscr{J}_m = 2^{m-g-1}.$$

This conjecture is verified on several examples of pairs (m,g) (see Tables 5.1 and 5.2). and is strenghtened both by intuition and the fact that it is known to be true when $g = 1$ [Die11].

A Summation-flavoured algorithm for Decomposition attacks

The new Summation notion allows us to design a new algorithm to harvest relations in Decomposition attacks, mixing Gaudry's [Gau09] and Nagao's [Nag10]. There are two ways of finding decompositions using Summation Ideals that we sum up briefly:

1.
   - Compute a symbolic Summation Set, with parameters for x-coordinates of points described by an element of $\mathrm{Jac}(\mathscr{H})$.
   - For an input $R \in \mathrm{Jac}(\mathscr{H})$, evaluate it at the coordinates given by R.
   - Build and solve the system describing a decomposition of R. It may not have a solution.

2.
   - For an input $R \in \mathrm{Jac}(\mathscr{H})$, compute a Summation Set already evaluated at the coordinates given by R.
   - Build and solve the system describing a decomposition of R. It may not have a solution.

Then the efficiency of this algorithm, and of any harvesting phase in Decomposition attacks in general, can be measured by the number of solutions of the systems to solve. Both methods seem really close, and in genus 1, they are equivalent and give the same number of solutions — the first is usually preferred. However, we show using our previous conjecture that when the genus of the curve is greater than 2, the degree obtained with the first approach is greater in Nagao's approach by a factor $2^{n(g-1)}$. This is confirmed by our experiments, and highlights

the fundamental impact of the genus on the construction of the Jacobian Variety. The analysis is done by favoring a even more geometrical approach, and defining *Specialized Summation Polynomials* and *Specialized Summation Variety* associated to a fixed $R \in \mathrm{Jac}(\mathscr{H})$:

$$\mathscr{V}_{m,R} = \{(P_1, \ldots, P_m) \ : \ \sum_{i=1}^{m} (P_i) = R\}.$$

We also conjecture that $\deg \mathscr{V}_{m,R} = 2^{m-2g}$. Overall, we obtain an algorithm that must solve systems with $d_{\mathrm{Nag}} = 2^{n(n-1)g}$ solutions if the target curve has genus g and is defined over some $\mathbb{F}_{q^n}$, which is the same as the degree obtained with a Nagao's modelling.

Degree reductions of the systems from Decomposition attacks, in even characteristic

As stated in the previous paragraph, the efficiency of the harvesting phase in Decomposition attacks can be estimated by the number D of solutions of the systems to solve. The reason behind this is that, in our situation, the bottleneck of the solving strategy for one system depends polynomially on D, see Section 1.3.3 for more details. In Decomposition attacks, D itself is exponential in the (square of the) extension degree, and exponential in the genus. Hence, for any hope of practical efficiency, this number must be reduced as much as possible. This can be done by exploiting the structure of the equations highlighted by the modelling. Such reduction were achieved in Decomposition attacks for elliptic curve in [FGHR14, FHJ+14].

For Hyperelliptic Curves in even characteristic, we observe specific algebraic structures linked to the degree of equations generating the systems, and exploit them to reduce the number of solutions. Either in Nagao's modelling or Summation modelling, the systems are generated with the following procedure. Over a hyperelliptic curve $\mathscr{H}$, a function with a known number of zeroes on the $\mathscr{H}$ can be expressed using a symbolic basis $a_1, \ldots, a_d$ of a certain linear space (see Riemann-Roch spaces, Definition 2.11). In particular, the intersection between $\mathscr{H}$ and such a function can be symbolically described by the coefficients of a polynomial in $(\mathbb{F}[a_1, \ldots, a_d])[x]$, that we define as the *Decomposition Polynomial* (Definition 6.2). In even characteristic, we analyze the properties of its coefficients and show that one is always a univariate polynomial, and that some other are always squares. We analyze the number of squares among the coefficients, and link it to the equation of the curve. More precisely, if the target hyperelliptic curve admits an equation as $y^2 + h_1(x)y = h_0(x)$ over some field of characteristic 2, then the number of squares is entirely determined by the *length* (Definition 6.6) of h_1.

Exploiting those properties is done differently whether we consider Nagaos's modelling or Summation modelling. In the former, square coefficients over $\mathbb{F}_{2^{kn}}$ lead to $n-1$ square equations over $\mathbb{F}_{2^k}$ in the system. Replacing those equations by their square root divides generally the number of solutions by 2 for each replacement. In Summation modelling, the squared coefficients of the Decomposition Polynomial are related to natural *weights* on the variables involved in the Summation Ideals. For genus g hyperelliptic curves defined over $\mathbb{F}_{2^{kn}}$, we show that a degree reduction of a factor up to $2^{(n-1)(g+1)}$ can be obtained from the number of solutions $d_{\mathrm{Nag}} = 2^{n(n-1)g}$. The exact factor is entirely determined by the length of h_1, and the (tight) lower bound we obtain is

$$d_{\mathrm{opt}} = 2^{(n-1)((n-1)g+1)}.$$

A complete classification for genus 2 curves is given, as there exist normal forms for them (described from [BD04, CF05] in Section 6.1.3), depending on $\deg h_1$. The following tables show the degree we obtained after reduction for Nagao's modelling in genus 2 in the best

cases, and the impact of this reduction on the running time for finding relations. In the first table, L_h stands for the length of the polynomial h_1 (see Definition 6.6). Similar tables are given for Summation modelling in the next paragraph.

Table 1: Degree reduction for Nagao's modelling in genus 2 for fields $\mathbb{F}_{2^{kn}}$

Curve Type	$\deg h_1$	L_h	n	**New degree**	Old degree
I_b with $h_1(x) = x^2$	2	0	2	$\mathbf{2 = d_{opt}}$	16
			3	$\mathbf{64 = d_{opt}}$	4096
			4	$\mathbf{2^{15} = d_{opt}}$	2^{24}
II	1	0	2	$\mathbf{2 = d_{opt}}$	16
			3	$\mathbf{64 = d_{opt}}$	4096
			4	$\mathbf{2^{15} = d_{opt}}$	2^{24}

Table 2: Impact of the degree reduction on Nagao's modelling (base field: $\mathbb{F}_{2^{45}}$)

Curve	Modelling	DRL	FGLM	Total	Ratio
Type I_b, $h_1(x) = x^2$, $d_{old} = 4096, d_{red} = 64$	Old	166.76s.	34152s. !!	34318s. !!	$\mathbf{1.7 \cdot 10^6}$
	New	**0.02s.**	**0.000s.**	**0.02s.**	
Type *II*, $h_1(x) = x$, $d_{old} = 4096, d_{red} = 64$	Old	185.56s.	33917s. !!	34102s !!	$\mathbf{1.1 \cdot 10^6}$
	New	**0.02s.**	**0.009s.**	**0.029s.**	

Here, the timings highlighted by exclamation marks seem far too long for the type of systems considered. They were obtained with Magma 2.19's FGLM algorithm. We comment on this, along with more details in Sections 6.1 and 6.3.2.

Action of the Frobenius automorphism over ideals of relations in positive characteristic

To obtain a precise analysis of the degree reduction in even characteristic for Summation modelling, we put ourselves in a general context of polynomial parametrizations in positive characteristic $p > 0$ with action of the Frobenius automorphism. More precisely, we fix a field $\mathbb{F}$ with $\mathrm{Char}(\mathbb{F}) = p > 0$ and we consider ideals as

$$I = \langle\, X_1 - P_1(\mathbf{a})^p, \ldots, X_k - P_k(\mathbf{a})^p, X_{k+1} - P_{k+1}(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a}) \rangle\,,$$

with P_i polynomials in $\mathbb{F}[a_1, \ldots, a_l]$. Then, the *ideal of relations* between the P_i is the elimination ideal $I_e = I \cap \mathbb{F}[X_1, \ldots, X_m]$ (Proposition 1.71). Now let $J = \langle\, X_1 - P_1(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a}) \rangle$, $J_e = J \cap \mathbb{F}[X_1, \ldots, X_m]$, and for any polynomial $f = \sum c_\alpha \mathfrak{m}_\alpha$, define $f^\sigma = \sum c_\alpha^p \mathfrak{m}_\alpha$. Keeping previous notations, we show the following:

Proposition 0.4. *Let $J_e = \langle\, g_1, \ldots, g_r \,\rangle$ and define $I' = \left\langle\, g_i^\sigma(X_1, \ldots, X_k, X_{k+1}^p, \ldots, X_m^p) \;:\; 1 \le i \le r \right\rangle$. Then I_e is the radical of I'*

This implies that it is equivalent to find points in $\mathbf{V}(I_e)$ or $\mathbf{V}(J_e)$. However, using the ideal J_e is more efficient in practice, because the weighted degree of J_e is reduced by roughly the product of the weights:

Proposition 0.5. *With the same notations as in the previous Proposition, we have*

$$\deg_w J_e = \frac{\deg I'}{p^{m-k}}.$$

The degree reduction process for Summation modelling in even characteristic follows from an instantiation to $p = 2$ and the parametrization of the Specialized Summation Variety of

those results. The degree of the systems describing relations in a Decomposition attack is then deduced from a careful analysis of the Weil Descent. Upon this analysis, a complete classification for genus 2 curves is also given. The best cases are presented below.

Table 3: Degree reduction for Summation modelling in genus 2 for fields $\mathbb{F}_{2^{kn}}$

Curve Type	$\deg h_1$	n	**New degree**	Old degree
I_b with $h_1(x) = x^2$	2	2	$\mathbf{2 = d_{opt}}$	16
		3	$\mathbf{64 = d_{opt}}$	4096
		4	$\mathbf{2^{15} = d_{opt}}$	2^{24}
II	1	2	$\mathbf{2 = d_{opt}}$	16
		3	**128**	4096
		4	$\mathbf{2^{17}}$	2^{24}

The next table shows the impact of the degree reduction. The column "Summation Set" gives the time to compute a specialized summation set. It also shows how considering the weight structure improved the computation of the ideal of relations.

Table 4: Impact of the degree reduction on Summation modelling (base field: $\mathbb{F}_{2^{45}}$)

Curve	Modelling	Summation Set	DRL	FGLM	Total	Ratio
Type I_b, $h_1(x) = x^2$, $d_{old} = 4096, d_{red} = 64$	Old	1.04s.	0.9s.	8.7s	10.64s.	**31**
	New	**0.27s.**	**0.06s.**	**0.01s.**	**0.34s.**	
Type *II*, $h_1(x) = x$, $d_{old} = 4096, d_{red} = 64$	Old	0.84s	0.65s.	7.7s	9.19s	**23**
	New	**0.27s.**	**0.14s.**	**0.01s.**	**0.42s.**	

More details are found in Section 6.2.1.

Harvesting on curves with realistic parameters and practical impact

Before this thesis, the only known approach to Decomposition attacks for hyperelliptic curves defined over $\mathbb{F}_{q^n}$ was Nagao's, which was unpractical as soon as $n > 2$. For example, more than 1 million seconds were needed to find a single decomposition for a genus 2 curve defined over $\mathbb{F}_{2^{45}}$ with a Magma implementation, solving hundreds of systems with 4096 solutions. Our comparisons with Summation modelling reveal that Nagao's approach performs overall better. In even characteristic, our degree reduction strategy for Nagao's modelling decreases the time to find a relation for a genus 2 curve defined over $\mathbb{F}_{2^{45}}$ by a factor of 75000. The Summation modelling reveals to be slower in general. Nevertheless, an immediate consequence of those improvements is that we increased the realm of practical experiments to binary genus 2 curves defined over fields as $\mathbb{F}_{2^{3k}}$.

A second consequence is that a practical harvesting phase on a curve with realistic parameters can now be considered. To show this, we find a genus 2 curve defined over $\mathbb{F}_{2^{93}}$, whose Jacobian variety have almost prime order and satisfies a generic security bound of 2^{92} operations, and run the harvesting with a dedicated implementation. The implementation mixes code-generation techniques and efficient Gröbner bases libraries (such as the FGb package [Fau10]). With this code, around 2 seconds are needed to find a relation (see Section 6.3.3 for more details). Using 8000 cores, the harvesting can build the matrix of relations in a bit more than 7 days. This highlights additional weaknesses of binary genus 2 curves against Decomposition attacks, and in particular the base field should never be an extension whose degree can be divided by 3.

Perspectives

Theoretical framework for Summation ideals over Abelian Varieties

Following the work of Gaudry [Gau09] and Diem [Die11], a framework for Summation Ideals for Elliptic Curve was proposed. The basic idea is that Summation Polynomials are associated to a choice of a projection $\pi : E \longrightarrow \mathbb{P}^1$, or equivalently, to a choice of a model for E. Several Summation Polynomials can then be deduced from the classical one using the action of $\mathrm{Aut}(\mathbb{P}^1) = \mathrm{PGL}_2$. This is the point of view that lead the authors of [FHJ+14] to fully exploit the group of symmetry acting over the Summation Polynomial in an Index-Calculus context. Elliptic Curves corresponds exactly to Abelian Varieties of dimension 1, and the projective line $\mathbb{P}^1$ is in fact the *Kummer Variety* of the Elliptic Curve. For a more general Abelian Variety, Summation Ideals can be defined using different projections over different models for the associated Kummer Variety.

It is natural to expect a generalization of Summation Ideals and Polynomials to any Abelian Variety using projections. This can shed new lights on the arithmetic of Abelian Varieties. Besides, Abelian Variety of dimension 2 corresponds exactly to Jacobian Varieties of Hyperelliptic Curves of genus 2. In this situation, the Kummer variety is a *Kummer Surface*, a quartic surface which is well-known in the litterature, even from a computational point of view [Fly93, Gau07, GL09, Duq10, BCHL16, RSSB16]. Hence explicit computations and experiments can be hoped on genus 2 curves.

Automorphisms of Summation Varieties

As we stated in the previous paragraph, the geometrical framework for Summation Polynomials of an Elliptic Curve E have lead Faugère & al. [FHJ+14] to exploit a large group of automorphisms . More precisely, by considering the addition of 2-torsion points in E, they identified a group G of automorphisms acting over the Summation Polynomials. Using a set of fundamental invariants expressing the action, they reduced the degree of systems arising from a Decomposition attack by a factor $\#G$. Moreover, the clear link with $\mathrm{Aut}(\mathbb{P}^1)$ enabled them to build factor bases that remained invariant by the addition of 2-torsion points to its elements, further improving on the efficiency of the Decomposition Attacks. Unfortunately for us, the object introduced by our generalization are not invariant under the action of rational 2-torsion points in a Hyperelliptic Jacobian Variety.

Exploiting more symmetries when $g \geq 2$ can be done by understanding the group of automorphsim acting over our new Summation Ideals.. With a general framework for Abelian Varieties, other groups of symmetry can be used on other sets of Summation Polynomials. For example in genus 2, the addition of a 2-torsion point in the Jacobian Variety expresses as a linear (projective) map over the Kummer Surface. Hence it can be represented by matrices in $\mathrm{Aut}(\mathbb{P}^3) = \mathrm{PGL}_4$ [Fly93]. Explicit formulae are likely to be handled for this size of matrices, leading to experimental computations. Ultimately, it could lead to a new Decomposition attack on genus 2 hyperelliptic curves.

Complete Transfer and Decomposition attack over a meaningful binary Elliptic Curve

A first threat to an Elliptic Curve defined over $\mathbb{F}_{q^6}$, q odd with $\log q = 23$, was presented by Joux and Vitse in [JV12]. Their method used first a transfer of a DLP instance to the Jacobian Variety J of an Hyperelliptic Curve of small genus ($g = 2, 3$) defined over $\mathbb{F}_{q^{6/g}}$, then a Decomposition attack in J. The complexity of their attack mainly revolves around the efficiency of the harvesting, since the linear algebra is a somewhat rigid phase. While they

mostly target the odd characteristic, their analysis also covers the even characteristic. However no example of computations were given so far. Our work on the reduction of the degree for Decomposition attack in even characteristic showed that a really efficient harvesting could be design on certain binary Hyperelliptic curves. Our results in Chapter 6 suggest that a genus 2 curve defined over some $\mathbb{F}_{2^{3k}}$, or a genus 3 Hyperelliptic Curve defined over some $\mathbb{F}_{2^{2k}}$ could be targeted.

It would be interesting to work toward a complete Transfer and Decomposition attack on an Elliptic Curve defined over $\mathbb{F}_{2^{6k}}$, with $k \approx 32$. A first step is to find convenient hyperelliptic covers of genus $2,3$, with base field being $\mathbb{F}_{2^{3k}}$ or $\mathbb{F}_{2^{2k}}$. Then a dedicated implementation of a Decomposition attack exploiting our degree reduction process and powerful Gröbner bases libraires must be done. Lastly, the running time for a complete Discrete Logarithm computation have to be estimated. It could be enough for a practical computation of a Discrete Logarithm.

Open Problem

Algorithmic of Gröbner Bases for ideals with positive dimension

The computational aspect of Gröbner bases for 0-dimensional ideal is already well understood. Recent works sharpen the complexity bounds under various additional hypothesis ([Spa12], [Sva14], [FSEDV16]), and major improvements on the computational efficiency are already observed. The most efficient computational strategy is to rely on the change-ordering algorithm FGLM, which can be really efficient in practice. However, the situation for positive dimensional ideal is far less understood. FGLM's algorithm is not efficient when the dimension of the ideal is positive. Another approach using the *Gröbner Walk* algorithm exists (implemented *e.g.* in Magma), but its complexity is not clear, and, in practice, it is generally much slower than a direct computation with F4 algorithm for an elimination order. Nevertheless, a direct computation for such orders is also much slower than a computation for a (weighted) total degree order. This thesis highlighted that it can be interesting to consider such ideals: examples are given by the Summation Varieties we introduced in Chapter 5.

Understanding the behaviour of elimination orders with respect to F4 and F5 algorithms can be a first step toward more efficiency in Gröbner Basis computations for positive dimensional ideals. This can lead to clearly understood computational strategies, or even to the design of new algorithms dedicated to elimination orders. Also, sharp complexity bounds can be derived from such new insights about positive dimension.

Organisation of the thesis

The manuscript is divided in two parts. The first part contains both the necessary material to understand the results of this thesis and presents the Discrete Logarithm Problem and its application in Cryptography, strongly centered around Algebraic Curves. The first chapter deals with algebraic varieties from a computational point of view, and follows mostly [CLO97]. A special emphasis is done on solving systems with a finite number of solutions. The second chapter introduces algebraic curves, inspired from [Ful08, Sil13]. The basics of the theory are followed by a focus on hyperelliptic and elliptic curves, a main concern for the results of this thesis. Chapter 3 deals with the Discrete Logarithm Problem. Section 3.1.2 states the problem together with its applications in Cryptography, and can be read indepndently from the rest of the manuscript. It is followed by a State-of-the-Art on the Discrete Logarithm Problem over Algebraic Curves. A strong focus on Index-Calculus is done, as it is the main topic of this thesis.

The second part gathers the contributions of this work in three separated chapters. The sieving approach to the harvesting in general Index-Calculus is first presented in Chapter 4. The text is really close to that of the article *Improved Sieving on Algebraic Curves* [VW15], co-authored with Vanessa Vitse and published at LatinCrypt 2015. Summation Ideals are the topic of Chapter 5. We start by defining this new notion, then getting practical with Index-Calculus in mind. A new algorithm for harvesting in Decomposition attacks is proposed in Section 5.4.4. The last Chapter of this manuscript mainly deals about the degree reduction of the systems in even characteristic. It also contains the results about the Frobenius action over ideals of relations in positive characteristic. It concludes with a precise description of the realistic simulation of the harvesting for a binary genus 2 curve defined over $\mathbb{F}_{2^{93}}$, whose Jacobian Variety achieves a 2^{92} operations generic security bound.

Part II

Preliminaries

Chapter 1

Polynomial Ideals and Algebraic Varieties

In this thesis, we are generally interested in algebraic varieties defined over finite fields, or in other words, the geometrical locus of a system of polynomial equations with coefficients in some $\mathbb{F}_q$. The set of all polynomial combinations between the defining equations of a system is called *an ideal*, and by the Ideal-Variety correspondence 1.9, any geometric property of the variety translates to an algebraic equivalent in the language of ideals and polynomials. Section 1.1 introduces this dictionnary for affine and projective varieties.

There is a clear separation in our interests for this work depending on the *dimension* of the target variety. Hence, this notion is rigorously defined in Section 1.2 before going any further. We choose an *Hilbert Series*-based approach, with a special emphasis on weighted structures, since several estimates in our contribution (see Section 6.2.1) rely on these important objects. Along the way, this also enables us to define the *(weighted) degree* of a polynomial, but also of an ideal and of a variety (Definition 1.28). Informally, those quantities generalize the degree of a univariate polynomial to count the expected number of solutions (in an algebraic closure) of a polynomial system. They are also needed in all the complexity estimates for Decomposition attacks, the main focus of this thesis.

Positive-dimensional and zero-dimensional varieties are targeted differently by our contributions. When the dimension is positive, we want to compute particular projections (Chapter 5). When the dimension is 0, that is to say, the variety has only a finite number of points with coordinates in the algebraic closure, we want to find them all. Equivalently, we want to efficiently solve the system of defining equations — see for example Section 3.4.2 and Chapter 6. Once an adequate *monomial order* is fixed, which means that we choose a way to sort multivariate monomials, both goals can be achieved by the computation of a *Gröbner basis* for this order (Definition 1.41). In other words, theses objects give us a computational tool to manipulate algebraic varieties, and are thus treated in Section 1.3.

The algorithmic of Gröbner bases is a fundamental brick of our work: our experiments can only be done by Gröbner bases computations, and most of our results involve improvements of such computations — Chapter 5 and 6. Therefore the entire Section 1.3.2 is devoted to this subject. We briefly remind the reader how Gröbner bases can be computed by linear algebra. The efficient algorithms F4 [Fau99] and [Fau02] are then informally presented. Their complexity is well-understood for *regular sequences* (Definition 1.48). However, their output is not suited for solving zero-dimensional systems, *i.e.* with a finite number of solutions, that appear in most of our contributions. The standard strategy for solving such systems relies on changing the monomial order. This is done using FGLM's algorithm [FGLM93, FM11], introduced thereafter. Before our work, this step dominated the running time in all experiments.

Since its complexity mainly depends on the number of solutions of the system (Proposition 1.58), reducing this number as much as possible is what we strive for (see Chapter 6).

After the needed theory on zero-dimensional varieties has been developed, Section 1.4 focuses on positive-dimensional varieties. In Chapter 5, we use the projection of a particular variety of positive dimension to design a new modelling for the harvesting phase in Decomposition attacks. This modelling is a core part of most of our results. Understanding the relations between a variety and its projections is the topic of Elimination theory, briefly developed in Section 1.4. The name stems from "by-hand" methods to solve linear and polynomial systems where combinations of the equations are used to eliminate some variables, until it is possible to solve it. From a computational point of view, describing a projection can be done by computing a Gröbner basis for an *elimination ideal* of the variety (Definition 1.59). The resultant of two polynomials (Definition 1.64) is also an important object of elimination theory. It is involved in the computation of Summation Polynomials in Section 3.4.2, and we also use it in our new modelling for Decomposition attacks (Section 5.1 and 5.4, Section 6.1.1).

This work makes a special emphasis on two special classes of varieties, detailed in Section 1.5.1, the last of this Chapter. First, *Polynomial parametrizations* are involved in our new definition of Summation Ideals in Chapter 5. They are closely related to *Ideals of relations*, which we use to generalize Semaev's Summation Polynomials [Sem04]. Second, we focus on Weil Restrictions of varieties defined over extensions of finite fields. Given a variety V defined over $\mathbb{F}_{q^n}$, the Weil Restriction associates a variety $\mathscr{W}_n(V)$ defined over $\mathbb{F}_q$, where $\mathbb{F}_{q^n}$-rational points in V are in one-to-one correspondence with $\mathbb{F}_q$-rational points of $\mathscr{W}_n(V)$. Such varieties are used to model the harvesting phase in Decomposition attacks as the solving of zero-dimensional polynomial systems. Understanding their degrees allows us to give precise estimates on the complexity of our new harvesting phases starting Chapter 5.

1.1 The Ideal-Variety Correspondence

We start by a reminder on *ideals* and *affine algebraic varieties*, and link them together using results from Hilbert — Basis Theorem, Weak and Strong Nullstellensatz. The one-to-one correspondence between *radical ideals* and affine algebraic varieties is also described. We also choose to consider non-irreducible varieties as varieties on their own; they correspond to *prime ideals*.

1.1.1 Affine Varieties and Radical Ideals

Most of the results in this Section are directly taken from [CLO97]. Let K be a field and denote by $\overline{K}$ its algebraic closure. For any field K, let $K[X_1,\dots,X_n]$ be its ring of polynomials in n variables.

Definition 1.1. *A set I in $K[X_1,\dots,X_n]$ is called a polynomial ideal if it is an additive subgroup closed under multiplication.*

If $f_1,\dots,f_r$ are polynomials in $K[X_1,\dots,X_n]$ we can define the ideal generated by $f_1,\dots,f_r$ as the set of all polynomial combinations between them, namely:

$$\langle f_1,\dots,f_r\rangle = \left\{ \sum_{i=1}^{r} h_i f_i,\ h_i \in K[X_1,\dots,X_n] \right\}.$$

The well-known Hilbert's Basis theorem says that all polynomial ideals can be described this way.

Theorem 1.2 (Hilbert's Basis Theorem)**.** *All polynomial ideals are finitely generated.*

We can associate to any ideal a set of points in an affine space called an *affine algebraic variety*. Let $\mathbb{A}^n(K)$, or $\mathbb{A}^n$ when the context is clear, be the n-dimensional affine space over K.

Definition 1.3. *Let $f_1,\dots,f_r$ be polynomials in $\overline{K}[X_1,\dots,X_n]$. An affine algebraic variety is the set of common zeroes of $f_1,\dots,f_r$ in $\mathbb{A}^n(\overline{K})$ and it is denoted by $V=\mathbf{V}(f_1,\dots,f_r)$. Let $L|K$ be any field extension. If $f_1,\dots,f_r\in L[X_1,\dots,X_n]$, then we say that V is defined over L. If V is defined over L, its set of L-rational points is denoted by*

$$\mathbf{V}_L(f_1,\dots,f_r)=\{P=(x_1,\dots,x_n)\in\mathbb{A}^n(L)\ :\ f_1(P)=f_1(x_1,\dots,x_n)=\dots=f_r(P)=0\}.$$

Any $P=(x_1,\dots,x_n)$ in $\mathbf{V}_L(f_1,\dots,f_r)$ is called a L-rational point.

What we define as algebraic varieties are sometime called *algebraic sets* in the litterature, while the name "variety" is reserved for *irreducible* algebraic sets — see Definition 1.11. In our work we deal with varieties defined over (extension of) finite fields $\mathbb{F}_q$, and we are usually looking for $\mathbb{F}_q$-rational points. When the context is clear we often omit the field in the subscript.

Proposition 1.4. *Let I be an ideal of $K[X_1,\dots,X_n]$. The set $\mathbf{V}(I)=\{P\in\mathbb{A}^n(\overline{K})\ :\ f(P)=0\ \forall\ f\in I\}$ is an affine algebraic variety, called the affine algebraic variety associated to I.*

When $I=\langle f_1,\dots,f_r\rangle$, or when S is a given generating set for I, we also use the notations $\mathbf{V}(I)=\mathbf{V}(S)=\mathbf{V}(f_1,\dots,f_r)$. An affine algebraic variety can be empty, but when the field is algebraically closed, this implies a strong property on its defining ideal. This is known as Hilbert's Weak Nullstellensatz.

Theorem 1.5 (Weak Nullstellensatz)**.** *Let K be an algebraically closed field, and $I\subset K[X_1,\dots,X_n]$ be an ideal. Then $V(I)=\emptyset$ if and only if $I=K[X_1,\dots,X_n]$.*

Example: Let $K=\mathbb{F}_q$ and P be an irreducible polynomial in $\mathbb{F}_q[X]$ with $\deg P=d>1$, and define the ideal $I=\langle P\rangle$ in $\mathbb{F}_q[X]$. Then $\mathbf{V}_K(I)=\emptyset$ while $I\neq K[X]$. Now let L the field of decomposition of P. Then P has roots $x_1,\dots,x_d$ in L, and $\mathbf{V}_L(I)=\{x_1,\dots,x_d\}=\mathbf{V}(I)$.

Starting from an affine algebraic variety V in $\mathbb{A}^n$, we can do the opposite and associate a polynomial ideal to V as

$$\mathbf{I}(V)=\{f\in\overline{K}[X_1,\dots,X_n]\ :\ f(P)=0\ \forall\ P\in V\}.$$

This set is indeed an ideal: let $f,g\in\mathbf{I}(V)$ and $h\in\overline{K}[X_1,\dots,X_n]$, then for any $P\in V$, we have $(f+g)(P)=f(P)+g(P)=0$ and $(f\cdot h)(P)=f(P)h(P)=0$. Similarly, when V is defined over K, we define $\mathbf{I}_L(V)=\{f\in L[X_1,\dots,X_n]\ :\ f(P)=0\ \forall\ P\in V\}$ for any extension $L|K$.

Overall we defined a *correspondence* between affine algebraic varieties and polynomial ideals. This correspondence is however not one-to-one on the $\mathbf{V}$ side. For example, let $I=\langle X\rangle$ and $J=\langle X^2\rangle$ in $\overline{K}[X]$. It is clear that $\mathbf{V}(I)=\mathbf{V}(J)=\{0\}$. Conversely, it is also clear that $\mathbf{I}(\{0\})=I$. It turns out that I is the *radical* of J.

Definition 1.6. *Let $I\subset K[X_1,\dots,X_n]$ be an ideal. The ideal I is radical if $f^m\in I$ for some integer m implies $f\in I$. The radical of I is the set $\sqrt{I}=\{f\in K[X_1,\dots,X_n]\ :\ \exists\, m\in\mathbb{N}\ s.t.\ f^m\in I\}$.*

Proposition 1.7. *The radical $\sqrt{I}$ of an ideal I is a radical ideal containing I. An ideal I is radical if and only if $\sqrt{I}=I$. The radical of I is the smallest radical ideal containing I.*

The notion of radical ideal leads to a one-to-one correspondence between radical polynomials ideals and affine algebraic varieties. This is in fact a reformulation of Hilbert's Strong Nullstellensatz.

Theorem 1.8 (Strong Nullstellensatz)**.** *Let K be an algebraically closed field, and $f, f_1, \dots, f_r \in K[X_1, \dots, X_n]$. If $f \in \mathbf{I}(\mathbf{V}(f_1, \dots, f_r))$, then there exists an integer m such that $f^m \in \langle f_1, \dots, f_r \rangle$.*

Corollary 1.9 (Ideal-Variety correspondence)**.** *Let K be an algebraically closed field and I be an ideal in $K[X_1, \dots, X_n]$. Then we have:*

$$\mathbf{I}(\mathbf{V}(I)) = \sqrt{I}.$$

If $I_1 \subset I_2$ are ideals, then $\mathbf{V}(I_2) \subset \mathbf{V}(I_1)$. Similarly, if $V_1 \subset V_2$ are varieties, then $\mathbf{I}(V_2) \subset \mathbf{I}(V_1)$. Let V be an affine algebraic variety in $\mathbb{A}^n$. Then we have:

$$\mathbf{V}(\mathbf{I}(V)) = V.$$

In other words, $\mathbf{I}$ and $\mathbf{V}$ are inclusion-reversing bijections, inverses of each other, between the set of radical ideals in $K[X_1, \dots, X_n]$ and the set of affine algebraic varieties in $\mathbb{A}^n$.

Proof. Let $f \in \sqrt{I}$. Then $f^m \in I$ for some m so that f^m vanishes on $\mathbf{V}(I)$, and f also vanishes on $\mathbf{V}(I)$. The other inclusion comes from the Strong Nullstellensatz. Let $I_1 = \langle f_1, \dots, f_r \rangle$ and $I_2 = \langle g_1, \dots, g_s \rangle$. For $P \in \mathbf{V}(I_2) = V(g_1, \dots, g_s)$, we have $g_i(P) = 0$ for all $1 \leq i \leq s$. Now for all $1 \leq i \leq r$, there are $h_i \in K[X_1, \dots, X_n]$ such that $f_i = \sum h_j g_j$, and thus $f_i(P) = 0$, which means $P \in \mathbf{V}(I_1) = V(f_1, \dots, f_r)$. If $V_1 \subset V_2$, then any polynomial that vanishes on V_2 vanishes in particular on V_1, so $\mathbf{I}(V_2) \subset \mathbf{I}(V_1)$. Let now $V = \mathbf{V}(f_1, \dots, f_r)$ be an affine algebraic variety. By definition, $V \subset \mathbf{V}(\mathbf{I}(V))$. For the other inclusion, notice that $f_1, \dots, f_r$ are in $\mathbf{I}(V)$. Since $\mathbf{V}$ reverses the inclusion, this gives $\mathbf{V}(\mathbf{I}(V)) \subset \mathbf{V}(f_1, \dots, f_r) = V$. Notice that the first statement of the proposition also means that for any affine algebraic variety V, $\mathbf{I}(V)$ is a radical ideal. To conclude, we observe that if I is radical, then $\sqrt{I} = I$ and we have $\mathbf{I}(\mathbf{V}(I)) = I$. □

Remark 1.10. *For any ideal I, we also have $\mathbf{V}(I) = \mathbf{V}(\sqrt{I})$.*

We give some more insights about *irreducible varieties* and their link to polynomial ideals.

Definition 1.11 (Irreducible Variety, Prime Ideal)**.** *An algebraic affine variety V is said to be irreducible if, whenever there are two affine varieties V_1 and V_2 such that $V = V_1 \cup V_2$, then either $V_1 = V$ or $V_2 = V$.*
An ideal I in $K[X_1, \dots, X_n]$ is said to be prime if, whenever $f, g \in K[X_1, \dots, X_n]$ are such that $f \cdot g \in I$, then either $f \in I$ or $g \in I$.

Prime ideals are radical: indeed, let I be a prime ideal and f be a polynomial such that $f^m \in I$. Then by definition, either f^{m-1} of f is in I, and the statement follows by induction.

Proposition 1.12. *An affine algebraic variety is irreducible if and only if $\mathbf{I}(V)$ is a prime ideal.*

1.1.2 Projective Varieties and Homogeneous Ideals

This Section follows [Sil13, Ful08]. In most of our applications, it is enough to consider affine algebraic varieties. However, to define properly several notions in Section 1.2 and Chapter 2, it is necessary to consider projective varieties. In some sense — see Proposition 1.17 — they can be thought as "completions" of affine varieties.

Definition 1.13 (Projective n-space)**.** *Let K be a field. The projective n space is defined as $\mathbb{P}^n(K) = K^{n+1}/\sim$, where $\sim$ denotes the equivalence relation $(x_0, \dots, x_n) \sim (x'_0, \dots, x'_n)$ if and only if there exists $\lambda \in K^*$ such that $x_i = \lambda x'_i$ for all i. A representant of the equivalence class is denoted $[x_0 : \dots : x_n]$ and called homogeneous coordinates.*

If the context is clear, we usually omit K when describing a projective space. Projective varieties are defined in term of *homogeneous* ideals and polynomials.

Definition 1.14. *Let $d \geq 1$. A polynomial f in $K[X_0,\ldots,X_n]$ is homogeneous of degree d if $f(\lambda X_0,\ldots,\lambda X_n) = \lambda^d f(X_0,\ldots,X_n)$ for all $\lambda \in K^*$. An ideal I in $K[X_0,\ldots,X_n]$ is homogeneous is there exists a generating set of homogeneous polynomials.*
Let I be a homogeneous ideal in $\overline{K}[X_0,\ldots,X_n]$. A projective variety is defined as the set of common zeroes in $\mathbb{P}^n(\overline{K})$ of the homogeneous polynomials in I, and denoted $\mathbf{V}(I)$. Let $L|K$ be any field extension. If the ideal I is in $L[X_1,\ldots,X_n]$, then we say that $\mathbf{V}(I)$ is defined over L, and its set of L-rational points is

$$\mathbf{V}_L(I) = \{P \in \mathbb{P}^n(L) \ : \ f(P) = 0 \ \forall \ P \in I\}.$$

Similarly, let V be a projective variety. The ideal associated to V is the ideal

$$\mathbf{I}(V) = \left\langle \ f \in \overline{K}[X_0,\ldots,X_n] \ : \ f \text{ is homogeneous and } f(P) = 0 \ \forall \ P \in V \ \right\rangle,$$

adding the subscript L if the polynomials are considered with coefficients in L and if V is defined over K. A projective variety V is irreducible if its associated ideal is prime.

The projective n-space contains many copies of $\mathbb{A}^n$. Some of them are particularly interesting: consider the projective hyperplane H_i defined by the ideal $\langle X_i \rangle$ in $K[X_0,\ldots,X_n]$ for some i, and its complement $U_i = \{[x_0 : \ldots : x_n] \in \mathbb{P}^n \ : \ x_i \neq 0\}$. There is a natural map

$$\begin{array}{rccc} \varphi_i : & \mathbb{A}^n & \longrightarrow & \mathbb{P}^n \\ & (y_1,\ldots,y_n) & \longmapsto & [y_0 : \ldots : y_{i-1} : 1 : y_{i+1} : \ldots : y_n], \end{array}$$

whose image is U_i, and whose inverse on U_i can be defined by

$$\begin{array}{rccc} \varphi_i^{-1} : & U_i & \longrightarrow & \mathbb{A}^n \\ & [x_0 : \ldots : x_n] & \longmapsto & (\frac{x_0}{x_i},\ldots,\frac{x_{i-1}}{x_i},\frac{x_{i+1}}{x_i},\ldots,\frac{x_n}{x_i}). \end{array}$$

In a similar way, we can link polynomials in n variables and homogeneous polynomials in $n+1$ variables.

Definition 1.15 ((De)Homogenization of a polynomial)**.** *Fix $0 \leq i \leq n$. Let $f \in K[X_0,\ldots,X_n]$ be a homogeneous polynomial. The dehomogenization of f wrt. X_i is the polynomial*

$$f_*(Y_1,\ldots,Y_n) = f(Y_1,\ldots,Y_{i-1},1,Y_{i+1},\ldots,Y_n).$$

Let g be any polynomial in $K[Y_1,\ldots,Y_n]$. The homogenization of g wrt. X_i is the polynomial

$$g^*(X_0,\ldots,X_n) = X_i^{\deg g} g\left(\frac{X_0}{X_i},\ldots,\frac{X_{i-1}}{X_i},\frac{X_{i+1}}{X_i},\ldots,\frac{X_n}{X_i}\right).$$

If I is any non-homogeneous ideal in $K[X_1,\ldots,X_n]$, the homogenization of I is the homogeneous ideal $I^* = \langle f^* \ : \ f \in I \rangle$ in $K[X_0,\ldots,X_n]$.

From these observations we can also link affine varieties and projective varieties.

Definition 1.16. *Fix $0 \leq i \leq n$. Let V be a projective variety in $\mathbb{P}^n$. The i^{th} affine patch of V is the set $V_i = \varphi_i^{-1}(V \cap U_i)$. Let V be an affine variety in $\mathbb{A}^n$, and $I = \mathbf{I}(V)$. The i^{th} projective closure of V is the set of common zeroes in $\mathbb{P}^n$ of the polynomials in the ideal $I^* = \langle \ f^* \ : \ f \in I \ \rangle$, and it is denoted $\overline{V}^i$.*

Proposition 1.17. *Fix $0 \leq i \leq n$. If V is an affine variety, its i^{th} projective closure $\overline{V}^i$ is a projective variety, and $\varphi_i(V) = \overline{V}^i \cap U_i$. If V is a non-empty irreducible projective variety, the i^{th} patch V_i of V is an affine algebraic variety and either $\overline{\varphi_i(V_i)}^i = V$, either $V_i = \emptyset$.*

Let V be an affine variety in $\mathbb{A}^n$, and $\overline{V}^i$ its i^{th} projective closure in $\mathbb{P}^n$. Recall that the Zariski topology on $\mathbb{P}^n$ is the topology where the open sets are the complements of the projective varieties (sets of common zeroes between homogeneous polynomials). Hence by definition, U_i is Zariski open, so $V \cap U_i$ is Zariski open for the topology of V induced by $\mathbb{P}^n$. It is well-known that Zariski open sets are either empty or dense. Hence, either $V \cap U_i$ is empty, in which case V is empty too, or Proposition 1.17 says that the Zariski closure of $\varphi_i(V)$ is $\overline{V}^i$. Therefore, in some sense, projective varieties are completion of affine varieties. The points in $\overline{V}^i \cap U_i$ are called *points at infinity* for V.

Example: Let K be an algebraically closed field, and consider the projective variety $\mathscr{V}$ in $\mathbb{P}^2$ defined by the homogeneous polynomial $C(X,Y,Z) = Y^2Z - X^3 - aXZ^2 - bZ^3$, for some $a,b \in K$. The 3^{rd} affine patch is the affine variety V in $\mathbb{A}^2$ defined by the equation $c(x,y) = C(x,y,1) = y^2 - x^3 - ax - b$. We verify that $\overline{V}^3 = \mathscr{V}$, and that it has a single point at infinity $\mathscr{O} = [0:1:0]$, that is to say $\mathscr{V} \cap \{Z = 0\} = \{\mathscr{O}\}$.

1.2 Dimension and Degrees

This Section is dedicated to the definition of the dimension and degree of a variety. Our approach involves Hilbert Series, hence the (weighted) degree of a polynomial is first recalled. Then, we give useful and well-known properties to compute Hilbert Series. The coordinate ring of a variety is defined next, and its Krull dimension is linked to the Hilbert Series. This allows us to give several equivalent definitions for the *dimension* of a variety, that are used in both this Chapter and the following one. Thanks to the Hilbert Series, the *(weighted) degree* of a variety can also be defined. It measures the number of solutions of the systems that we consider, and hence it is key to estimate the efficiency and improvements of our new approaches in Chapters 5 and 6.

1.2.1 The Hilbert Series of an Ideal

We first recall the definitions of monomials and (weighted) degree of a multivariate polynomial. Those classic notions are needed for the definition of both the Hilbert Series and Gröbner bases.

Definition 1.18. *A monomial in $K[X_1,\dots,X_n]$ is a polynomial of the form $X_1^{\alpha_1}X_2^{\alpha_2}\dots X_n^{\alpha_n}$, with $\alpha_1,\dots,\alpha_n$ nonnegative integers. If we let $\alpha = (\alpha_1,\dots,\alpha_n)$ we also write $X_1^{\alpha_1}\dots X_n^{\alpha_n} = \mathbf{X}^\alpha$.*

Any polynomial $f \in K[X_1,\dots,X_n]$ can be written as a linear combination of monomials $f = \sum_{\alpha\in\mathbb{N}^n} c_\alpha \mathbf{X}^\alpha$, with $c_\alpha \in K$ and finitely many $c_\alpha \neq 0$. When $c_\alpha \neq 0$, the polynomial $c_\alpha\mathbf{X}^\alpha$ is called a term of f, and $\mathbf{X}^\alpha$ a monomial of f. The set of all monomials of f is called the support of f, and is denoted by $Supp(f)$.

Let $|\alpha| = \sum_{i=1}^n \alpha_i$. The total degree of f is $\deg f = max\{|\alpha| \ : \ \mathbf{X}^\alpha \in Supp(f)\}$. A polynomial f is said to be homogeneous of degree d when all its monomials have the same degree d.

This definition of homogeneous polynomials is equivalent to Definition 1.14 when $d \neq 0$. By convention, we let $\deg 0 = -1$.

Definition 1.19. *Let $d \in \mathbb{N}$. The set of all homogeneous polynomials of degree d in $K[\mathbf{X}]$ is denoted by $K[\mathbf{X}]_d$. For a homogeneous ideal $I \subset K[\mathbf{X}]$, we denote by I_d the set $I \cap K[\mathbf{X}]_d$.*

The sets in Definition 1.19 are linear spaces of finite dimension. Indeed, $K[\mathbf{X}]_d$ is generated by the monomials of degree d. Hence it makes sense to consider the dimension of the

quotient spaces $K[\mathbf{X}]_d/I_d$. When I is radical, such spaces give a grading of the coordinate ring (Definition 1.24) of the projective variety $\mathbf{V}(I)$, that is to say:

$$K[\mathbf{X}]/I = \bigoplus_{d\geq 0} K[\mathbf{X}]_d/I_d. \tag{1.1}$$

All this information can be encoded using the Hilbert Series.

Definition 1.20 (Hilbert Series of a homogeneous ideal)**.** *Let I be a homogeneous ideal in $K[\mathbf{X}]$. The Hilbert Series of I is the power series $HS_I(T)$ generating the number of monomials of degree d in the quotient algebra:*

$$HS_I(T) = \sum_{d\geq 0} (\dim_K K[\mathbf{X}]_d/I_d)T^d.$$

When I is non-homogeneous, the Hilbert Series of I is the Hilbert Series of its homogenization I^*.

The next Propositions are used to compute the Hilbert Series.

Proposition 1.21 ([Spa12], Prop. 1.40, p.34)**.** *The Hilbert Series of $\langle 0\rangle$ is:*

$$HS_{\langle 0\rangle}(T) = \frac{1}{(1-T)^n}.$$

Proposition 1.22 ([Spa12], Prop. 1.41, p.35)**.** *Let I be a homogeneous ideal in $K[\mathbf{X}]$ and f be a homogeneous polynomial of degree d. If f does not divide zero in $K[\mathbf{X}]/I$, then we have:*

$$HS_{I+\langle f\rangle}(T) = (1-T^d)HS_I(T).$$

The Hilbert Series can always be represented as a rational function, from which some properties of the associated variety can be read.

Proposition 1.23 (Hilbert-Serre)**.** *Let I be a homogeneous ideal in $K[\mathbf{X}]$. There is a polynomial $P(T)\in\mathbb{Z}[T]$ and a minimal integer $0\leq D\leq n$ such that:*

$$HS_I(T) = \frac{P(T)}{(1-T)^D}.$$

with $P(1)\neq 0$ if $D>0$.

The form of the Hilbert Series given by Proposition 1.23 is sometimes called the *irreducible form*.

1.2.2 Dimension of varieties

In this Section, we fix a field K and denote its algebraic closure by $\overline{K}$. The coordinate ring of a variety and its fraction field (Definition 1.24) are important objects for the study of Algebraic Curves in Chapter 2.

Definition 1.24. *Let V be an irreducible affine variety and $I=\mathbf{I}(V)$ its associated ideal in $\overline{K}[X_1,\ldots,X_n]$. The coordinate ring of V is the quotient ring $\overline{K}[X_1,\ldots,X_n]/I$, and it is denoted by $\overline{K}[V]$. The function field of V is the fraction field of $\overline{K}[V]$, and is denoted by $\overline{K}(V)$.*
If V is a projective (irreducible) variety, let V_0 be any non-empty affine patch for V. The coordinate ring, resp. the function field of V is defined as $\overline{K}[V_0]$, resp. $\overline{K}(V_0)$.

If V is defined over K, then every notion can be defined with K instead of $\overline{K}$. It can be shown that two (non-empty) affine patches lead to canonically isomorphic function fields, so that Definition 1.24 is consistent.

The irreducible form of an Hilbert Series can be linked to the coordinate ring of $\mathbf{V}(I)$. Recall that the Krull dimension of a ring A is the length of the maximal increasing chain of distinct prime ideals. In this manuscript, we sometimes write "dimension of an ideal I" and use the notation $\dim I$ when we mean the Krull dimension of the quotient $K[\mathbf{X}]/I$.

Proposition 1.25 ([Kem11], Thm. 11.13, p.58)**.** *Let I be a homogeneous ideal in $K[\mathbf{X}]$ with Hilbert Series in irreducible form $HS_I(T) = \frac{P(T)}{(1-T)^D}$. Then D equals the Krull dimension of I. If I is not homogeneous, then $D = \dim(I) + 1$.*

If I is a homogeneous ideal such that $K[\mathbf{X}]/I$ has Krull dimension 0, its Hilbert Series is a polynomial. We are now ready to give definitions of the dimension of an algebraic variety. The equivalence of every statement is well-known, see for example [CLO97, Chap. 9, §5].

Definition 1.26. *Let $V = \mathbf{V}(I)$ be an affine irreducible variety defined over K. The dimension of V is either:*

- *the length of the maximal increasing chain of irreducible distinct subvarieties of V.*
- *the Krull dimension of $K[V]$.*
- *the transcendence degree of $K(V)$ over K.*
- *the degree of the denominator of the irreducible form of HS_I minus one.*

If V is not irreducible, $\dim V$ is defined as the maximum of the dimension of its irreducible components. If V is projective, then $\dim V$ is the dimension of any non-empty affine patch of V. By convention $\dim \emptyset = -1$.

In particular, if $\mathbf{V}(I)$ is projective, then $\mathrm{HS}_I(T) = \frac{P(T)}{(1-T)^{\dim \mathbf{V}(I)+1}}$.

Zero-dimensional varieties are important both for this thesis (see Decomposition attacks in Chapter 3) and practical applications. The following proposition gives useful characterizations for our applications.

Proposition 1.27 ([CLO97], Chap. 5, §5)**.** *Let $V = \mathbf{V}(I)$ be an affine variety defined over a field K, with algebraic closure $\overline{K}$. Then the following statements are equivalent:*

1. $\dim V = 0$
2. $\dim_K K[\mathbf{X}]/I$ *is finite (as a K-linear space).*
3. $\#V_{\overline{K}}(I)$ *is finite.*

In this case, we have $\#V_{\overline{K}}(I) = \dim_K K[\mathbf{X}]/\sqrt{I}$ and we define the degree of V as:

$$\deg V = \#V_{\overline{K}}(I).$$

Throughout this work, when we are interested in polynomial system solving, we sometimes refer to ideals (and varieties) of dimension 0 as zero-dimensional systems.

1.2.3 Degrees of varieties

Evaluating the numerator of the Hilbert Series at 1 counts the number of points in $\mathbf{V}(I)$ cut by D "generic" hyperplanes. To see why, consider an irreducible projective variety V of dimension D, with associated ideal I in $K[\mathbf{X}]$. From Proposition 1.23, there is a polynomial $P(T)$ such that $\mathrm{HS}_I(T) = \frac{P(T)}{(1-T)^{D+1}}$. Assume that V is not contained in the hyperplane given by a linear form f. From Proposition 1.22 we infer $\mathrm{HS}_{I+\langle f\rangle}(T) = \frac{P(T)}{(1-T)^D}$, so that the numerator remains unchanged. If we now assume that we can find D such hyperplanes $H_i = \mathbf{V}(f_i)$, the homogeneous ideal $I + \langle f_1, \ldots, f_D\rangle$ has dimension 1, and generates the 0-dimensional projective variety $V \cap H_1 \cap \cdots \cap H_D$. The numerator of the Hilbert Series is still $P(T)$, thus $P(1) = \#(V \cap H_1 \cap \cdots \cap H_D)$. This prompts the next definition.

Definition 1.28 (Degree of a variety)**.** *Let I be an ideal with Hilbert Series in irreducible form $\frac{P(T)}{(1-T)^D}$. The degree of I is defined as $\deg I = P(1)$. If V is a variety, the degree of V is defined as $\deg V = \deg \mathbf{I}(V)$.*

Since the polynomial P has integer coefficients, the degree is always an integer. It can be shown it is never negative.

Example: Let f_1, f_2 be homogeneous polynomials of degree $d_1, d_2 \geq 1$ in $K[x,y,z]$, and consider the ideal $I = \langle f_1, f_2\rangle$. From Proposition 1.21 we have $\mathrm{HS}_{K[x,y,z]}(T) = \frac{1}{(1-T)^3}$. Using Proposition 1.22, we find the polynomial $P_1(T) = \sum_{i=0}^{d_1-1} T^i$ such that:

$$\mathrm{HS}_{\langle f_1\rangle} = \frac{1-T^{d_1}}{(1-T)^3} = \frac{P_1(T)}{(1-T)^2}.$$

The variety $\mathbf{V}(f_1)$ is a projective hypersurface and has (projective) dimension 1 — notice that its associated homogeneous ideal has Krull dimension 2, which is coherent with Definition 1.26. Such varieties are called *algebraic curves* and are the topic of Chapter 2. The curve $\mathbf{V}(f_1)$ has degree $P_1(1) = d_1$ as a variety, which can be expected since it is generated by a single polynomial of degree d_1. We now add the polynomial f_2 to $\langle f_1\rangle$. Geometrically we look at the intersection of two curves. Thus, it should have dimension 0 in general (or in other words, be a finite set of points), but it can happen that f_2 is a factor of f_1. In geometric words, f_2 could be a branch of the curve f_1, and algebraically, f_2 could be a divisor of zero in $K[x,y]/\langle f_1\rangle$, which is precisely when Proposition 1.22 fails. Let's assume it is not the case, so that we find a polynomial $P_2 = P_1(T)(\sum_{i=0}^{d_2-1} T^i)$ with

$$\mathrm{HS}_I = \frac{(1-T^{d_2})P_1(T)}{(1-T)^2} = \frac{P_2(T)}{1-T}.$$

Thus $\mathbf{V}(I)$ has (projective) dimension 0, as the geometric intuition suggested. We observe that $\deg I = P_2(1) = d_1 d_2$, wich is confirmed by Bezout's theorem: two projective curves of degree respectively d_1 and d_2 intersect at exactly $d_1 d_2$ points (counted with multiplicities), eventually at infinity. With a slight abuse of names and notations, the Hilbert Series for the underlying affine variety is the polynomial $P_2(T)$.

We now list the properties of the degree that we shall need in this thesis. First, the degree of a 0-dimensional ideal can be interpreted as the dimension of the quotient algebra.

Proposition 1.29. *Let I be a 0-dimensional homogeneous ideal. Then $\dim_K K[\mathbf{X}]/I$ is finite and $\dim_K K[\mathbf{X}]/I = \deg I$.*

Proof. We already observed that $\mathrm{HS}_I(T) = P(T)$ is a polynomial when $\dim I = 0$. Then by definition, $\deg I = P(1) = \sum_{d=0}^{\deg P} \dim_K K[\mathbf{X}]_d/I_d = \dim_K K[\mathbf{X}]/I$. □

The Hilbert Series is multiplicative with respect to tensor products, that is to say, if $K[\mathbf{X}]/I \simeq K[\mathbf{X}_1]/I_1 \otimes K[\mathbf{X}_2]/I_2$, then $\mathrm{HS}_I(t) = \mathrm{HS}_{I_1}(t) \times \mathrm{HS}_{I_2}(t)$. An useful example for our applications is the case of cartesian products of varieties. Let V_1, V_2 be two algebraic varieties. Then we can show that $K[V_1 \times V_2] \simeq K[V_1] \otimes K[V_2]$ — informally, any coordinate function on the cartesian product comes from a pair of coordinate functions from V_1 and V_2. The next result is then a consequence of the definition of degree and the multiplicative behaviour of the Hilbert Series wrt. tensor products.

Proposition 1.30. *Let V_1, V_2 be two algebraic varieties. Then* $\deg(V_1 \times V_2) = \deg V_1 \times \deg V_2$.

Lastly, in Chapter 6 we will need that the degree is decreasing, in the sense of the next Proposition.

Proposition 1.31. *Let $J \subset I$ be two ideals in $K[\mathbf{X}]$, such that* $\dim I = \dim J$. *Then* $\deg I \leq \deg J$.

Proof. Up to homogenization, we can take I and J homogeneous. First assume that I and J are both 0-dimensional, with respective Hilbert Series being the polynomials $\mathrm{HS}_I(T)$ and $\mathrm{HS}_J(T)$. Then we have $\deg I = HS_I(1) = \dim K[\mathbf{X}]/I \leq \dim K[\mathbf{X}]/J = \mathrm{HS}_J(1) = \deg J$. Assume now that $\dim I = \dim J = D > 0$. The observation before Definition 1.28 shows that the numerator of the Hilbert Series is unchanged by adding a linear form f to I, provided f does not divide zero in $K[\mathbf{X}]/I$ and in this situation, the dimension of the ideal is decreased by 1 — see also [CLO97, Chap.9 §5]. Let's assume for the moment that we can find $f_1, \ldots, f_D$ such linear forms. In particular, they are not divisors of zero in J. Let $I_0 = I + \langle f_1, \ldots, f_D \rangle$ and $J_0 = J + \langle f_1, \ldots, f_D \rangle$. Then $\dim I_0 = \dim J_0 = 0$, $J_0 \subset I_0$ and $\mathrm{HS}_{I_0}(T)$ resp. $\mathrm{HS}_{J_0}(T)$ is the numerator of $\mathrm{HS}_I(T)$ resp. $\mathrm{HS}_J(T)$. The result follows.

Now we show that we can find such $f_1, \ldots, f_D$. Let $P_1, \ldots, P_k$ be a minimal decomposition of I [CLO97, Chap. 4,§5, p. 208-209] in prime ideals. In particular, $P_i \not\subset P_j$ for all $i \neq j$. Let also V be the linear space of all linear forms. If $V \subset P_i$ for some i, then the maximal ideal $\langle X_1, \ldots, X_n \rangle$ is contained in P_i, so that $I = P_i$. But this means that $\dim I = 0$, which contradicts our hypothesis. We can proceed inductively to find $f_1, \ldots, f_D$ linear forms that do not divide 0 in $I + \langle f_1 \rangle, I + \langle f_1, f_2 \rangle$,... □

Weighted structures There are situations where natural weights appear on the variables. Exploiting weights has several benefits: first, the theoretical complexity of computing a Gröbner basis for a weighted homogeneous system can be sharpened. Second, running time for practical computations are faster, predicted by complexity estimates — see Section 1.3.2 for more details on the complexity of computing a Gröbner basis. Lastly, the output is usually more "homogeneous". Such weight systems are explicitly exploited in Chapter 6, and we also use adequate systems of weights as computational strategies throughout our experiments in Sections 5.3 and 5.4.2. See also [FSEDV16] for other practical applications.

In all this paragraph, we denote by $w = (w_1, \ldots, w_n)$ a system of weights over $K[\mathbf{X}]$, unless stated otherwise. We only consider integer weights in this work.

Definition 1.32. *For a monomial $\mathbf{X}^\alpha$, the weighted degree of $\mathbf{X}^\alpha$ is denoted by $|\alpha|_w$ and is equal to $|\alpha|_w = w_1\alpha_1 + \cdots + w_n\alpha_n$. The weighted total degree of f is* $\deg_w f = max\{|\alpha|_w : \mathbf{X}^\alpha \in Supp(f)\}$.

A polynomial f is said to be weighted homogeneous of weighted degree d when all its monomials have the same weighted degree d. An ideal is weighted homogeneous when it admits a weighted homogeneous generating set.

Remark 1.33. *If $w_1 = \cdots = w_n = 1$, then the weigthed degree is the total degree of monomials and polynomials as in Definition 1.18.*

When it is clear from the context that weights are involved, we may write "homogeneous" instead of "weighted homogeneous". As in the beginning of the Section, the set of all homogeneous polynomials of weighted degree d is denoted by $K[\mathbf{X}]_d^{(w)}$ and for any ideal I in $K[\mathbf{X}]$, we write $I_d^{(w)} = I \cap K[\mathbf{X}]_d^{(w)}$.

Definition 1.34 (Hilbert Series for weighted ideals)**.** *Let I be a weighted homogeneous ideal in $K[\mathbf{X}]$. The weighted Hilbert Series of I is the power series:*

$$HS_I^{(w)}(T) = \sum_{d\geq 0} (\dim_K K[\mathbf{X}]_d^{(w)}/I_d^{(w)})T^d.$$

In other words, the weighted Hilbert Series is the generating power series of the number of monomials of weighted degree. The weighted structure acts on the shape of the Hilbert Series for the whole algebra, but results similar to those of the previous paragraphs can be obtained.

Proposition 1.35 ([Spa12], Prop. 1.40, 41, 42, p.34-36)**.** *The weighted Hilbert Series of $\langle 0\rangle$ is:*

$$HS_{\langle 0\rangle}^{(w)}(T) = \frac{1}{\prod_{i=1}^{n}(1-T^{w_i})}.$$

Let I be a weighted homogeneous ideal in $K[\mathbf{X}]$ and f be a weighted homogeneous polynomial of degree d. If f does not divide zero in $K[\mathbf{X}]/I$, then we have:

$$HS_{I+\langle f\rangle}^{(w)}(T) = (1-T^d)HS_I(T).$$

Moreover, there is a polynomial $P(T) \in \mathbb{Z}[T]$ such that:

$$HS_I(T) = \frac{P(T)}{\prod_{i=1}^{n}(1-T^{w_i})}.$$

Next we define the weigthed degree of an ideal and a variety.

Definition 1.36 (Weighted degree of a variety)**.** *Let I be an ideal in $K[\mathbf{X}]$ equipped with weights $w = (w_1,\ldots,w_n)$, and $d = \dim I$. Let $Q(T) = (1-T)^d HS_I^{(w)}(T)$. The weigthed degree of I is*

$$\deg_w I = Q(1).$$

The weighted degree of a variety is $\deg_w V = \deg_w \mathbf{I}(V)$.

The weighted degree can be a rational, see the example at the end of this Section. The next proposition illustrates the impact of the weights on the Hilbert Series for ideals, and allows to relate the weighted degree and the standard degree. It is fundamental in our contributions of Chapter 6.

Proposition 1.37 (Computation of the weighted degree, [Ver16], Prop. 3.10)**.** *Consider the injective homomorphism of graded algebras*

$$\begin{array}{rccc} \varphi: & (K[Y_1,\ldots,Y_n],w) & \longrightarrow & (K[X_1,\ldots,X_n],(1,\ldots,1)) \\ & Y_i & \longmapsto & X_i^{w_i}. \end{array}$$

Let I be an ideal in $K[Y_1,\ldots,Y_n]$. Then we have

$$\deg_w I = \frac{\deg \varphi(I)}{\prod_{i=1}^{n} w_i}.$$

Example: Consider the weighted algebra $K[x,y]$ where $\deg x = 1$ and $\deg y = 2$, and the algebra $K[X,Y]$ with standard weights. Let also $\varphi(x) = X$, $\varphi(y) = Y^2$. The variety $\mathbf{V}(x)$ corresponds to the line of the x-axis, and has degree $\deg_w \mathbf{V}(x) = \deg\langle X\rangle/2 = 1/2$. Analogously, observe that the line $\mathbf{V}(y)$ has weighted degree 1. The intersection is the origin, and is associated to the radical ideal $\langle x,y\rangle$. We calculate $\mathrm{HS}_{\langle X,Y^2\rangle}(T) = 1+T$ so that the image ideal has degree 2, and we get $\deg_w \mathbf{V}(x,y) = 1$.

1.3 Gröbner Bases and their algorithmic

When we work with an algebraic variety, it is usually described by polynomial equations fitting the geometric intuition. However, those equations are rarely suited to any of our goals (computing projections, solving zero-dimensional systems). If we replace them by suitable polynomial combinations, the ideal (or variety) remains unchanged, but the new equations allow us to perform the task we want. Several choices can be made for such combinations; to decide how to combine the polynomials, a *monomial order* must be chosen, that is to say, a way of sorting the monomials of the algebra. Then, the main algebraic properties of an ideal, and hence the geometric properties of the associated variety, can be read on the leading ("greatest") monomials of the ideal. *Gröbner Bases* of an ideal are generating sets of polynomials which contain all the necessary leading monomials for a given order. Section 1.3.1 gives the definitions and properties that we need for our contributions.

Measuring the efficiency of the harvesting in Decomposition attacks can be done by estimating the degree of the 0-dimensional systems arising from the algebraic modelling. Classically, 0-dimensional systems over finite fields are solved in three steps: first a Gröbner basis for a degree order is computed. Second, a change of monomial ordering is done to obtain a basis for an elimination order. Lastly, the system is solved. Before our work, the second step was the bottleneck of the solving process in Decomposition attacks. In Chapter 6 we improve this step in even characteristic. In Sections 1.3.2 and 1.3.3, we detail all the steps in the strategy, together with algorithms and complexity estimates.

Computing degree order Gröbner bases is our first focus. In his thesis [Buc65, Buc06] Buchberger gave an algorithm to compute a Gröbner Basis for an ideal from a set of generators. While a proof of termination is given, a complexity analysis of the algorithm was complicated. The work of Lazard [Laz83] highlighted the link between the computation of Gröbner Bases and linear algebra by performing Gaussian Elimination on *Macaulay matrices*. This framework leads to the efficient algorithms F4 and F5 [Fau99, Fau02] to compute Gröbner Bases for total degree orders, informally presented in Section 1.3.2. We also give complexity estimates (Theorem 1.53 [BFS14]) depending on the *degree of regularity* (Definition 1.46). Those estimates are used to analyze the harvesting in Decomposition attacks in Section 3.4.2.

Total degree orders are not suited for resolution of 0-dimensional systems. Bases in lexicographical orders are on the contrary fit for solving: we explain why starting Section 1.3.3. However, a direct computation is usually intractable with current algorithms for any meaningful experiments and practical applications. We recall the best known strategy for the resolution of zero-dimensional systems, relying on a change of monomial ordering from a degree order to a lexicographical order using FGLM's algorithm [FGLM93] — more recently, its sparse variant [FM11]. Whenever we need to solve a zero-dimensional system in this thesis, we always use this strategy. FGLM's complexity depends on the degree of the system (also the number of solutions, Proposition 1.27) and is stated in Proposition 1.58. It is the main indicator of the efficiency of the harvesting in Decomposition attacks, as it usually dominates the solving process of the systems in this situation.

1.3.1 Monomial orders and Gröbner Bases

Definition 1.38 (Monomial Ordering)**.** *A monomial ordering* $>_{\mathtt{m}}$ *on* $\mathbb{N}^n$ *is a relation satisfying the following properties:*

1. $>_{\mathtt{m}}$ *is a total ordering on* $\mathbb{N}^n$*.*
2. *If* $\alpha >_{\mathtt{m}} \beta$*, then for all* $\gamma \in \mathbb{N}^n$*,* $\alpha + \gamma >_{\mathtt{m}} \beta + \gamma$*.*

3. $>_{\mathfrak{m}}$ *is well-ordering on* $\mathbb{N}^n$*, which means that every non-empty subset of* $\mathbb{N}^n$ *has a smallest element.*

Given a monomial ordering $>_{\mathfrak{m}}$*, we say that* $\mathbf{X}^{\alpha} >_{\mathfrak{m}} \mathbf{X}^{\beta}$ *if and only if* $\alpha >_{\mathfrak{m}} \beta$*.*

Thanks to the second condition, a monomial order is compatible with the multiplication of monomials: $\mathbf{X}^{\alpha}\mathbf{X}^{\beta} >_{\mathfrak{m}} \mathbf{X}^{\alpha}\mathbf{X}^{\gamma}$ if and only if $\mathbf{X}^{\beta} >_{\mathfrak{m}} \mathbf{X}^{\gamma}$. Using a monomial ordering we can define leading monomial, term and coefficient of a polynomial.

Definition 1.39. *Let* $f = \sum c_{\alpha}\mathbf{X}^{\alpha}$ *in* $K[X_1,\ldots,X_n]$*, and fix a monomial order* $>_{\mathfrak{m}}$*.*

- *The leading monomial* $LM(f)$ *of* f *is the greatest monomial with respect to* $>_{\mathfrak{m}}$ *in* $Supp(f)$*.*
- *The leading coefficient* $LC(f)$ *is the coefficient of* $LM(f)$*.*
- *The leading term* $LT(f)$ *is the polynomial* $LC(f)\cdot LM(f)$*.*

We now introduce classical monomial orders.

Definition 1.40. *Let* $\alpha,\beta \in \mathbb{N}^n$*.*

- **Lexicographical Order:** *We say that* $\alpha >_{lex} \beta$ *if there is* $1 \le i_0 \le n$ *such that* $\alpha_{i_0} > \beta_{i_0}$ *and* $\alpha_i = \beta_i$ *for all* $1 \le i < i_0$*. Equivalently, the leftmost non-zero entry in the vector* $\alpha - \beta$ *is positive.*
- **Degree Reverse Lex Order (DRL):** *We say that* $\alpha >_{DRL} \beta$ *if* $|\alpha| > |\beta|$ *or* $|\alpha| = |\beta|$ *and the rightmost non-zero entry in* $\alpha - \beta$ *is negative.*
- **Elimination Order:** *Let* $1 \le k \le n$*, and fix two monomials orders* $>_1$ *and* $>_2$*.*
  We say that $\alpha >_{k-elim} \beta$ *if* $(\alpha_1,\ldots,\alpha_k) >_1 (\beta_1,\ldots,\beta_k)$ *or* $(\alpha_1,\ldots,\alpha_k) =_1 (\beta_1,\ldots,\beta_k)$ *and* $(\alpha_{k+1},\ldots,\alpha_n) >_2 (\beta_{k+1},\ldots,\beta_n)$*.*
- **Weighted Degree Reverse Lex Order (w-DRL)** *Fix a weight vector* $w \in \mathbb{N}^n$*, and let* $\alpha,\beta \in \mathbb{N}^n$*. The Weighted Degree Reverse Lex Order (w-DRL) is defined as* $\alpha >_{w-DRL} \beta$ *if* $|\alpha|_w > |\beta|_w$ *or* $|\alpha|_w = |\beta|_w$ *and the rightmost non-zero entry in* $\alpha - \beta$ *is negative.*

Examples: Let $f = 4xy^2z + 4z^2 - 5x^3 + 7x^2z^2$ in $K[x,y,z]$.

- In lex order, rewriting f with monomials in decreasing order gives $f = -5x^3 + 7x^2z^2 + 4xy^2z + 4z^2$. In this case, $\mathrm{LT}(f) = -5x^3$, $\mathrm{LC}(f) = -5$, $\mathrm{LM}(f) = x^3$, and $\deg f = 3$.
- In DRL order, we have $f = 4xy^2z + 7x^2z^2 - 5x^3 + 4z^2$ with $\mathrm{LT}(f) = 4xy^2z$, $\mathrm{LC}(f) = 4$, $\mathrm{LM}(f) = xy^2z$ and $\deg f = 4$.
- Consider f in $K[z,x,y]$ for the 1-Elimination order. Then $f = 7z^2x^2 + 4z^2 + 4zxy^2 - 5x^3$ with $\mathrm{LT}(f) = 7x^2z^2$, $\mathrm{LC}(f) = 7$ and $\mathrm{LM}(f) = x^2z^2$ and $\deg f = 4$.
- Let $w = (2,1,2)$. In w-DRL order we have $f = 7x^2z^2 - 5x^3 + 4xy^2z + 4z^2$ with $\mathrm{LT}(f) = 7x^2z^2$, $\mathrm{LC}(f) = 7$ and $\mathrm{LM}(f) = x^2z^2$ and $\deg_w f = 8$.

Elimination orders are also called Block orders, a terminology that stems from the fact that monomials are first compared for $>_1$ on a first block of variables, then with $>_2$ on the rest of the variables. The term "Elimination" comes from the observation that any monomial involving at least one variable from the first set is always greater than a monomial involving only variables from the second set. Thus for an elimination order, the smaller elements in a given set of polynomial may not involve variables of the first block, that have been "eliminated". It is possible to consider more than two blocks of variables. In particular, the lexicographical order is an elimination order with as many blocks as there are variables. When a monomial order is chosen over $K[X_1,\ldots,X_n]$ and no more details are given, then the variables are ordered as $X_1 > X_2 > \ldots > X_n$.

Gröbner Bases Gröbner bases are generating sets of polynomials with convenient properties, which depend on the monomial order: bases in total degree orders are interesting for computation purpose, while eliminations orders are better to solve the underlying system of equations. They are also our main computational tools. For any set S in $K[X_1,\dots,X_n]$ and a monomial order, let $\mathrm{LT}(S) = \{\mathrm{LT}(f) \ : \ f \in S\}$.

Definition 1.41 (Gröbner Bases)**.** *Let I be an ideal in $K[X_1,\dots,X_n]$ and fix a monomial order.*

- *The initial ideal of I is $\langle LT(I)\rangle$.*
- *A set $\{g_1,\dots,g_r\} \subset I$ is a Gröbner Basis for I if its set of leading terms generates the initial ideal of I:*
$$\langle LT(g_1),\dots,LT(g_r)\rangle = \langle LT(I)\rangle.$$
- *A Gröbner Basis G for I is said to be reduced if:*
  - *$LC(p) = 1$ for all $p \in G$;*
  - *For all $p \in G$, no monomial of p lies in $\langle LT(G-\{p\})\rangle$, i.e. $Supp(p) \cap \langle LT(G-\{p\})\rangle = \emptyset$.*

Gröbner Bases have several properties, and we list the ones that we need below. The proofs can be found in [CLO97, Chap. 2]. A constructive proof of existence was given first in Buchberger's thesis [Buc65].

Proposition 1.42. *Let I be an ideal of $K[X_1,\dots,X_n]$ and fix a monomial order. A Gröbner Basis $G = \{g_1,\dots,g_r\}$ of I exists, it generates I. Moreover, the ideal I admits a unique reduced Gröbner Basis.*

Gröbner bases give computational answers to several problems in the theory of algebras. Among them is the *Ideal membership* problem.

Definition-Proposition 1.43 (Normal Form, [CLO97], p. 82)**.** *Let I be an ideal in $K[X_1,\dots,X_n]$, and $G = \{g_1,\dots,g_r\}$ be a Gröbner basis for a fixed monomial order. For any polynomial f in $K[X_1,\dots,X_n]$, there exists a unique r in $K[X_1,\dots,X_n]$ such that*

- *No term of r is in $\langle LT(I)\rangle$.*
- *There are $c_1,\dots,c_r$ such that $f = \sum c_i g_i + r$ and $LM(c_i g_i) \leq LM(f)$.*

Moreover, $r = 0$ if and only if $f \in I$. The polynomial r is called the Normal Form *of f wrt. G.*

The normal form gives a way of representing elements in the quotient algebra $K[X_1,\dots,X_n]/I$. Hence Gröbner bases allow practical computations in the coordinate rings of algebraic varieties (see Definition 1.24). It is also used by FGLM's algorithm described in Section 1.3.3.

1.3.2 Computing Gröbner Bases with linear algebra

The case of linear systems The link between linear algebra and Gröbner bases is better understood if we first go back to linear systems. Indeed, they are particular cases of polynomial systems where all equations have degree 1. Consider the following linear system:

$$\begin{aligned} a_{1,1}X_1 + \cdots + a_{1,n}X_n &= 0,\\ &\vdots\\ a_{m,1}X_1 + \cdots + a_{m,n}X_n &= 0. \end{aligned}$$

To solve it, we write $\mathbf{A} = (a_{i,j})$ as a $m \times n$ matrix and $\mathbf{X} = (X_1, \ldots, X_n)$ to get $\mathbf{A} \cdot \mathbf{X} = 0$. Gaussian Elimination is then performed on $\mathbf{A}$ to get a triangular form, which means we get an equivalent system with shape

$$\begin{aligned} b_{1,1}X_1 + \cdots + b_{1,d}X_d + \cdots + b_{1,n}X_n &= 0, \\ \ddots & \\ b_{m,d}X_d + \cdots + b_{m,n}X_n &= 0. \end{aligned}$$

In this triangular shape the system can be solved starting from the bottom equation and going up until all equations have been used.

Macaulay Matrices This idea can be generalized to a polynomial system with equations of distinct degrees greater than 1. Let $f_1, \ldots, f_s$ be homogeneous polynomials with $\deg f_i = d_i$, and $I = \langle f_1, \ldots, f_s \rangle$. Let $\mathscr{M}_{d,n}$ be the set of all monomials of degree at most d in $K[X_1, \ldots, X_n]$. All polynomials in I of degree at most d are linear combinations of $\mathfrak{m} f_i$, for all $\mathfrak{m} \in \mathscr{M}_{d-d_i,n}$. This can be rewritten as a matrix-vector product, where the columns of the matrix are indexed by the monomials in $\mathscr{M}_{d,n}$, and the vector is the vector of all $\mathfrak{m} f_i$. This matrix is called the *Macaulay matrix* of degree d.

Definition 1.44 (Macaulay Matrix)**.** *Let $f_1, \ldots, f_s$ be polynomials in $K[X_1, \ldots, X_n]$, with $\deg f_i = d_i$, and fix a monomial ordering $>_\mathfrak{m}$. For a fixed integer d, let $\mathscr{M}_{d,n}$ be the set of all monomial of degrees at most d in $K[X_1, \ldots, X_n]$, sorted by decreasing order with respect to $>_\mathfrak{m}$. Let its cardinal be $r_d = \binom{n+d}{n}$, and let also $R_d = r_{d-d_1} + \cdots + r_{d-d_s}$. The degree d Macaulay Matrix for $>_\mathfrak{m}$ and the f_i's is denoted $Mac_{d,>_\mathfrak{m}}(f_1, \ldots, f_s)$ and lies in $\mathscr{M}_{R_d,r_d}(K)$. Its rows are indexed by the polynomials $\mathfrak{m}_{d-d_i,k} f_i$, for all $k \leq r_{d-d_i}$ and for all $i \leq s$. Its columns are indexed by the monomials in $\mathscr{M}_{d,n}$. The entry corresponding to the row $\mathfrak{m}_{d-d_i,k} f_i$ and column $\mathfrak{m}_{d,j}$ is the coefficient of $\mathfrak{m}_{d,j}$ in $\mathfrak{m}_{d-d_i,k} f_i$.*

Writing $f_i = \sum_\alpha c_\alpha \mathbf{X}^\alpha$ and $\mathfrak{m}_{d-d_i,k} = \mathbf{X}^\beta$, then the Macaulay matrix contains c_α in the column $\mathfrak{m}_{d,j} = \mathbf{X}^{\alpha+\beta}$ and in the line $\mathfrak{m}_{d-d_i,k} f_i$:

$$Mac_{d,>_\mathfrak{m}}(f_1, \ldots, f_s) = \begin{array}{c} \begin{array}{ccccc} \mathfrak{m}_{d,1} & & >_\mathfrak{m} \cdots \mathfrak{m}_{d,j} \cdots & & \mathfrak{m}_{d,r_d} \end{array} \\ \begin{pmatrix} & & \cdots & \cdots & \\ & & & & \\ \vdots & & & & \vdots \\ \vdots & & c_\alpha & & \vdots \\ \vdots & & & & \vdots \\ & & & & \\ & & \cdots & \cdots & \end{pmatrix} \end{array} \begin{array}{l} \mathfrak{m}_{d-d_1,1} f_1 \\ \mathfrak{m}_{d-d_1,2} f_1 \\ \vdots \\ \mathfrak{m}_{d-d_i,k} f_i \\ \vdots \\ \mathfrak{m}_{d-d_s,r_{d-d_s}} f_s \end{array}$$

Example: Let $f_1 = x^2y + 2x^2 + 2y + 4$ and $f_2 = xy - x^2 - 4y^2 + 4x$ in $K[x,y]$. The degree 3 Macaulay Matrix for the DRL order and f_1, f_2 is

$$\mathrm{Mac}_{3,>_{DRL}}(f_1, f_2) = \begin{array}{c} \begin{array}{cccccccccc} x^3 & x^2y & xy^2 & y^3 & x^2 & xy & y^2 & x & y & 1 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 2 & 4 \\ -1 & 1 & -4 & 0 & 4 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & -4 & 0 & 4 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array} \begin{array}{l} f_1 \\ xf_2 \\ yf_2 \end{array}$$

Any linear combination of the rows gives the same linear combination of the associated polynomials. Therefore if we perform Gaussian Elimination *without swapping the columns* on the degree d Macaulay matrix for $f_1, \ldots, f_s$, the rows give a new set of generators with nice properties for the set of all polynomial of degree at most d in the ideal $I = \langle f_1, \ldots, f_s \rangle$. The key result is that for a degree d high enough, a Gröbner Basis for I is obtained this way.

Theorem 1.45 (Lazard, [Laz83]). *Let $f_1,\dots,f_s$ be polynomials in $K[X_1,\dots,X_n]$. There exists a degree D such that the rows of the row echelon form of $Mac_{D,>_\mathfrak{m}}(f_1,\dots,f_s)$ form a Gröbner Basis of $I=\langle f_1,\dots,f_s\rangle$.*

The theorem does not give an expression for the integer D. To have a clear stopping criterion, any algorithm relying on linear algebra to compute Gröbner bases must use additional properties. This is achieved by F4 [Fau99] and F5 [Fau02] algorithms, which use linear algebra on Macaulay matrices to compute Gröbner bases for total degree orders, and rely on additional criteria to terminate and speed-up the computations.

Efficient algorithms: F4 and F5 We first briefly present F4's principle. The overall idea is to combine the classical Buchberger algorithm relying on the notion of *critical pairs* and Gaussian elimination over Macaulay matrices.

In Buchberger's algorithm, iteration is done over critical pairs: one pair is selected and reduced (by multivariate division) at each iteration. The F4 algorithm selects all the critical pairs satisfying a selection criterion, and reduce them all at the same time by computing row-echelon form of their Macaulay matrix. Some computations are stored at each degree, and used to build the next Macaulay matrices. This reduces their size by eliminating useless rows, which should speed-up the Gaussian Elimination. Iteration is done on the degree of the matrices as in Lazard's approach [Laz83], but termination of the algorithm is reached when there are no more critical pairs, as in Buchberger's algorithm. An efficient implementation of F4 exists in Magma [BCP97]: it is the primary tool we used in our experiments. Another efficient library for prime finite fields and rational numbers is given by the FGb [Fau10] Maple package.

A problem in the F4 approach is that, during the computation of the row echelon form, there is a large amount of linear combinations of the rows that reduce to zero. This means no information is deduced from these combinations, and that time is spent for useless computations. A criterion to detect such reductions to zero is given in [Fau02], leading to F5 algorithm. For regular sequences of homogeneous polynomials (Definition 1.48), it is proved that all useless computations are avoided. The complexity can be bounded by the cost of computing the row echelon form for the biggest Macaulay matrix [Bar04]. We now investigate the situation for zero-dimensional ideals and varieties, which are our main interests.

Degree of regularity The degree bound for zero-dimensional homogeneous ideals is called the *degree of regularity*, and it can be estimated for *regular sequences*. In our applications, we mainly need to estimate the complexity of solving affine zero-dimensional systems. The notions can be extended and the complexity is also understood [Bar04]. As in all this thesis, we only consider integer weights.

Definition 1.46 (Degree of regularity). *Let I be a 0-dimensional homogeneous ideal in $K[X_1,\dots,X_n]$, equipped with weights $w=(w_1,\dots,w_n)$. The degree of regularity of I is the smallest integer $d_{reg}^{(w)}(I)\geq 0$ such that the linear combinations of homogeneous polynomials of (weighted) degree d in I generate all the monomials of (weighted) degree d in n variables:*

$$\begin{aligned} d_{reg}^{(w)}(I) &= \min_{d\in\mathbb{N}}\left\{\ d\ :\ \dim_K Span(\ f\ :\ f\in I, \deg_w f = d\ ) = \binom{n+d-1}{d}\ \right\}\\ &= \min_{d\in\mathbb{N}}\{\ d\ :\ \dim_K I_d = \dim_K K[X_1,\dots,X_n]_d\ \}. \end{aligned}$$

When the weights are $(1,\dots,1)$, we also use the notation $d_{reg}(I)$.

Let G be the reduced Gröbner Basis for a total degree order for some zero-dimensional homogeneous ideal I. It can then be shown that the degree of regularity is a bound on the

total degree of the elements in G. Recall that the Hilbert Series for a homogeneous zero-dimensional ideal is a polynomial (Proposition 1.25). This gives a convenient way to read the degree of regularity of an ideal from its Hilbert Series.

Proposition 1.47. *Let I be a 0-dimensional homogeneous ideal in $K[X_1,\dots,X_n]$ equipped with weights $(w_1,\dots,w_n)$. The (weighted) degree of regularity of I is:*

$$d_{reg}^{(w)}(I) = \deg(HS_I(T)) + \max_{1\leq i\leq n}\{w_i\}.$$

Regular sequences Such sequences of polynomials behave particularly well in computations of Gröbner bases, and the degree of regularity of the ideal they define is well-understood (Proposition 1.50).

Definition 1.48. *A sequence $(f_1,\dots,f_r)$ of non-zero homogeneous polynomials in $K[\mathbf{X}]$ is called regular if for all $1\leq i\leq r-1$, f_{i+1} does not divide zero in $K[\mathbf{X}]/\langle f_1,\dots,f_i\rangle$.*

As a direct consequence of Proposition 1.22, they can be characterized by the Hilbert Series of the ideal they generate.

Proposition 1.49. *Let $\mathbf{F}=(f_1,\dots,f_r)$ be a sequence of non-zero homogeneous polynomials in $K[\mathbf{X}]$ with weights $w=(w_1,\dots,w_n)$. The following statements are equivalent:*

- $\mathbf{F}$ *is regular.*
- $HS_{\langle\mathbf{F}\rangle}(T) = \dfrac{\prod_{i=1}^{r}(1-T^{\deg f_i})}{\prod_{i=1}^{n}(1-T^{w_i})}$.
- $\dim\langle\mathbf{F}\rangle = n-r$, $\dim\mathbf{V}(\langle\mathbf{F}\rangle) = n-r-1$.

We are now in a position to give a bound on the degree of regularity. It is used in the complexity analyses of Section 3.4.2.

Proposition 1.50 (Bézout bound, Macaulay bound, [Bar04, Spa12])**.** *Let $(f_1,\dots,f_n)$ be a regular sequence in $K[X_1,\dots,X_n]$, with weights $w=(w_1,\dots,w_n)$. Let $d_i=\deg f_i$ and let $I=\langle f_1,\dots,f_n\rangle$. Then we have:*

- *Bézout bound:* $\deg I = \dfrac{\prod_{i=1}^{n} d_i}{\prod_{i=1}^{n} w_i}$.
- *Macaulay Bound:* $d_{reg}(I) = \max_{1\leq i\leq n}\{w_i\} + \sum_{i=1}^{n}(\deg_w f_i - w_i)$.

Example: For the standard weights, the Bézout bound is the classical product of the degree of the hypersurfaces given by the f_i's, and the Macaulay bound is $1+\sum_{i=1}^{n}(\deg f_i - 1)$.

Affine Ideals All the previous notions can be extended following [Bar04] to non-homogeneous ideals — which we also call affine ideals. Starting this paragraph, we denote by $\tilde{f}$ the homogeneous component of highest degree of any polynomial f, and similarly, if $I=\langle f_1,\dots,f_s\rangle$, we write $\tilde{I}=\langle\tilde{f}_1,\dots,\tilde{f}_s\rangle$.

Definition 1.51 (Affine regular sequence)**.** *A sequence of non-zero non-homogeneous polynomials $(f_1,\dots,f_s)$ in $K[X_1,\dots,X_n]$ is called regular, or affine regular, if $(\tilde{f}_1,\dots,\tilde{f}_s)$ is a regular sequence.*

If an ideal I in $K[X_1,\dots,X_n]$ is generated by an affine regular sequence of n polynomials, then from Proposition 1.49 $\tilde{I}$ is homogeneous and has dimension 0. Thus it makes sense to talk about its degree of regularity, and the next definition is coherent.

Definition 1.52 (Affine degree of regularity)**.** *Let $(f_1,\dots,f_n)$ be an affine regular sequence in $K[X_1,\dots,X_n]$ equipped with weights $w=(w_1,\dots,w_n)$ and let $I=\langle f_1,\dots,f_n\rangle$. Then the affine degree of regularity is defined as $d^{(w)}_{a,reg}(I)=d^{(w)}_{reg}(\tilde{I})$.*

The affine degree of regularity also bounds the size of the biggest Macaulay matrix in Gröbner bases computations when the input system is not homogeneous — see [Bar04]. The fundamental difference with homogeneous system is that *degree falls* can happen during the computation. In other words, polynomials of degree smaller than the current iteration can be obtained by combinations of the lines in the Macaulay matrix. The complexity statement for affine systems does not account for such reductions.

As row-echelon forms are involved in the computation of Gröbner Basis by linear algebra, the complexity also depends on the so-called *matrix multiplication exponent*, traditionally denoted by ω. It is defined as the smallest exponent such that the product of two $N\times N$ matrices can be achieved in $O(N^\omega)$. Classical (and practical) values for this exponent are:

- $\omega=3$ (Schoolbook multiplication)
- $\omega\leq 2.807$ (Strassen's algorithm [Str69])

For completeness we mention that $\omega\leq 2.376$ is reached by the (unpractical) Coppersmith-Winograd's algorithm [CW90]. The recent work of Le Gall [Gal14] reduced the exponent to $\omega\leq 2.3728639$.

We now state the complexity for computing a Gröbner basis of a (weighted) homogeneous ideal for a total (weighted) degree order using F5 algorithm. A bound in the affine case is obtained by replacing n by $n+1$, d_{reg} by $d_{a,reg}$ and with $w_{n+1}=1$. It is used in the complexity analyses of Section 3.4.2.

Theorem 1.53 ([BFS14], [FSEDV16])**.** *Let $I=\langle f_1,\dots,f_n\rangle$ be a homogeneous ideal in $K[X_1,\dots,X_n]$ of dimension 0 in $K[X_1,\dots,X_n]$ equipped with weights $w=(w_1,\dots,w_n)$. Let ω be the exponent for the multiplication of matrices. The complexity in number of field operations to compute a Gröbner Basis of I for the (weighted) DRL order with the F5 algorithm is asymptotically bounded when n goes to infinity by*

$$O\left(nd^{(w)}_{reg}(I)\left(\frac{1}{\prod_{i=1}^n w_i}\binom{d^{(w)}_{reg}(I)+\sum_{i=1}^n w_i-1}{d^{(w)}_{reg}}\right)^\omega\right).$$

When $K[X_1,\dots,X_n]$ is equipped with the standard weights $w_1=\dots=w_n=1$, the complexity is:

$$O\left(nd_{reg}(I)\binom{d_{reg}(I)+n-1}{d_{reg}}^\omega\right).$$

1.3.3 Solving Zero Dimensional Systems

In practice, computing a Gröbner Basis for a degree order (such as DRL order) is easier than for an elimination order. However, Gröbner bases for degree order are not suited for the resolution of the system, while elimination orders, and particularly the lexicographical order, are. This can be seen on the next propositions, which describe the shape of a lexicographical basis as some triangular system.

Proposition 1.54. *Let $I\subset K[X_1,\dots,X_n]$ be a zero-dimensional ideal, and $G=\{g_1,\dots,g_r\}$ a Gröbner basis of I for the lexicographical order $>_{lex}$. Assume the g_i's are ordered decreasingly for $>_{lex}$,* i.e *$LM(g_1)>_{lex}\cdots>_{lex}LM(g_r)$. Then $g_r\in K[X_n]$, and there exists a strictly increasing sequence $1=i_1<i_2<\cdots<i_m=r$ of integers such that, for all $1\leq j\leq m-1$ and all $i_j\leq k\leq$*

$i_{j+1}-1$, $g_k \in K[X_j,\dots,X_n]$ *and* $g_k \notin K[X_{j+1},\dots,X_n]$. *More precisely, the Gröbner basis* G *has the following shape:*

$$G = \left\{ \begin{array}{c} g_1(X_1,\dots,X_n), \\ \vdots \\ g_{i_2-1}(X_1,\dots,X_n), \\ g_{i_2}(X_2,\dots,X_n), \\ \vdots \\ g_{i_3}(X_3,\dots,X_n), \\ \vdots \\ g_{r-1}(X_{n-1},X_n), \\ g_r(X_n) \end{array} \right\}$$

Solving a system in this form is staightforward using the roots of the univariate polynomial g_r. An even more convenient shape of lexicographical basis exists.

Definition 1.55 (Shape position)**.** *A* 0*-dimensional ideal* $I \subset K[X_1,\dots,X_n]$ *is said to be in Shape position if the reduced Gröbner basis* G *for the lexicographical order has the following shape:*

$$G = \left\{ \begin{array}{c} X_1 - g_1(X_n), \\ X_2 - g_2(X_n), \\ \vdots \\ X_{n-1} - g_{n-1}(X_n), \\ g_n(X_n) \end{array} \right\}$$

It has been proved in [BMMT94] that if I is radical and K is large enough, then after a random linear change of variables, the probability that a lexicographical Gröbner basis is in Shape position is overwhelming. From this shape, the degree of the ideal can be read as $\deg I = \deg g_n$ and solving the system is immediate once the roots of g_n have been found.

FGLM's algorithm F4 and F5 algorithms were designed to exploit the structure of the (weigthed) DRL order. In particular, computing directly a lexicographical Gröbner basis is usually harder in practice than computing a degree order basis. For 0-dimensional ideals, an efficient approach was proposed in [FGLM93] with the FGLM change-ordering algorithm. It takes in input a Gröbner basis G_1 for a monomial order $>_1$, a monomial order $>_2$, and outputs a Gröbner basis G_2 for $>_2$. We informally describe the ideal of the algorithm below.

Define the *staircase* of a set of polynomial as the set of the leading monomials of its elements. By Theorem 1.27, the K-algebra $K[X_1,\dots,X_n]/I$ has finite dimension $\deg I$, a basis being given by (the classes of) the monomials of $K[X_1,\dots,X_n]$ "under the staircase" defined by G_1. In other words, the set of all the monomials in $K[X_1,\dots,X_n]$ that are smaller wrt. $>_1$ to all elements in G_1's staircase forms a K-basis of $K[X_1,\dots,X_n]/I$. Informally, the algorithm performs a basis transformation to determine the staircase of G_2. This is done by determining the multiplicative structure of $K[X_1,\dots,X_n]/I$ as a K-algebra, that is to say computing for each i the matrix of the multiplication by X_i in $K[X_1,\dots,X_n]/I$.

Definition 1.56 (Multiplication Matrices)**.** *Let* I *be a zero-dimensional ideal in* $K[X_1,\dots,X_n]$ *and fix a monomial order* $>_{\mathfrak{m}}$. *Let* G *be the unique reduced Gröbner basis of* I *wrt.* $>_{\mathfrak{m}}$, *and let* $D = \deg I = \dim K[X_1,\dots,X_n]/I$. *The* i^{th} *multiplication matrix for* $>_{\mathfrak{m}}$ *is the* $D \times D$ *matrix* $T_{n,>_{\mathfrak{m}}}$ *of the linear map*

$$\begin{array}{rccc} [X_i]: & K[X_1,\dots,X_n]/I & \longrightarrow & K[X_1,\dots,X_n]/I \\ & \bar{f} & \longmapsto & \overline{X_i f} \end{array}$$

in the basis given by the staircase of G, *where* $\bar{f}$ *denotes the normal form of* $f \in K[X_1,\dots,X_n]$ *wrt.* G *(see Definition 1.43).*

The transformation from one basis to another is then obtained by finding linear dependencies between the (classes of the) monomials of $K[X_1,\dots,X_n]$, sorted wrt. $>_2$. The complexity of the algorithm is well-understood (see Proposition 1.58), and mainly depends on the dimension of $K[\mathbf{X}]/I$. When $\dim I = 0$ and I is homogeneous, we have by Proposition 1.29 that $\deg I = \dim_K K[\mathbf{X}]/I$, so the complexity can be expressed using the degree of the ideal. A "weighted" version of this statement can also be obtained.

Proposition 1.57. *Let I be an ideal such that $\dim I = 0$, and fix a weight system $w = (w_1,\dots,w_n)$. Then $\deg_w I = \dim_K K[\mathbf{X}]/I$.*

Proof. On the one hand, we know that $\deg I = \dim_K K[\mathbf{X}]/I$. From [CLO97, Chap. 3, Theorem 4], this dimension is also the cardinal of the set of "monomials under the staircase" of I, that is to say, the cardinal of the set $\mathscr{M} = \{\mathbf{X}^\alpha \ : \ \mathbf{X}^\alpha \notin \langle \mathrm{LT}(I)\rangle\}$. On the other hand, using Proposition 1.37 and its notations, we have

$$\deg_w I = \frac{\deg \varphi(I)}{\prod_{i=1}^n w_i} = \frac{\dim_K K[\mathbf{X}]/\varphi(I)}{\prod_{i=1}^n w_i},$$

and the numerator is also the number of monomials under the staircase of $\varphi(I)$. Hence the wanted equality is obtained if we can prove that $\mathscr{M}' = \{\mathbf{X}^\alpha \ : \ \mathbf{X}^\alpha \notin \langle \mathrm{LT}(\varphi(I))\rangle\}$ has $\prod_{i=1}^n w_i \times \#\mathscr{M}$ elements.

The cardinality of the staircase does not depend on any choice of a (reduced) Gröbner basis for any order, so we can consider the staircase given by G, a Gröbner basis for I for the w-DRL order, and $\mathscr{M} = \{\mathbf{X}^\alpha \ : \ \mathbf{X}^\alpha <_{w-DRL} \mathrm{LT}(g) \ \forall \ g \in G\}$. As $\varphi(G)$ is a Gröbner basis for $\varphi(I)$ for the DRL order [Ver16, p. 94], then $\mathscr{M}' = \{\mathbf{X}^\alpha \ : \ \mathbf{X}^\alpha <_{DRL} \mathrm{LT}(\varphi(g)) \ \forall \ g \in G\}$, where $\mathrm{LT}(\varphi(g)) = \mathrm{LT}(g)^w = X_1^{\alpha_1 w_1}\dots X_n^{\alpha_n w_n}$, and the result follows. □

Informally, the above proof essentially says that going through the homomorphism φ dilates the volume of the staircase by $\prod_{i=1}^n w_i$. We can now state the complexity of FGLM's algorithm.

Proposition 1.58 ([FGLM93], [FGHR], [Ver16])**.** *Let I be a zero-dimensional ideal in $K[X_1,\dots,X_n]$, and G_1 be a Gröbner basis for a monomial order $>_1$. If ω is the matrix multiplication exponent, then a Gröbner basis G_2 for a monomial order $>_2$ can be computed from G_1 in a number of field operations bounded by*

$$O(n \deg I^\omega).$$

If $K[X_1,\dots,X_n]$ is equipped with a weight system $w = (w_1,\dots,w_n)$, then the number of field operations can be bounded by

$$O(n \deg_w I^3) = O\left(n \left(\frac{\deg \varphi(I)}{\prod_{i=1}^n w_i}\right)^3\right),$$

where φ is the homomorphism defined in Proposition 1.37.

For solving 0-dimensional system $\mathscr{S}$, $>_1$ is usually a (weighted) DRL order and $>_2$ is a lexicographical order. It is conjectured that the ω exponent can also be considered for weighted structures. Indeed, the way this exponent is obtained in [FGHR] does not seem to depend on the presence of weights or not.

From [BMMT94] we can assume that the ideal generated by $\mathscr{S}$ is in Shape Position with basis $G_2 = \{X_1 - g_1(X_n),\dots,X_{n-1} - g_{n-1}(X_n), g_n(X_n)\}$. In this context, a very efficient change-ordering algorithm has been proposed in [FM11] taking advantage of the sparsity of the multiplication matrices and Wiedemann's sparse linear algebra algorithm [Wie86]. It does not improve on the asymptotic complexity given in Proposition 1.58, but is far more efficient

in practice than the original approach. We give a brief and informal description.

The polynomial g_n is in fact the minimal polynomial of the matrix $T_{n,>1}$ of multiplication by X_n in $K[X_1,\dots,X_n]/\mathscr{S}$. Using adequate dot products of vectors, this amounts to computing the minimal polynomial for a linear recurring sequence. If T_n is sparse, this can be done efficiently as the first step of Wiedemann's algorithm. Then recovering $g_1,\dots,g_{n-1}$ can be done by solving several linear systems of $\deg\mathscr{S}$ equations, that can be built almost for free using previous computations. Hence, a basis in Shape Position for I can be computed with knowledge of T_n only. Under additional assumptions, authors of [FM11] showed that T_n can be directly read from G_1. For the systems arising in Decomposition attacks, these hypotheses are always satisfied. This allows us to use the Sparse-FGLM algorithm in our realistic simulations for harvesting in Section 6.3.3.

Once a lexicographical basis has been computed, the last step to finish the solving of the system is to find roots of univariate polynomial of degree $\deg I$ — up to a random change of variables, but in our application Shape Position is obtained directly. A well-known root-finding algorithm for polynomials over $\mathbb{F}_q$ is given in [vzGG13, Chap. 14.5, p. 392]. It is also used in the harvesting phase of several Index-Calculus variants, see Sections 3.3.3 and 3.3.4. For a polynomial of degree d, it runs in $O(M(d)\log d\log(dq))$, where $M(d)$ is the time for multiplying polynomials of degree d over $\mathbb{F}_q$. Naive multiplication leads to $\tilde{O}(d^2\log q)$, and fast multiplication algorithms to $O(d\log^2 d\log(dq)\log\log d)$.

Example: Consider the polynomials $f_1 = y^2-x^3-x+1$ and $f_2 = x^2+4y^2-xy+2$ in $\mathbb{F}_{31}[x,y]$. Solving this system over any extension $L|\mathbb{F}_{31}$ amounts to finding the intersection points with coordinates in L of the 2 algebraic curves defined by the f_i. For the DRL order with $x>y$, the Gröbner Basis is

$$G_{DRL} = \left\{ \begin{array}{l} y^4+21y^3+30xy+7y^2+30x+20y+19, \\ xy^2+22y^3+21y^2+21x+11y+21, \\ x^2+30xy+4y^2+2 \end{array} \right\},$$

The staircase of G_{DRL} is

$$S = \{1,x,y,y^2,y^3,xy\},$$

so that $\dim\mathbb{F}_{31}[x,y]/\langle f_1,f_2\rangle = \deg\langle f_1,f_2\rangle = 6$. Hence $\mathbf{V}(f_1,f_2)$ is 0-dimensional and has at most 6 points with coordinates in $\overline{\mathbb{F}_{31}}$. We compute a lexicographical basis to obtain a set in Shape position

$$G_{lex} = \left\{ \begin{array}{l} x+h_1(y) = x+24y^5+15y^4+6y^3+10y^2+23y+17, \\ h_2(y) = y^6+21y^5+19y^4+8y^3+12y^2+18y+17 \end{array} \right\}.$$

Solving is then reduced to finding the roots of $h_2(y)$ in L, then evaluating h_1 at each root to find the corresponding value of x. In this example, h_2 has two roots $19,25$ in $\mathbb{F}_{31}$, so that the set of $\mathbb{F}_{31}$-rational points of $\mathbf{V}(f_1,f_2)$ is

$$\mathbf{V}_{\mathbb{F}_{31}}(f_1,f_2) = \{(-h_1(19),19),(-h_1(25),25)\} = \{(29,19),(4,25)\}.$$

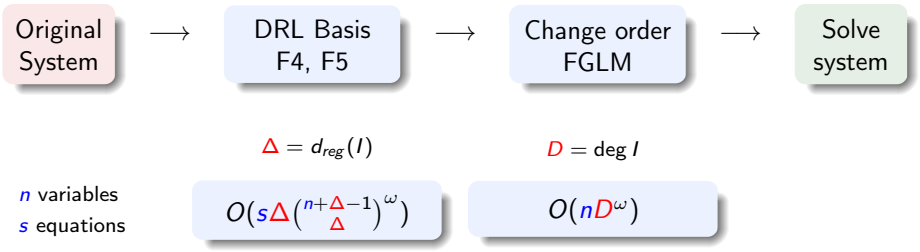


Figure 1.1: Strategy of resolution for a system of dimension 0

In our applications, the running time is usually dominated by the Change-order step, even using its Sparse variant. Therefore a main concern for us is to reduce as much as possible the degree $\deg I$ of the ideal. This can be done using the algebraic properties of the system: symmetries [Col97, FGHR14, FHJ+14, Ste13, Stu08, FS12, FS13], weighted homogeneous structures [FSEDV16], multi-homogeneous structure [FDS11, Spa12], sparsity [FSS14], determinantal structure [Spa12, BFJ+16]... For our contributions, we consider systems whose set of solutions (equivalently, ideals whose associated variety) is stable under the action of a group of symmetry, and also systems whose defining equations are given by a (power of) the Frobenius automorphism in positive characteristic.

1.4 Elimination Theory

This Section mostly follows [CLO97, Chap. 3]. Among our contributions, the introduction of Summation Ideals involves the computation of a projection of the Summation Variety (in Section 5.1). *Elimination ideals* describe all the projections of a variety, and can be described by computing Gröbner Bases for elimination order. Another key object of Elimination theory, namely the Resultant of two polynomial wrt. one variable, is then briefly introduced with the properties that we will need. It is also used to compute Semaev's Summation Polynomials (Proposition 3.13), and appear in several occasions in our work for the modelling of systems arising in Decomposition attacks (Sections 5.1 and 6.1.1).

Definition 1.59 (Elimination Ideals)**.** *Let I be an ideal in $K[X_1,\dots,X_n]$, and let $1 \leq k \leq n$ be an integer. The k^{th} elimination ideal of I is the ideal I_k in $K[X_k,\dots,X_n]$ defined as $I_k = I \cap K[X_k,\dots,X_n]$.*

Describing such ideals can be done by means of Gröbner basis and elimination order.

Proposition 1.60. *Let I be an ideal in $K[X_1,\dots,X_n]$, and for $0 \leq k \leq n$, let I_k be the k^{th} elimination ideal of I. Let also G be a Gröbner basis of I for an elimination order $>_{k-elim}$ (see Definition 1.40). Then the set $G_k = G \cap K[X_k,\dots,X_n]$ is a Gröbner basis for I_k.*

Proof. By definition $G_k \subset I_k$, hence $\langle \mathrm{LT}(G_k)\rangle \subset \langle \mathrm{LT}(I_k)\rangle$. Let f be in I_k. In particular, f is in I so $\mathrm{LT}(f)$ is divisible by $\mathrm{LT}(g)$, for some $g \in G$. Since $f \in K[X_k,\dots,X_n]$, then $\mathrm{LT}(g) \in K[X_k,\dots,X_n]$ too. Because of the monomial order, this means that $g \in K[X_k,\dots X_n]$, hence $g \in G_k$ and $\langle \mathrm{LT}(I_k)\rangle \subset \langle \mathrm{LT}(G_k)\rangle$. From Definition 1.41, G_k is a Gröbner basis of I_k. □

Next we relate elimination ideals with projections of an affine algebraic variety onto a set of variables.

Lemma 1.61. *Let I be an ideal in $K[X_1,\dots,X_n]$, and for a fixed $0 \leq k \leq n$, let I_k be the k^{th} elimination ideal. Consider the variety $V = \mathbf{V}(I)$ in $\mathbb{A}^n$ and the projection $\pi_k : \mathbb{A}^n \longrightarrow \mathbb{A}^{n-k}$ defined as $\pi_k(a_1,\dots,a_n) = (a_k,\dots,a_n)$. Then we have $\pi_k(V) \subset \mathbf{V}(I_k)$.*

Proof. Let $f \in I_k$ and $(a_1,\dots,a_n) \in V$. Then $f(a_1,\dots,a_n) = 0$. Since $f \in K[X_k,\dots,X_n]$, this rewrites as $f(\pi_k(a_1,\dots,a_n)) = 0$. □

Lemma 1.61 also states that $\pi_k(V) = \{(a_k,\dots,a_n) \in \mathbf{V}(I_k) : \exists\, a_1,\dots,a_{k-1} \in K \text{ s.t. } (a_1,\dots,a_n) \in V\}$. It can happen that the projection $\pi_k(V)$ is not an affine algebraic variety. For example, consider the variety in $\mathbb{A}^3$ defined by

$$V : \begin{cases} xy = 1, \\ xz = 1 \end{cases}$$

which admits the Gröbner basis $G = \{xz-1, y-z\}$ for the 1st elimination order. Then $I_1 = \langle y-z\rangle$, and $(0,0) \in \mathbf{V}(I_1)$, but there is no point in V that projects onto $(0,0)$. More precisely, $\pi_1(V) = \{(a,a) \in \mathbb{A}^2 \ : \ a \neq 0\}$, which is not an affine algebraic variety. The next theorem gives a more explicit description of the link between $\pi_k(V)$ and $\mathbf{V}(I_k)$.

Theorem 1.62 (Closure Theorem)**.** *Let I be an ideal in $K[X_1,\dots,X_n]$ and for a fixed $0 \leq k \leq n$, let I_k be the k^{th} elimination ideal. Then $\mathbf{V}(I_k)$ is the (set-theoretically) smallest affine algebraic variety containing $\pi_k(V)$.*

Proof. See *e.g.* [CLO97, Thm. 3, p. 125]. □

Examples:

- In the previous example, the diagonal in the plane (yOz), with equation $y-z=0$, is the smallest algebraic variety containing $\pi_k(V)$, and it contains stricly $\pi_1(V)$.
- Let K be the algebraic closure of $\mathbb{F}_{1031}$ and consider the ideal in $\mathbb{F}_{1031}[a_1,a_2,x,y,z]$ generated by $I = \langle a_1^2 - x + 499, 2a_1a_2 + y + 582, a_2^2 - z\rangle$, with associated variety $V = \mathbf{V}(I)$. The second elimination ideal I_2 admits the Gröbner basis $G = \{x(1027z+965) + y^2 + 133y + +556\}$, and Theorem 1.62 ensures that $\pi_2(V) \subset \mathbf{V}(I_2)$.

Theorem 1.62 can be precised by a description of the "missing" points betwen $\pi_k(V)$ and $\mathbf{V}(I_k)$ — see [CLO97, Thm. 3, p. 125] — but we do not need this description in this thesis. A natural question is now the following: given a point in $\mathbf{V}(I_k)$, when does it lift up to a point in V? This problem is solved by the next theorem when the elimination is done for one variable. We first state the theorem, then explain how it can be used to lift up solutions when more than one variable have been eliminated. The proof involves resultants of polynomials and their properties; we postpone it to the end of this Section.

Theorem 1.63 (Extension Theorem)**.** *Let $I = \langle f_1,\dots,f_s\rangle$ be en ideal in $K[X_1,\dots,X_n]$ and I_1 be the first elimination ideal of I. For $1 \leq i \leq s$, write f_i in the form*

$$f_i = g_i(X_2,\dots,X_n)X_1^{N_i} + \text{ terms with lower degree in } X_1,$$

where $g_i \neq 0$. Let $\mathbf{c}^* = (c_2,\dots,c_n)$ be in $\mathbf{V}(I_1)$. If $\mathbf{c}^*$ is not in $\mathbf{V}(g_1,\dots,g_s)$, then there exists $c_1 \in K$ such that $(c_1,\mathbf{c}^*) \in \mathbf{V}(I)$.

When a k^{th} elimination ideal is considered, with $k > 1$, then we can consider the increasing chain of ideals

$$I_k \subset I_{k-1} \subset \cdots \subset I_1 \subset I_0 = I,$$

and apply the theorem to I_k and I_{k-1} as ideals in $K[X_{k-1},X_k,\dots,X_n]$, then to I_{k-1} and I_{k-2} as ideals in $K[X_{k-2},X_{k-1},\dots,X_n]$ and so on until an element of $\mathbf{V}(I_k)$ has been lifted to $\mathbf{V}(I)$, or until an element fails to lift up. Observe in Theorem 1.63 that, if one of the g_i's is a constant, then all element of $\mathbf{V}(I_1)$ lifts to $\mathbf{V}(I)$. This comes from the fact that, in this case, $\langle g_1,\dots,g_s\rangle = K[X_2,\dots,X_n]$, whose associated variety is empty by Weak Nullstellensatz (Theorem 1.5). We now introduce an important polynomial in elimination theory, the Resultant.

Definition 1.64 (Sylvester Matrix, Resultant)**.** *Let $f = \sum_{i=0}^{m} a_{m-i}x^i$ and $g = \sum_{i=0}^{n} b_{m-i}x^i$, with $a_0 \neq 0 \neq b_0$, be univariate polynomials with coefficient in some ring. The Sylvester Matrix $S(f,g)$ of f and g is the $m \times n$ matrix defined by*

$$S(f,g) = \begin{pmatrix} a_0 & 0 & \cdots & 0 & b_0 & 0 & \cdots & 0 \\ a_1 & a_0 & \ddots & \vdots & b_1 & b_0 & \ddots & \vdots \\ a_2 & a_1 & \ddots & 0 & b_2 & b_1 & \ddots & 0 \\ \vdots & & \ddots & a_0 & \vdots & & \ddots & b_0 \\ & \vdots & & a_1 & & \vdots & & b_1 \\ a_{m-1} & & & & b_{n-1} & & & \\ a_m & a_{m-1} & & \vdots & b_n & b_{n-1} & & \vdots \\ 0 & a_m & & & 0 & b_n & & \\ \vdots & \ddots & \ddots & a_{m-1} & \vdots & \ddots & \ddots & b_{n-1} \\ 0 & \cdots & 0 & a_m & 0 & \cdots & 0 & b_n \end{pmatrix}.$$

The resultant $Res_x(f,g)$ of f and g with respect to x is the determinant of $S(f,g)$.

We now list useful properties of the Resultant.

Proposition 1.65. *Let $f = \sum_{i=0}^{m} a_{m-i}X_1^i$ and $g = \sum_{i=0}^{n} b_{m-i}X_1^i$ be polynomials in $(K[X_2,\ldots,X_n])[X_1] = K[X_1,\ldots,X_n]$, with $a_0 \neq 0 \neq b_0$ and $m \geq n$. Let also $I = \langle f,g\rangle$.*

1. *The Resultant $Res_{X_1}(f,g)$ is in the second elimination ideal I_2.*
2. *The polynomials f and g have a common factor in $K[X_1,\ldots,X_n]$ if and only if $\mathrm{Res}_{X_1}(f,g) = 0$.*
3. *Let $\mathbf{c}^* = (c_2,\ldots,c_n)$ in K^{n-1} such that $\deg_{X_1} f(X_1,\mathbf{c}^*) = m$ and $\deg_{X_1} g(X_1,\mathbf{c}^*) = n$. Then*

$$\mathrm{Res}_{X_1}(f,g)(\mathbf{c}^*) = a_0(\mathbf{c}^*)^{m-n}\,\mathrm{Res}_{X_1}(f(X_1,\mathbf{c}^*), g(X_1,\mathbf{c}^*)).$$

Proof. See [CLO97, Chap. 3, §5, §6]. □

We can now present the proof of Theorem 1.63.

Proof. Let $\mathbf{c}^* = (c_2,\ldots,c_n)$ be in $\mathbf{V}(I_1)$ and not in $V(g_1,\ldots,g_s)$, and consider the ring homomorphism

$$\begin{array}{rccc} \lambda: & K[X_1,\ldots,X_n] & \longrightarrow & K[X_1] \\ & f & \longmapsto & f(X_1,\mathbf{c}^*). \end{array}$$

The image of I by λ is an ideal $\lambda(I)$ of $K[X_1]$, which is a principal ideal domain. Therefore, there exists $h \in K[X_1]$ such that $\lambda(I) = \langle h\rangle$. Assume first that h is a not a non-zero constant. Since K is algebraically closed, the Fundamental Theorem of Algebra says that there exists $c_1 \in K$ such that $h(c_1) = 0$. Thus for any $f \in I$, $f(c_1,\mathbf{c}^*) = 0$ so that $(c_1,\mathbf{c}^*) \in \mathbf{V}(I)$. It remains to show that h cannot be in K^*, which we do by contradiction. Assume $h = u \in K^*$ and fix any $f \in I$, with $\deg_{X_1} f = m$. Observe that $f(X_1,\mathbf{c}^*) = u$, and that by hypothesis, there exists a g_i such that $g_i(\mathbf{c}^*) \neq 0$. Now consider the polynomial $h = \mathrm{Res}_{X_1}(f_i,f)$. Proposition 1.65 states that

$$h(\mathbf{c}^*) = g_i(\mathbf{c}^*)^m\,\mathrm{Res}_{X_1}(u, f(X_1,\mathbf{c}^*)).$$

From the Definition of the Sylvester Matrix, we see that $S(u,f(X_1,\mathbf{c}^*))$ is a diagonal $m \times m$ matrix with u on its diagonal, hence $\mathrm{Res}_{X_1}(u,f(X_1,\mathbf{c}^*)) = u^m \neq 0$. This means that $h(\mathbf{c}^*) = g_i(\mathbf{c}^*)^m u^m \neq 0$. But Proposition 1.65 also states that $h \in I_1$, so that $h(\mathbf{c}^*) = 0$, a contradiction. □

1.5 Varieties of interest

The Summation variety introduced in Chapter 5 is a core element of our contribution. While it can be abstractly described in a straightforward manner, we give an explicit polynomial embedding using a *polynomial parametrization*. This embedding allows us to compute Summation Polynomials for Hyperelliptic Curves as a particular elimination ideal called the *ideal of relations* (Definition 1.70). This terminology stems from Invariant theory (see *e.g.* [Stu08]) where the problem of determining all the non-trivial polynomial relations between invariants is important. For these reasons, Section 1.5.1 focuses on such varieties and highlights their link with ideals of relations.

Decomposition attacks target curves defined over extension fields $\mathbb{F}_{q^n}$. There, the sets of valid decompositions can be described by polynomial equations generating an ideal of positive dimension. To find points in the associated variety, constraints must be added to the system, so that it is made 0-dimensional. The *Weil Restriction* of a variety over $\mathbb{F}_q$ gives a natural and practical way to put such constraints and to model decompositions as 0-dimensional systems, with the additional benefit that instead of solving over $\mathbb{F}_{q^n}$, we solve systems defined over $\mathbb{F}_q$. In Section 1.5.2 we recall the construction of the Weil Restriction for a variety defined over $\mathbb{F}_{q^n}$ using the action of Galois automorphism. The process gives a variety $\mathscr{W}_n(V)$ defined over $\mathbb{F}_q$, such that $\dim_{\mathbb{F}_q} \mathscr{W}_n(V) = n \cdot \dim_{\mathbb{F}_{q^n}} V$ and $\deg \mathscr{W}_n(V) = (\deg V)^n$. Those expressions are used in our estimates throughout Chapters 5 and 6.

1.5.1 Polynomial Parametrizations and Ideals of Relations

Notations: For integers m, l, we use the notations $\mathbf{X} = (X_1, \ldots, X_m)$ and $\mathbf{a} = (a_1, \ldots, a_l)$ throughout this section.

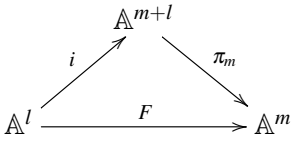
Definition 1.66 (Polynomial Parametrization)**.** *Let m, l be integers, and $P_1, \ldots, P_m$ be polynomials in $K[\mathbf{a}]$. A polynomial parametrization is an ideal I in $K[\mathbf{a}, \mathbf{X}]$ of the form*

$$I = \langle X_1 - P_1(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a}) \rangle .$$

Let V be a variety defined by a polynomial parametrization I. The problem of determining Cartesian equations for V is called the *implicitization problem*. Starting from the parametrization, it can be solved by computing a Gröbner basis for an adequate elimination order, as we now describe. Let $I = \langle X_1 - P_1(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a}) \rangle$ be a polynomial parametrization and $V = \mathbf{V}(I)$. If we let $F : \mathbb{A}^l \longrightarrow \mathbb{A}^m$ be a function defined as $F(\mathbf{a}) = (P_1(\mathbf{a}), \ldots, P_m(\mathbf{a}))$, it is straightforward to see that V can be seen as the graph of the function F. Define $i : \mathbb{A}^l \longrightarrow \mathbb{A}^{m+l}$ and $\pi_m : \mathbb{A}^{m+l} \longrightarrow \mathbb{A}^m$ by

$$\begin{aligned} i(\mathbf{a}) &= (\mathbf{a}, P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) \\ \pi_m(\mathbf{a}, \mathbf{x}) &= \mathbf{x}, \end{aligned}$$

then the following diagram is commutative:



From this we obtain $F(\mathbb{A}^l) = \pi_m(V)$, and also that $\dim \mathbf{V}(I) = l$. In the previous section we saw that the projection of an affine variety is not necessarily an affine variety. The discussion shows that solving the implicitization problem for V amounts to applying the Closure Theorem 1.62 to V.

Theorem 1.67 (Implicitization)**.** *Let V be an algebraic variety with polynomial parametrization $I = \langle X_1 - P_1(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a})\rangle$. Define $F : \mathbb{A}^l \longrightarrow \mathbb{A}^m$ as $F(\mathbf{a}) = (P_1(\mathbf{a}), \ldots, P_m(\mathbf{a}))$. Let I_l be the l^{th} elimination ideal of $I \subset K[\mathbf{a}, \mathbf{X}]$. Then $\mathbf{V}(I_l)$ is the smallest algebraic variety containing $F(\mathbb{A}^l)$.*

Remark 1.68. *This theorem can also be proved when the field is not algebraically closed, see [CLO97, Thm. 1, p. 130].*

This result prompts the next definition.

Definition 1.69. *We say that an algebraic affine variety V admits a polynomial parametrization if there is a polynomial parametrization $I \subset K[\mathbf{a}, \mathbf{X}]$ such that $V = \mathbf{V}(I_l)$. If we write $I = \langle X_1 - P_1(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a})\rangle$, we also say that V is parametrized by the P_i's.*

Examples: Over a field K with $\mathrm{Char}(K) \neq 2, 3$:

- The parabola $y = x^2 - 2x + 2$ admits a polynomial parametrization $x = 1 + t, y = 1 + t^2$. To see this, we observe that $\mathrm{Res}_t(x - 1 - t, y - 1 - t^2) = y - x^2 + 2x - 2$, and that the set $G = \{t - x + 1, x^2 - 2x - y + 2\}$ is a Gröbner basis for the first elimination order with $x > y$ for $\langle x - 1 - t, y - 1 - t^2\rangle$. Hence $G_1 = \{x^2 - 2x + 2 - y\}$ is the (reduced) Gröbner basis for I_1 and we have $V = \mathbf{V}(I_1)$.

- The surface defined by $x^2 - y^2z^2 + z^3 = 0$ admits a polynomial parametrization $f_1 = x - t(u^2 - t^2), f_2 = y - u, f_3 = z - u^2 + t^2$. Indeed, if we let $I = \langle f_1, f_2, f_3\rangle$, computations shows that $g_1 = \mathrm{Res}_u(f_2, f_3) = t^2 - z + y^2$ and $g_2 = \mathrm{Res}_u(f_1, f_3) = tz - x$ are in I_1. Then we find that $x^2 - y^2z^2 + z^3 = \mathrm{Res}_t(tg_2 - zg_1, g_2)$ belongs to I_2. A computation shows that $I_2 = \langle x^2 - y^2z^2 + z^3\rangle$.

- Let $\mathbf{V}(y^2 - x^2, z - x^3)$ be the twisted cubic, with polynomial parametrization $x = t, y = t^2, z = t^3$, and consider its tangent surface $\mathscr{S}$, that is to say, the ruled surface generated by all the tangents of the twisted cubic. It is not easy to derive a Cartesian equation starting from the equations defining the twisted cubic. However, elementary calculus leads to the following polynomial parametrization for $\mathscr{S}$ as $x = t - u, y = t^2 - 2tu, z = t^3 - 3t^2u$. Computing a Gröbner basis for the 2^{nd} elimination order gives a Cartesian equation of $\mathscr{S}$ as $f = x^3z - \frac{3}{4}x^2y^2 - \frac{3}{2}xyz + y^3 + \frac{1}{4}z = 0$.

Given polynomials $P_1, \ldots, P_m$ in $K[\mathbf{a}]$, an interesting question is to find non-trivial algebraic relations between the P_i's. For example, if the P_i's generate a polynomial algebra invariant under a matrix group action, knowing a Gröbner basis of the ideal of relations is used to obtain a unique writing of a polynomial expressed in the invariants of the group action. This is also known as *symmetrization*.

Definition 1.70 (Ideal of Relations)**.** *Let $P_1, \ldots, P_m$ be polynomials in $K[\mathbf{a}]$. The ideal of relations between the P_i's is defined as the set $I_R = \{g \in K[\mathbf{X}] \ : \ g(P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) = 0$ in $K[\mathbf{a}]\}$.*

For $f, g \in I_R$, it is clear that $f + g$ is also in I_R. If g is in $K[\mathbf{X}]$, then it is also clear that $f \cdot g \in I_R$, so I_R is an ideal. The next proposition highlights the link between ideals of relations and polynomial parametrizations.

Proposition 1.71. *For $P_1, \ldots, P_m$ polynomials in $K[\mathbf{a}]$, let $I = \langle X_1 - P_1(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a})\rangle$ be a polynomial parametrization, and I_R be the ideal of relations between the P_i's.*

1. *$f \in I \Leftrightarrow f(\mathbf{a}, P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) = 0$ in $K[\mathbf{a}]$.*

2. *$I_R = I \cap K[\mathbf{X}]$. In other words, I_R is the l^{th} elimination ideal of I in $K[\mathbf{a}, \mathbf{X}]$.*

Proof. If $f \in I$, then by definition there are polynomials q_i's such that $f = \sum_{i=1}^{m} q_i(X_i - P_i(\mathbf{a}))$, and thus $f(\mathbf{a}, P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) = 0$. This also shows that $I \cap K[\mathbf{X}] \subset I_R$. Observe that for any integer $k \geq 0$, the identity $(X+Y)^k = X^k + YC_k(X,Y)$ holds for the polynomial $C_\alpha(X,Y)$ defined by

$$C_1 = 1, \text{ and } C_k = \sum_{i=1}^{k-1} \binom{k}{i} X^i Y^{k-i-1} \text{ for } k \geq 2.$$

Hence, if $\mathfrak{m} = \mathbf{a}^\alpha \mathbf{X}^\beta = a_1^{\alpha_1} \ldots a_l^{\alpha_l} X_1^{\beta_1} \ldots X_m^{\beta_m}$ is a monomial in $K[\mathbf{X}]$, there are polynomials $C_{\beta_1}, \ldots, C_{\beta_m}$ in $K[\mathbf{a}, \mathbf{X}]$ such that:

$$\begin{aligned}
\mathfrak{m}(\mathbf{a}, \mathbf{X}) &= \mathfrak{m}(\mathbf{a}, P_1(\mathbf{a}) - (P_1(\mathbf{a}) - X_1), \ldots, P_m(\mathbf{a}) - (P_m(\mathbf{a}) - X_m)) \\
&= \mathbf{a}^\alpha \prod_{i=1}^{m} (P_i(\mathbf{a}) - (P_i(\mathbf{a}) - X_i))^{\beta_i} \\
&= \mathbf{a}^\alpha \prod_{i=1}^{m} (P_i(\mathbf{a})^{\beta_i} + (P_i(\mathbf{a}) - X_i) C_{\beta_i}(\mathbf{a}, X_i)).
\end{aligned}$$

For the next step, we assume temporarily that $m = 2$ to simplify the description. We have

$$\begin{aligned}
\mathfrak{m}(\mathbf{a}, \mathbf{X}) &= \mathbf{a}^\alpha \left(P_1(\mathbf{a})^{\beta_1} + (P_i(\mathbf{a}) - X_i) C_{\beta_1}(\mathbf{a}, X_i) \right) \left(P_2(\mathbf{a})^{\beta_2} + (P_2(\mathbf{a}) - X_2) C_{\beta_2}(\mathbf{a}, X_2) \right) \\
&= \mathfrak{m}(\mathbf{a}, P_1(\mathbf{a}), P_2(\mathbf{a})) + \mathbf{a}^\alpha \left(C_{\beta_1}(\mathbf{a}, X_1) P_2(\mathbf{a})^{\beta_2} + (P_2(\mathbf{a}) - X_2) C_{\beta_2}(\mathbf{a}, X_2) \right) (P_1(\mathbf{a}) - X_1) + \\
&\quad \mathbf{a}^\alpha P_1(\mathbf{a})^{\beta_1} C_{\beta_2}(\mathbf{a}, X_2)(P_2(\mathbf{a}) - X_2) \\
&= \mathfrak{m}(\mathbf{a}, P_1(\mathbf{a}), P_2(\mathbf{a})) + \tilde{C}_1 \cdot (P_1(\mathbf{a}) - X_1) + \tilde{C}_2 \cdot (P_2(\mathbf{a}) - X_2),
\end{aligned}$$

for some polynomials $\tilde{C}_1, \tilde{C}_2$ in $K[\mathbf{a}, \mathbf{X}]$. Now we come back to the general case, and an induction on m shows that there are polynomials $\tilde{C}_1, \ldots \tilde{C}_m$ in $K[\mathbf{a}, \mathbf{X}]$ such that

$$\mathfrak{m}(\mathbf{a}, \mathbf{X}) = \mathfrak{m}(\mathbf{a}, P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) + \tilde{C}_1 \cdot (P_1(\mathbf{a}) - X_1) + \cdots + \tilde{C}_m \cdot (P_m(\mathbf{a}) - X_m).$$

Since any f in $K[\mathbf{a}, \mathbf{X}]$ can be written $f = \sum_\alpha c_\alpha \mathfrak{m}^\alpha$, we can then find polynomials $B_1, \ldots, B_m$ in $K[\mathbf{a}, \mathbf{X}]$ such that

$$f(\mathbf{a}, \mathbf{X}) = f(\mathbf{a}, P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) + B_1 \cdot (P_1(\mathbf{a}) - X_1) + \cdots + B_m \cdot (P_m(\mathbf{a}) - X_m). \tag{1.2}$$

In particular, if $f(\mathbf{a}, P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) = 0$, then $f \in I$ and the first statement is proved. Now take $f \in I_R$, equation 1.2 rewrites as $f(\mathbf{X}) = B_1 \cdot (P_1(\mathbf{a}) - X_1) + \cdots + B_m \cdot (P_m(\mathbf{a}) - X_m) \in K[\mathbf{X}]$ and the second statement is proved. □

Thanks to Proposition 1.71 we now associate polynomial parametrizations with ideal of relations as their elimination ideal. We can precise the structure of the affine algebraic variety admitting a polynomial parametrization.

Corollary 1.72. *Let $P_1, \ldots, P_m$ be polynomials in $K[\mathbf{a}]$. A polynomial parametrization I in $K[\mathbf{a}, \mathbf{X}]$ by the P_i's and its ideal of relations I_R are prime ideals, and the associated varieties are irreducible.*

Proof. Assume f, g are in $K[\mathbf{X}]$, and that $f \cdot g \in I_R$; then $f(P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) \cdot g(P_1(\mathbf{a}), \ldots, P_m(\mathbf{a})) = 0$ in $K[\mathbf{a}]$ by definition, so one of the factor must be 0, equivalently f or g is in I_R. A similar argument in conjonction with the first statement of Proposition 1.71 shows that a polynomial parametrization is also a prime ideal. Irreducibility comes from Proposition 1.12. □

1.5.2 Weil Restrictions

Let $K = \mathbb{F}_{q^n}$ from some $n > 1$ and q a power of a prime, and $G = \mathrm{Gal}(\mathbb{F}_{q^n}|\mathbb{F}_q)$ be the Galois Group of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$, that is to say, the group of field automorphisms of $\mathbb{F}_{q^n}$ that fix any element of $\mathbb{F}_q$.

Definition 1.73. *Let $\tau \in G$ and $V = \mathbf{V}(f_1, \ldots, f_s)$ with $f_i \in \mathbb{F}_{q^n}[X_1, \ldots, X_r]$ be an algebraic variety defined over $\mathbb{F}_{q^n}$.*

- *For any $P = (x_1, \ldots, x_r) \in V$, we define $P^\tau = (\tau(x_1), \ldots, \tau(x_r))$.*
- *We define $V^\tau = \{P^\tau \ : \ P \in V\}$.*
- *For any polynomial f with coefficients over $\mathbb{F}_{q^n}$, we define f^τ as the polynomial obtained by action of τ over its coefficients: if $f = \sum_\alpha c_\alpha \mathfrak{m}_\alpha$, then $f^\tau = \sum_\alpha \tau(c_\alpha)\mathfrak{m}_\alpha$.*

Proposition 1.74. *For any $\tau \in G$, if $V = \mathbf{V}(f_1, \ldots, f_s)$ is defined over $\mathbb{F}_{q^n}$, then we have $V^\tau = \mathbf{V}(f_1^\tau, \ldots, f_s^\tau)$.*

Proof. Let $P = (x_1, \ldots, x_r) \in V$, so that by definition $f_i(x_1, \ldots x_r) = 0$ for all $1 \leq i \leq s$. Let $f_i = \sum c_{\alpha,i}\mathfrak{m}_{\alpha,i}$. Since τ is a field automorphism, we have $f_i(P) = \sum c_{\alpha,i}\mathfrak{m}_{\alpha,i}(x_1, \ldots, x_r) = 0$ if and only if $\tau(f_i(P)) = \sum \tau(c_{\alpha,i})\mathfrak{m}_{\alpha,i}(\tau(x_1), \ldots, \tau(x_r)) = 0$ or equivalently, $f_i^\tau(P) = 0$. □

The Weil Restriction of V over $\mathbb{F}_q$ is a variety defined over $\mathbb{F}_q$, whose $\mathbb{F}_q$-rational points are in one-to-one correspondence with the $\mathbb{F}_{q^n}$-points of V. There are two possible and equivalent constructions.

Definition 1.75. *The Weil Restriction of V over $\mathbb{F}_q$ is $\mathscr{W}_n(V) = \prod_{\tau \in G} V^\tau$. If a generating set $\mathbb{S}$ for $I = \mathbf{I}(V)$ is given, we also use the notation $\mathscr{W}_n(\mathbb{S})$ or $\mathscr{W}_n(I)$.*

The Weil Restriction is defined over $\mathbb{F}_q$ as it is invariant under the "twisted" action of $\mathrm{Gal}(\overline{\mathbb{F}_q}|\mathbb{F}_q)$ defined by

$$\tau \cdot (P^\sigma, \ldots, P^{\sigma^{n-1}}, P) = ((P^{\overline{\tau}^{-1}\circ\sigma})^\tau, \ldots, (P^{\overline{\tau}^{-1}})^\tau), \ \forall \tau \in \mathrm{Gal}(\overline{\mathbb{F}_q}|\mathbb{F}_q),$$

where $\overline{\tau} = \tau|_{\mathbb{F}_{q^n}}$. It also satisfies functorial and universal properties that we do not detail here — see for example [CF05, Chap.7] or [Vit11, Chap. 6]. From Definition 1.75, it can also be noted that $\dim_{\mathbb{F}_q} \mathscr{W}_n(V) = n \cdot \dim_{\mathbb{F}_{q^n}} V$. Because the action of an automorphism in the Galois group does not change the monomial support of a polynomial, we see from the definition of the degree of a variety V that $\deg V = \deg V^\tau$ for any $\tau \in G$. By Proposition 1.30, we then obtain that $\deg \mathscr{W}_n(V) = (\deg V)^n$.

For the sake of completeness we give a description of Weil Restrictions in the language of ideals. Let $V = \mathbf{V}(f_1, \ldots, f_r)$ and $I = \langle f_1, \ldots, f_r \rangle$. Let also σ be any generator of G. Define $I^{\sigma^i} = \left\langle f_1^{\sigma^i}, \ldots, f_r^{\sigma^i} \right\rangle$ in $\mathbb{F}_{q^n}[X_1, \ldots, X_s]$ for any $1 \leq i \leq n$. We now see $f_j^{\sigma^i}$ as polynomials in $\mathbb{F}_{q^n}[X_{i,1}, \ldots, X_{i,s}]$, and we form the ideal

$$\mathbb{I} = \left\langle f_1^{\sigma^i}, \ldots, f_r^{\sigma^i} : 1 \leq i \leq n \right\rangle \subset \mathbb{F}_{q^n}[X_{1,1}, \ldots, X_{1,r}, \ldots, X_{n,1}, \ldots, X_{n,s}].$$

Since for any f and any i, f and f^{σ^i} have the same monomials, we have $\deg I = \deg I^{\sigma^i}$ for all i, hence $\deg \mathbb{I} = \prod_{i=0}^{n-1} \deg I^{\sigma^i} = (\deg I)^n$. Using Proposition 1.74, we can also show that $\mathscr{W}_n(V) = \mathbf{V}(\mathbb{I})$. In other words we described the ideal theoretic translation of the Weil Restriction of V. Notice that this can be used for ideals independantly from the fact that they are radical or not.

A practical Definition for the Weil Restriction The Weil Restriction is involved in the modelling of the harvesting phase in a Decomposition attack as 0-dimensional systems. However, the above description using Galois theory is not practical. For computational purpose, we now describe a variety isomorphic to $\mathscr{W}_n(V)$ — see [CF05, Chap.7, p. 129]. Let $\theta_1,\ldots,\theta_n$ be a $\mathbb{F}_q$-basis of $\mathbb{F}_{q^n}$. Using this basis, we let $X_i = \sum_{i=1}^n X_{ij}\theta_i$. Any polynomial $f = \sum c_\alpha \mathfrak{m}_\alpha$ in $\mathbb{F}_{q^n}[X_1,\ldots,X_r]$ can then be rewritten as

$$f(\sum_{i=1}^n X_{1i}\theta_i,\ldots,\sum_{i=1}^n X_{ri}\theta_i) = \sum_{i=1}^n f_i(X_{11},\ldots,X_{1n},\ldots,X_{r1},\ldots,X_{rn})\theta_i,$$

for some $f_i \in \mathbb{F}_q[X_{11},\ldots,X_{rn}]$, and $f=0$ if and only if all $f_i = 0$. Now, if $V = \mathbf{V}(f_1,\ldots,f_r)$, then we can write $f_i = \sum_{j=1}^n f_{ij}\theta_i$ and define a variety $W = \mathbf{V}(f_{11},\ldots,f_{rn})$. By definition W is defined over $\mathbb{F}_q$, and verifies the universal properties of the Weil Restriction, so that W is isomorphic to $\mathscr{W}_n(V)$. Moreover, the $\mathbb{F}_q$-rational points of W are in one-to-one correspondence with the $\mathbb{F}_{q^n}$-rational points of V. Building the variety W from the input equations for V is easily done with a computer algebra system such as Magma. To obtain a 0-dimensional subvariety of W, additional constraints must be added; this is explained in Section 3.4.3.

Example: Let $K = \mathbb{F}_{31}$ and $L|K$ be a quadratic extension with an element t such that $t^2 = 2t+28$ and $\{1,t\}$ is a K-basis of L. Let $P(X) = X^2 + (24t+29)X + 10t + 24$ in $L[X]$. The variety $\mathbf{V}(P)$ has dimension 0 and we find the roots of P as $\mathbf{V}_L(P) = \mathbf{V}(P) = \{11t+28, 27t+5\}$. Now, we evaluate

$$\begin{aligned} P(x_0+x_1t) &= (x_0^2 + 29x_0 + 28x_1^2 + 21x_1 + 24) + (2x_0x_1 + 24x_0 + 2x_1^2 + 15x_1 + 10)t \\ &= P_0(x_0,x_1) + P_1(x_0,x_1)t, \end{aligned}$$

and set $W = \mathbf{V}(P_0,P_1)$. A Gröbner Basis computation for DRL order with $x_0 > x_1$ gives

$$G_{DRL} = \left\{ \begin{array}{c} x_1^3 + 5x_1^2 + 14x_0 + 20x_1 + 25, \\ x_0^2 + 28x_1^2 + 29x_0 + 21x_1 + 24, \\ x_0x_1 + x_1^2 + 12x_0 + 23x_1 + 5 \end{array} \right\}.$$

The staircase of G_{DRL} is $\{1, x_0, x_1, x_1^2\}$, so that W has at most $(\deg P)^2 = 4$ points with coordinates in an algebraic closure of K. In lexicographical order with $x_0 > x_1$, we find in Shape Position

$$G_{lex} = \left\{ \begin{array}{c} x_0 + 20x_1^3 + 7x_1^2 + 28x_1 + 4, \\ x_1^4 + 17x_1^3 + 4x_1^2 + 5x_1 + 13 \end{array} \right\}.$$

The univariate polynomial has exactly two roots in K, which are 11 and 27. Evaluating the first polynomial at those values, we recover the values 28 and 5 for x_0, which lead to the roots of P over L. In other words, $\mathbf{V}_L(P) \simeq \mathbf{V}_K(P_0,P_1)$.

Positive dimension to Zero-dimension with Weil Restrictions We fix an extension $\mathbb{F}_{q^n}$. Consider an affine variety V define over $\mathbb{F}_{q^n}$, and assume that we are given an embedding $V \longrightarrow \mathbb{A}^m(\mathbb{F}_{q^n})$. Let $g = \operatorname{codim} V$ so that $\dim V = m-g$, and so that there is an integer $r \geq g$ and polynomials $f_1,\ldots,f_r \in \mathbb{F}_{q^n}[X_1,\ldots,X_m]$ such that $\mathbf{V}(f_1,\ldots,f_r) = V$.

No general algorithmic method exists to find a $\mathbb{F}_{q^n}$-point in V as it has positive dimension. By adding constraints to the system, we can make the dimension fall to 0. We can then use the strategy from Section 1.3.3, and we may find $\mathbb{F}_{q^n}$-rational points. The problem now is to determine what can be "good" linear constraints on the elements of V. Since we are working in $\mathbb{F}_{q^n}$, an example of natural constraint is that the points must have their coordinates in a subfield, say $\mathbb{F}_q$. This amounts to adding the equations $X_i^q - X_i$ to the system $(f_1,\ldots,f_r)$. However, we do not want the degree of the system to rise too much, as the efficiency of the

solving process depends crucially on the degree of the equations. At best, we want to add linear constraints, or geometrically, to cut V by enough hyperplanes.

This is where we can use the linear structure of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$. Let $\theta_1,\dots,\theta_n$ be a $\mathbb{F}_q$-basis of $\mathbb{F}_{q^n}$ and build the "practical" Weil Restriction $W = \mathscr{W}_n(V)$. Properties of the restriction says that $\dim W = n(m-g)$, $\operatorname{codim} W = ng$, and that $W = \mathbf{V}(f_{11},\dots,f_{1n},\dots,f_{r1},\dots,f_{rn})$ with the f_{ij} polynomials in $\mathbb{F}_q[X_{11},\dots,X_{1n},\dots,X_{m1},\dots,X_{mn}]$ defined in the previous paragraph using the writing $X_i = \sum_{i=1}^{n} X_{ij}\theta_i$. Recall that the Frobenius automorphism $\sigma(x) = x^q$ generating $\mathrm{Gal}(\mathbb{F}_{q^n}|\mathbb{F}_q)$ is a $\mathbb{F}_q$-linear map. From this we infer that $X_i^q - X_i = 0$ if and only if $(X_{i1},\dots,X_{in}) \in \ker(\sigma - \mathrm{Id})$. Since 1 is a simple root of the minimal polynomial $T^n - 1$ of σ, this kernel has dimension 1 and thus must be defined by $n-1$ independent linear equations. In practice, we identify $\mathbb{F}_{q^n}$ with some $\mathbb{F}_q[t]/\langle P\rangle$, where P is an irreducible polynomial over $\mathbb{F}_q$ with $\deg P = n$. This gives a power basis $1,t,\dots,t^{n-1}$, so that the $n-1$ equations are $X_{i,2} = \dots = X_{i,n} = 0$, see also Section 3.4.2.

Overall, we add $m(n-1)$ linear equations to the system of the f_{ij}'s, giving at least $gn + m(n-1)$ equations. This means the intersection of W with the restrictions of all $\mathbf{V}(X_i^q - X_i)$ has dimension 0 when

$$gn + m(n-1) \geq mn \Leftrightarrow g \geq \frac{m}{n}. \tag{1.3}$$

This happens in all the applications we are interested in (Section 3.4.2, Chapter 5 and Chapter 6). For the sake of completeness, we mention that smoother constraints can be added by asking that the X_i's belong to $\mathbb{F}_q$-linear subspaces of $\mathbb{F}_{q^n}$ of higher dimension d. Equations (1.3) then becomes $g \geq \frac{md}{n}$.

Chapter 2

Algebraic Curves

All this thesis is centered around an object which is built over Algebraic Curves. Their general theory is a rich and active research field with striking consequences, a famous one being the proof of the Fermat-Wiles theorem, and somewhat unexpected practical applications, such as Curve Based Cryptography. Indeed, any curve can be associated with an algebraic group, namely its *Jacobian Variety*. Hence the Discrete Logarithm Problem can be considered on such objects, when the base field is finite. We often abuse the terminology and use "Discrete Logarithm on Curves" when we mean on the Jacobian Variety of said curve.

This Chapter describes the necessary material to understand what is the Jacobian Variety of a Curve, and gives the most meaningful examples for this thesis. Starting with the general basics of the theory, Section 2.1 introduces *divisors* and *functions* over a curve. They are used to define the *Picard Group* of the curve, and the Jacobian Variety is then defined as a particular subgroup (Definition 2.9). The *genus* of the curve is an important parameter that helps the classification of algebraic curves. We define it by means of the Riemann-Roch theorem (Theorem 2.12), following mostly [Sil13]. Moreover, when a curve has genus g, then its Jacobian Variety can also be described as an abelian variety of dimension g (see Section 2.1.3). This means that the group law can be expressed using rational functions on the coordinates once a projective embedding has been chosen. A short discussion about the geometric interpretation of the Jacobian arithmetic concludes this Section.

We then focus on *Hyperelliptic Curves*, targeted by our contributions of Chapters 5 and 6. In the litterature, Hyperelliptic Curves are sometimes defined as curves which admit a double cover of the projective line. All the theory can be developed from this, but it involves more material than actually needed for our applications. For this reason, we choose to directly define Hyperelliptic Curves by their well-known *Weierstrass model* (Definition 2.17). What makes Hyperelliptic Jacobian Varieties interesting for practical applications is that the arithmetic can be described by polynomial arithmetic using the *Mumford Representation* and *Cantor's algorithm* — see Section 2.2.2. .

While hyperelliptic arithmetic stays too costly in general for practical implementations, recent works using *theta functions arithmetic* [Gau07, GL09, LR16, RSSB16, BCHL16] showed a potential competitivity of genus 2 curves with *Elliptic Curves*, topic of Section 2.3. Following the introduction of Elliptic Curve Cryptosystems, several elliptic curve standards have been proposed [LM10, NIS99, Res10]. They are now widely used in practical implementations (SSL/TLS, ...) and embedded systems. Hence Elliptic Curves are the main reason for this work. They are sometimes defined as hyperelliptic curves of genus 1; we choose to define them as genus 1 curve with a rational point. Apart from there genus (it is always 1), they stand out of the realm of algebraic curves as being the only class of curves that identify to its Jacobian Variety (Proposition 2.22). The arithmetic expresses geometrically by the

well-known *chord-tangent method*, and is detailed in Section 2.3.2. Brief comments are given considering practical and secured implementation using special models of curves ((twisted) Edwards, (twisted) Hessian, Montgomery,...).

2.1 Curves and their Jacobian Varieties

General notions about the theory of algebraic curves are detailed here. First, we recall results about functions over a curve. This is needed in Section 2.1.2 to introduce the Picard group of a curve. We develop briefly the notion of *divisor* over a curve, which gives an algebraic formalism to express local properties of functions. The Picard Group is then introduced as a quotient of the group of all divisors, and the Jacobian Variety is defined as the *degree 0 Picard Group*, see Definition 2.9. We recall the classic Weil-Hasse bounds for the cardinality of a curve and its Jacobian Variety when the base field is finite. Those bounds are used in nearly all complexity estimates for Index-Calculus over Jacobian Varieties. Next, the genus is introduced thanks to Riemann-Roch's theorem (Theorem 2.12), and used to give a way to describe elements in the Jacobian Variety. This representation is used extensively in Chapter 4 to describe our new Sieving approach to harvesting. Lastly, the geometric structure of Jacobian Varieties is described, highlighting that they belong to the family of *Abelian Varieties*. All the presentation mostly follows [Sil13] and [Ful08].

2.1.1 Functions over a curve

Definition 2.1. *An algebraic (projective) curve is an irreducible (projective) algebraic variety of dimension* 1.

Let $\mathscr{C}$ be an algebraic curve. It is possible that $\mathscr{C}$ contains singular points, which means that the tangent space at this point has dimension greater than expected. Equivalently, the Jacobian Matrix vanishes at this point. However, it is well-known — see for example [Ful08, Chap. 7.5] — that any projective curve admits a non-singular model. More precisely, we say that two curves are birational when their function field (Definition 1.24) are isomorphic. The correct statement is then that every projective curve is birational to a non-singular projective curve. Hence wlog. we now only consider non-singular curves. We also call them *smooth* curves.

Proposition 2.2 ([Sil13], Chap. 2.1, p.18)**.** *Let $\mathscr{C}$ be a smooth algebraic curve. Up to a linear change of variable, $K(\mathscr{C})$ is a finite and separable extension of $K(x)$.*

From the definition of the function field (Definition 1.24), any $f \in K(\mathscr{C})$ can be written as a rational fraction g/h with $h \notin \mathbf{I}(\mathscr{C}) = I$. Usually, many choices of g,h can be made as $K[\mathscr{C}]$ may not be an UFD, and two rational fractions $\frac{g}{h}, \frac{g'}{h'}$ define the same function if $gh' - g'h \in I$. By $P \in \mathscr{C}$ we mean that the point has coordinates in $\overline{K}$. If $\mathscr{C}$ is defined over K, we write $P \in \mathscr{C}(K)$ for points with coordinates in K.

Definition 2.3. *Let $\mathscr{C}$ be a projective curve and fix a point $P \in \mathscr{C}$. We say that f is defined (or regular) at P if there exists $g,h \in \overline{K}[\mathscr{C}]$ such that $f = \frac{g}{h}$ and $h(P) \neq 0$. The ring of regular functions at P is denoted by $\overline{K}[\mathscr{C}]_P$. The maximal ideal at P is the set $\mathscr{M}_P = \{f \in \overline{K}[\mathscr{C}] : f(P) = 0\}$.*

If $\mathscr{C}$ is defined over K, any $\overline{K}$ can be replaced by K. It is straightforward to check that the ring of regular functions at a point is indeed a ring and that $\mathscr{M}_P$ is a maximal ideal of $K[\mathscr{C}]$. Thanks to Definition 2.3, a function defined at P can be evaluated by $f(P) = \frac{g(P)}{h(P)}$ for suitable g,h. This does not depend on the choice of representant for f: if $f = \frac{g}{h} \sim \frac{g'}{h'}$, then we have $gh' = g'h + I$. From the definition of I, we get $g(P)h'(P) = g'(P)h(P)$ for any $P \in \mathscr{C}$.

A more interesting result is that $K[\mathscr{C}]_P$ is a *discrete valuation ring*, which means that $K[\mathscr{C}]_P$ local, with maximal ideal $\mathscr{M}_P$, and that $\mathscr{M}_P$ is principal. The order of a regular function at a point can now be defined. It is an algebraic formulation of the concept of multiplicity. For any polynomial ideal I and nonnegative integer d, we define $I^d = \langle\ f^d\ :\ f \in I\ \rangle$.

Definition 2.4. *Let $\mathscr{C}$ be an algebraic curve and $P \in \mathscr{C}$. The order a regular function f at P is*

$$\begin{array}{rccc} ord_P: & \overline{K}[\mathscr{C}]_P & \longrightarrow & \mathbb{N} \cup \{\infty\} \\ & f & \longmapsto & Sup\ \{d \in \mathbb{N}\ :\ f \in \mathscr{M}_P^d\}. \end{array}$$

It is extended to any function $f = \frac{g}{h}$ in $\overline{K}(\mathscr{C})$ by $ord_P(\frac{g}{h}) = ord_P(g) - ord_P(h)$. Any $t \in K[\mathscr{C}]_P$ such that $ord_P(t) = 1$ is called a uniformizing parameter at P.

Any uniformizing parameter generates $\mathscr{M}_P$, and therefore two of them differ only from a unit in $K[\mathscr{C}]$.

Definition 2.5. *Let $\mathscr{C}$ be an algebraic curve defined over K. Let $P \in \mathscr{C}$, and $f \in K[\mathscr{C}]_P$.*

- *f has a zero at P is $f(P) = 0$.*
- *f has a pole at P if $1/f$ has a zero at P.*

A function f is regular at P if and only if $\mathrm{ord}_P(f) \geq 0$, and has a zero at P when $\mathrm{ord}_P(f) > 0$. Similarly, f has a pole at P when $\mathrm{ord}_P(f) < 0$. The next result is key in the definition of the Jacobian Variety.

Proposition 2.6. *Let $\mathscr{C}$ be an algebraic curve, and $f \in \overline{K}(\mathscr{C})$ with $f \neq 0$. Then f has the same finite number of zeros and poles.*

2.1.2 Divisors and the Jacobian variety

We now associate abelian groups with algebraic curves. Let K be a field with algebraic closure $\overline{K}$, and consider an algebraic curve $\mathscr{C}$.

Definition 2.7. *The group of divisors on $\mathscr{C}$ is the free abelian group $Div\,\mathscr{C}$ generated by the points of $\mathscr{C}$. Hence, a divisor $D \in Div\,\mathscr{C}$ is a formal $\mathbb{Z}$-linear combination of points of $\mathscr{C}$, that is to say:*

$$D = \sum_{P \in \mathscr{C}} n_P P,\ n_P \in \mathbb{Z},$$

with finitely many $n_P \neq 0$. The support of D is the set $\mathrm{Supp}(D) = \{P \in \mathscr{C}\ :\ n_p \neq 0\}$. If $\mathscr{C}$ is defined over K then a divisor is defined over K if its support is invariant under the action of the Galois group $Gal(\overline{K}\,|\,K)$. The group of divisors defined over K is denoted when it matters by $Div_K\,\mathscr{C}$. The degree of D is the integer $\deg D = \sum_{P \in \mathscr{C}} n_P$. The subgroup of degree 0 divisors is denoted by $Div^0\,\mathscr{C}$.

If a divisor is defined over K, this does not necessarily mean that the points in its support are K-rational.

Definition 2.8. *Let $f \in \overline{K}(\mathscr{C})$. The principal divisor associated to f is defined by*

$$\mathrm{div}\, f = \sum_{P \in \mathscr{C}} ord_P(f) \cdot P.$$

The subgroup of principal divisors is denoted $Prin\,\mathscr{C}$.

Since $\mathrm{ord}_P(fg) = \mathrm{ord}_P(f) + \mathrm{ord}_P(g)$ for any $f, g \in \overline{K}[\mathscr{C}]_P$, we see that $\mathrm{div}\, f + \mathrm{div}\, g = \mathrm{div}(fg)$, so that $\mathrm{Prin}\,\mathscr{C}$ is indeed a subgroup of $\mathrm{div}\,\mathscr{C}$. If $\mathscr{C}$ is defined over K, then any $f \in K(\mathscr{C})$ gives a principal divisor defined over K. Thanks to Proposition 2.6, principal divisors have degree 0, so they indeed form a subgroup of $\mathrm{Div}^0\,\mathscr{C}$.

Definition 2.9 (Jacobian Variety as degree 0 Picard Group)**.** *Two divisors D_1 and D_2 are said to be* linearly equivalent *when $D_1 - D_2 = \operatorname{div} f$ for some function $f \in \overline{K}(\mathscr{C})$. The Jacobian Variety of $\mathscr{C}$ is the group of linear equivalence classes among degree 0 divisors, or in other words, the quotient group $\mathrm{Jac}(\mathscr{C}) = Div^0\ \mathscr{C}/Prin\ \mathscr{C}$. If $\mathscr{C}$ is defined over K, then $\mathrm{Jac}_K(\mathscr{C})$ is the subgroup of $\mathrm{Jac}(\mathscr{C})$ which is invariant under the action of $Gal(\overline{K}\,|\,K)$.*

The set of all classes, or equivalently, the quotient group Div $\mathscr{C}$/Prin $\mathscr{C}$ is known as the Picard Group of the curve. When $\mathscr{C}$ is defined over K, it is possible to show that two divisors D_1, D_2 in $\mathrm{Div}^0_K\ \mathscr{C}$ are linearly equivalent if and only if they are linearly equivalent over K, that is to say $D_1 = D_2 + \operatorname{div} f$ for some $f \in K(\mathscr{C})$. As a consequence, we obtain $\mathrm{Jac}_K(\mathscr{C}) = \mathrm{Div}^0_K\ \mathscr{C}/\mathrm{Prin}_K\ \mathscr{C}$ when $\mathscr{C}$ is defined over K.

Riemann-Roch's theorem and the genus of a curve When working with a quotient group, a representation for the classes is needed. Convenient representants for elements in $\mathrm{Jac}(\mathscr{C})$ can be obtained using the *genus* of the curve and Riemann-Roch's theorem, which we now introduce.

Definition 2.10. *A divisor $D = \sum_{P \in \mathscr{C}} n_P P$ on $\mathscr{C}$ is effective when $n_P \geq 0$ for all P, and it is denoted by $D \geq 0$. For any $D_1, D_2 \in Div\ \mathscr{C}$, we write $D_1 \geq D_2$ when $D_1 - D_2$ is effective.*

Definition 2.11. *Let $\mathscr{C}$ be an algebraic curve, and $D \in Div^0\ \mathscr{C}$. The Riemann-Roch space associated to D is the finite dimensional $\overline{K}$-linear space*

$$\mathscr{L}(D) = \{f \in \overline{K}(\mathscr{C})^* \ :\ \operatorname{div} f \geq -D\} \cup \{0\}.$$

We also let $l(D) = \dim_{\overline{K}} \mathscr{L}(D)$.

It is not immediate that Riemann-Roch space have finite dimension; a proof is given in [Ful08]. For $l(D)$ to be greater than 0, it is necessary that $\deg D \geq 0$. For any $D \in \operatorname{div}\mathscr{C}$ and any point $P \in \mathscr{C}$, we have $l(D) \leq l(D+P) \leq l(D)+1$, see [Ful08, Prop.3, p.99]. In other words, adding a point to a divisor increases the dimension of the Riemann-Roch space by at most 1.

Theorem 2.12 (Riemann-Roch, [Ful08])**.** *Let $\mathscr{C}$ be an algebraic curve. There exists a unique integer $g \geq 0$, and a divisor E such that for every $D \in Div\ \mathscr{C}$,*

$$l(D) - l(E - D) = \deg D - g + 1.$$

The integer g is called the genus of $\mathscr{C}$.

Usually there are many E that fit into the statement of the theorem — they are called canonical divisors and related to differential forms, see [Sil13, Chap.2] or [Ful08, Chap.8]. The weaker form, known as Riemann's theorem, gives a lower bound for $l(D)$. Some immediate consequences of Theorem 2.12 are:

- $l(E) = g$ by setting $D = 0$;
- $\deg E = 2g - 2$ by setting $D = E$;
- if $\deg D > 2g - 2$, then $l(E - D) = 0$ and hence $l(D) = \deg D - g + 1$.

The next corollary gives a representation of elements of the Jacobian thanks to the genus. It is also used to work rigorously with divisors, for example in our Sieving approach in Chapter 4.

Corollary 2.13 (Representation for elements in the Jacobian variety)**.** *Let $\mathscr{C}$ be an algebraic curve of genus g defined over a field K, with at least one K-rational point $\mathscr{O}$. For any $D \in Div^0\ \mathscr{C}$, there exists an integer $0 \leq k \leq g$ and $P_1, \ldots, P_k \in \mathscr{C}$ such that D is equivalent to $P_1 + \cdots + P_k - k\mathscr{O}$.*

Proof. Let $D \in \mathrm{Div}^0\,\mathscr{C}$ and consider the divisor $D_1 = D - g\mathscr{O}$. From Riemann-Roch's theorem we have $l(D_1) \geq \deg D_1 - g + 1 = 1$ so there exists a non-constant $f \in \mathscr{L}(D_1)$. This function f is such that $\mathrm{div}\, f - D + g\mathscr{O}$ is effective of degree g. Therefore, there exists $P_1, \ldots, P_g \in \mathscr{C}$ such that $\mathrm{div}\, f - D + g\mathscr{O} = P_1 + \cdots + P_g$. Now some cancellation can happen if some P_i's are $\mathscr{O}$. □

In other words, any degree 0 divisor can be represented in $\mathrm{Jac}(\mathscr{C})$ by a divisor of the form $P_1 + \cdots + P_k - k\mathscr{O}$, where $k \leq g$ and $\mathscr{O}$ is a distinguished point of $\mathscr{C}$. We say that this representation is minimal when k is the smallest possible integer. It is possible [GPS02] to refine Corollary 2.13 by showing that a minimal representation is unique. For completeness we give the demonstration.

Proposition 2.14 (Reduced divisors)**.** *For any divisor D of degree 0 on a curve, there exists a unique and minimal representant $P_1 + \cdots + P_k - k\mathscr{O}$ of the class of D in $\mathrm{Jac}(\mathscr{C})$. It is called a reduced divisor along $\mathscr{O}$, and the integer k is called the weight of D.*

Proof. The existence comes from Corollary 2.13. Let D be a degree 0 divisor with $D \sim P_1 + \cdots + P_k - kP_\infty = D_0$, k being minimal. If $k = 0$, then D is a principal divisor, so assume that $k \geq 1$. If $l(D_0 + (k-1)\mathscr{O}) > 0$, there exists a function f and an effective divisor E such that $\mathrm{div}\, f + D_0 + (k-1)\mathscr{O} = E$. In other words, $D_0 \sim E - (k-1)\mathscr{O}$, which contradicts the minimality of k. Hence $l(D_0 + (k-1)\mathscr{O}) = 0$, and since adding a point to a divisor increases the dimension by at most 1, $l(D_0 + k\mathscr{O}) \leq 1$. As $D_0 + k\mathscr{O}$ is effective, its associated space contains the constant, so we find $\mathscr{L}(P_1 + \cdots + P_k) = K$. Let now $E - k\mathscr{O} \sim P_1 + \cdots + P_k - k\mathscr{O}$. By definition there is a function f such that $\mathrm{div}\, f + P_1 + \cdots + P_k = E$, so that $f \in \mathscr{L}(P_1 + \cdots + P_k)$. Thus f is constant, which implies that $E = P_1 + \cdots + P_k$. □

2.1.3 Jacobian Varieties as Abelian Varieties

The Jacobian variety enjoys a group structure, but is also an algebraic variety. More precisely, there exists an algebraic variety whose points identify one-to-one to the elements of $\mathrm{Jac}(\mathscr{C})$, that we now briefly describe. Denote by $\mathscr{C}^{(g)} = \mathscr{C}^g/\mathfrak{S}_g$ its symmetric product, that is to say, the cartesian product of g copies of the curve where the order of the points in a g-tuple is not considered. Assume that $\mathscr{C}$ has at least one point $\mathscr{O}$ and define a map

$$\begin{array}{ccc} \mathscr{C}^{(g)} & \longrightarrow & \mathrm{Jac}(\mathscr{C}) \\ (P_1, \ldots, P_g) & \longmapsto & P_1 + \cdots + P_g - g\mathscr{O} \end{array}$$

Corollary 2.13 shows this map is at least surjective. Observe that if $\mathscr{C}$ is defined over K, then its symmetric product is defined over K. However, there are elements in $\mathrm{Jac}_K(\mathscr{C})$ which cannot be described by K-rational points of $\mathscr{C}$, see Section 2.2.2 for an example. The fundamental result is that this map is compatible with the group law on $\mathrm{Jac}(\mathscr{C})$, and that it can be described by rational maps (informally, once coordinates have been chosen for the variety, rational fractions in those coordinates). Such algebraic varieties are called *Abelian varieties* and are important objects in algebraic geometry. We give a more precise definition.

Definition 2.15. *An Abelian Variety is a projective algebraic group. In other words, it is a projective algebraic variety, endowed with a group law that can be expressed by regular rational functions of the coordinates.*

Geometrically, the group law can be described as follow. Let D_1, D_2 in $\mathrm{Jac}(\mathscr{C})$. For simplicity, we assume that D_1, D_2 have weight g, which means $D_1 = P_1 + \cdots + P_g - gP_\infty, D_2 = Q_1 + \cdots + Q_g - gP_\infty$, and all P_i and Q_i are distinct. We look for a reduced divisor D equivalent to the degree 0 divisor $D_3 = D_1 + D_2$. For simplicity, we assume that D_3 has weight g. In other words, we look for a function $f \in K(\mathscr{C})$ such that $\mathrm{div}\, f = D_3 - D$, so that we need $f \in \mathscr{L}(gP_\infty - D_1 - D_2)$. From Riemann-Roch's theorem, this space has dimension at least 1 and $l(3gP_\infty) = 2g + 1$. Adding a point to a given divisor increases the dimension of the

associated space by at most 1, and from this we deduce that $l(gP_\infty - D_1 - D_2) = 1$. Therefore, f in unique up to a constant. Computing the group law amounts to computing this function and find its remaining zeroes.

The main problem is that it is difficult to determine this function starting from reduced divisors, that is to say, from input points. When the curve is hyperelliptic, a more practical representation allowing for algorithmic computation of this group law is introduced in Section 2.2.2 with *Mumford Representation*. When the curve is elliptic, explicit formula can be obtained straightforwardly by a geometric description of the addition, see Section 2.3.

Cardinal bounds over finite fields When the field of definition is finite, it is obvious that a curve and its Jacobian variety has a finite number of rational points. It is less obvious that there is a general formula to estimate this number. A first result in this direction was obtained by Hasse for elliptic curves. Weil later proposed a generalized conjecture for any projective variety, and proved the results for all curves of a given genus and Abelian Varieties. In this thesis we only need statements for curves and their Jacobian variety.

Theorem 2.16 (Hasse, Weil)**.** *Let $\mathscr{C}$ be an algebraic curve of genus g, defined over $\mathbb{F}_q$. We have the following bounds:*

- $|\#\mathscr{C}(\mathbb{F}_q) - q + 1| \leq 2g\sqrt{q}$.
- $(\sqrt{q}-1)^{2g} \leq \#\mathrm{Jac}_{\mathbb{F}_q}(\mathscr{C}) \leq (\sqrt{q}+1)^{2g}$.

When the genus g is fixed, an interpretation of Theorem 2.16 is that $\#\mathscr{C} = O(q)$ and $\#\mathrm{Jac}(\mathscr{C}) = O(q^g)$. It is often used implicitly in the complexity analyses of Chapter 3.

2.2 Hyperelliptic Curves

2.2.1 Equations for hyperelliptic curves

Traditionally, hyperelliptic curves are defined as degree 2 coverings of the projective line. This presentation can be reformulated in terms of function fields by saying that $\mathscr{H}$ is hyperelliptic when $K(\mathscr{H})$ is a separable degree 2 extension of $K(x) = K(\mathbb{P}^1)$. The natural automorphism of $K(\mathscr{H})$ of order 2 and which fixes $K(x)$ induces the *hyperelliptic involution* on $\mathscr{H}$. For more details, see for example [CF05, Chap. 4]. While this approach is elegant and general, it does not suit straightforwardly a computational point of view — some work and additional material are needed to derive equations. For this reason, we choose to directly define hyperelliptic curves by an affine equation.

Definition 2.17. *Let K be a field and $g \geq 1$ an integer. An imaginary hyperelliptic curve $\mathscr{H}$ of genus g is an algebraic curve that admits an affine equation in the form of*

$$\mathscr{H} \ : \ y^2 + h_1(x)y = h_0(x),$$

with $h_0, h_1 \in K[x]$, $\deg h_1(x) \leq g$ and $\deg h_0(x) = 2g+1$, such that the equations $2y - h_1(x) = 0$ and $yh_1'(x) - h_0'(x) = 0$ have no common solutions.

Other models exist, called *real*, but they are less used in practice and therefore we do not elaborate further on them. The last condition implies that $\mathscr{H}$ has no singular points. Such a model is called an imaginary model for the hyperelliptic curve, and admits a single point at infinity $P_\infty = [0:1:0]$. As a distinguished point P_∞ always exists, there is a canonical way of embedding $\mathscr{H}$ into its Jacobian variety, using the map $\mathscr{H} \longrightarrow \mathrm{Jac}(\mathscr{H}), P \longmapsto P - P_\infty$. In the next Section we will show that this is in fact a bijection if $g = 1$.

Definition 2.18. *Let $\mathscr{H}$ be an hyperelliptic curve defined by an equation as $y^2 + h_1(x)y = h_0(x)$. The application*

$$\begin{array}{rccc} \iota : & \mathscr{H} & \longrightarrow & \mathscr{H} \\ & (x,y) & \longmapsto & (x, -y - h_1(x)) \end{array}$$

is called the hyperelliptic involution. For any $P \in \mathscr{H}$, $\iota(P)$ is called the opposite of P.

It is straightforward to verify that this application is correctly defined and satisfies $\iota^2 = \mathrm{id}$. Moreover this involution is compatible with the addition in the Jacobian: for $P \in \mathscr{H}$, we have $\mathrm{div}(x - x_P) = P + \iota(P) - 2P_\infty$, and then $\iota(P) - P_\infty \sim -(P - P_\infty)$ — this is also why we call $\iota(P)$ the opposite of P. The points such that $\iota(P) = P$ are called Weierstrass points (or sometimes ramification points), and have order 2 in $\mathrm{Jac}(\mathscr{H})$.

We now consider the particular case of genus 2 curves, which are a particular focus of our work.

Proposition 2.19. *Let $\mathscr{H}$ be a smooth genus 2 projective curve. Then there is a double cover $\varphi : \mathscr{H} \longrightarrow \mathbb{P}^1$.*

Proof. Let W be a canonical divisor on $\mathscr{H}$. We can assume wlog. that W is effective. Indeed, by Riemann-Roch's theorem, $l(W) = 2$, so there are two linearly independent functions $g_1 \neq g_2$ that generates $\mathscr{L}(W)$. Thus if $(\lambda_1, \lambda_2) \neq (0,0)$ and $f = \lambda_1 g_1 + \lambda_2 g_2$, then $W' = \mathrm{div} f + W$ is effective and linearly equivalent to W, and we can take $g_1 = 1$. Since $\deg W = 2$, then $W = P_1 + P_2$ for some $P_1, P_2 \in \mathscr{H}$, which means g_2 has two poles P_1, P_2. Then the map $\varphi : \mathscr{H} \longrightarrow \mathbb{P}^1$ given by $\varphi(P) = [g_2(P) : 1]$ and $\varphi(P_i) = [1 : 0]$ is a well-defined non-constant (hence surjective) morphism of curves, which has degree two. □

If we admit the alternate definition of hyperelliptic curves (curves that are double covers of the projective line), then this shows that all genus 2 curves are hyperelliptic.

Short Weierstrass equations in odd characteristic Let K be a field not of characteristic 2, and $\mathscr{H}$ be an hyperelliptic curve of genus g. Using the transformation $(x,y) \mapsto (x, y - h_1(x)/2)$, we can get an isomorphic hyperelliptic curve $\mathscr{H}_w$ with equation in the form of

$$\mathscr{H}_w \ : \ y^2 = h(x),$$

with $h \in K[x]$, $\deg h = 2g + 1$. This equation is called a short Weierstrass model. In this model, singularities happen only if f has roots with multiplicity ≥ 2. The hyperelliptic involution is given by $\iota(x,y) = (x, -y)$. Further in this work, when we consider hyperelliptic curves over fields of characteristic $\neq 2$, we always assume that they are given by a short Weierstrass equations. When $\mathrm{Char}(K) = 2$, a classification for genus 2 curves is known — see Section 6.1.3, which is inspired from [BD04, CJ03].

2.2.2 Arithmetic of hyperelliptic curves: the Mumford Representation

The Mumford representation is an elegant and compact polynomial representation for elements in the Jacobian.

Theorem 2.20 (Mumford Representation)**.** *Let $\mathscr{H} \ : \ y^2 + h_1(x)y = h_0(x)$ be an hyperelliptic curve of genus g defined over K. Any $D \in \mathrm{Jac}_K(\mathscr{H})$ can be uniquely represented by a couple of polynomials $(u,v) \in K[x] \times K[x]$ such that*

- *u is monic.*
- *$\deg v < \deg u \leq g$*
- *$u \,|\, v^2 + h_1(x)v - h_0(x)$.*

Let $D = P_1 + \cdots + P_k - kP_\infty \in \mathrm{Jac}(\mathscr{H})$, $k \leq g$, and let (x_i, y_i) be the coordinates for P_i. We can define $u(x) = \prod_{i=1}^{k}(x - x_i)$, and v by $v(x_i) = y_i$ for all i. When all P_i appear just one time in D, v can be defined by Lagrange interpolation. The third condition in Theorem 2.20 ensures the points described by u and v belong to $\mathscr{H}$. If D has weight g, the Mumford representation states that $\mathscr{L}((g+1)P_\infty - D)$ is spanned by u_D and $y - v_D$.

Example: Let $\mathscr{H} : y^2 = x^5 + 617x^3 + 533x^2 + 150x + 457$ over $\mathbb{F}_{1031}$, and of genus 2. Let $P_1 = (916, 306)$ and $P_2 = (397, 44)$ in $\mathscr{H}$, and define $D = P_1 + P_2 - 2P_\infty$. The Mumford representation of D is $u_D(x) = (x - 916)(x - 397) = x^2 + 749x + 740$, and since the line $y - (664x + 372)$ passes through P_1, P_2, we can take $v_D(x) = 664x + 372$. If (x_i, y_i) are the coordinates for P_i, then $v_D(x_i) = y_i$, so the roots of u_D are roots of $v_D^2 + h_1(x)v_D - h_0(x)$, and the last condition is verified. Consider now the reduced divisor E with representation $u_E(x) = x^2 + 311x + 835, v_E(x) = 277x + 889$. Although u_E has no roots over $\mathbb{F}_{1031}$, we find that $u_E = (x - 145t - 215)(x - 886t - 505)$ in a degree 2 extension $L \simeq \mathbb{F}_{1031}[t]/\langle t^2 - 2t + 14 \rangle$. We indeed verify that the L-rational points $Q_1 = (145t + 215, 987t + 646)$ and $Q_2 = (886t + 505, 44t + 558)$ belong to $\mathscr{H}$, so that $E = Q_1 + Q_2 - 2P_\infty$. This is an example of divisor defined over K by non K-rational points. Using its Mumford representation, checking if a divisor is defined over K is immediate.

Mumford representation also allows a nice description of the group law using Cantor's algorithm (Algorithm 1, [Can97]), and therefore practical implementations of Jacobian varieties for hyperelliptic curves. The idea of the algorithm is to first compute a polynomial representation for the degree 0 divisor $D_1 + D_2$, then to use the divisibility condition in the representation to reduce it until it has correct size.

Algorithm 1 Addition in the Jacobian Variety of Hyperelliptic Curves

Input: $D_1 = (u_1, v_1), D_2 = (u_2, v_2)$ in $\mathrm{Jac}(\mathscr{H})$, and $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$.
Output: The Mumford representation (u, v) of the unique reduced divisor $D = D_1 + D_2$.

$d_1, a_1, b_1 \leftarrow \mathrm{xgcd}(u_1, u_2)$; /* $d_1 = a_1u_1 + b_1u_2$ */
$d, a, b \leftarrow \mathrm{xgcd}(d_1, v_1 + v_2 + h_1)$; /* $d = ad_1 + b(v_1 + v_2 + h_1)$ */
$s_1 \leftarrow aa_1,\ s_2 \leftarrow ab_1,\ s_3 \leftarrow b$;
$u \leftarrow \dfrac{u_1u_2}{d^2},\ v \leftarrow \dfrac{s_1u_1v_2 + s_2u_2v_1 + s_3(v_1v_2 + h_0)}{d} \mod u$;
while $\deg u > g$ **do**
  $u' \leftarrow \dfrac{h_0 - vh_1 - v^2}{u},\ v' = (-h_1 - v) \mod u'$;
  $u \leftarrow u',\ v \leftarrow v'$;
end while
return $\left(\dfrac{u}{\mathrm{LC}(u)}, v\right)$

We give a brief description of what is done in the loop, where for simplicity we assume that $\gcd(u_1, u_2) = 1$, so that $d = d_1 = 1$. Before the loop, v is the solution modulo u of the chinese remainders system $v \equiv v_i \mod u_i$. If $\deg u \leq g$, then (u, v) is the Mumford Representation of $D_1 + D_2$. If $\deg u = m > g$, then the divisor $D_3 = D_1 + D_2$ is not reduced and can be written $D_3 = P_1 + \cdots + P_m - mP_\infty$. As $\mathrm{div}(y - v) = D_3 + Q_1 + \ldots + Q_l - lP_\infty$ for some $0 < l < m$, we have $D_3 \sim D_4 := \iota(Q_1) + \cdots + \iota(Q_l) - lP_\infty$, which has smaller weight. Thus we want to find polynomials describing D_4. On the one hand, the roots of u in $\overline{K}$ are roots of $v^2 + h_1v - h_0$, hence there is $u' \in K[x]$ such that $uu' = v^2 + h_1v - h_0$. On the other hand, we check that $v^2 + h_1v - h_0 = (y - v)(y + h_1 + v)$. This means that $\mathrm{div}(u) + \mathrm{div}(u') = \mathrm{div}(y - v) + \mathrm{div}(y + h_1 + v)$, so that u' and $-h_1 - v \mod u'$ are what we need. Figure 2.1 gives a geometric intuition for a genus 2 curve with equation $y^2 = f(x)$ defined over the reals.

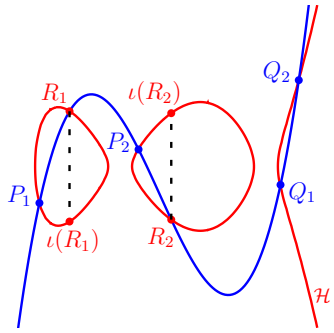


Figure 2.1: Geometric visualization of the addition.
The blue cubic f gives a principal divisor $\operatorname{div} f = P_1 + P_2 + Q_1 + Q_2 + R_1 + R_2 - 6P_\infty$, so we define $(P_1 + P_2 - 2P_\infty) + (Q_1 + Q_2 - 2P_\infty) = (\iota(R_1) + \iota(R_2) - 2P_\infty)$

Example: Consider the divisors $D = P_1 + P_2 - 2P_\infty = (u_D, v_D)$ and $E = Q_1 + Q_2 - 2P_\infty = (u_E, v_E)$ of the previous example. We want to compute the Mumford representation (u_F, v_F) of the sum $F = D + E$. Let $u = u_D u_E$. The extended euclidean algorithm gives $(1019x + 573)u_D + (12x + 559)u_E = 1$, and we obtain $v = (1019x + 573)u_D v_E + (12x + 559)u_E v_D \mod u = 43x^3 + 441x^2 + 176x + 332$. We verify that $y - v$ is the equation of the cubic passing through P_1, P_2, Q_1, Q_2. Such a function has a pole of order 6 at infinity, hence the divisor $\operatorname{div} f - D - E = R_1 + R_2 - 2P_\infty$ has weight 2 and is therefore reduced. The polynomial $h_0 - vh_1 - v^2$ is in fact the resultant of $y - v$ and the curve's equation with respect to y, and thus describe the abscissae of all the intersection points. In particular, $x(R_1)$ and $x(R_2)$ are the roots of $\dfrac{h_0 - vh_1 - v^2}{u} = 213x^2 + 231x + 662$, that we make monic to find $u_F(x) = x^2 + 974x + 158$. From the previous paragraph, if we take $v' = v \mod u_F$, then $v'(x(R_i)) = y(\iota(R_i))$, so we have to account for the action of the hyperelliptic involution, and set $v_F = -v \mod u_F = 548x + 902$. To verify our result, we compute the roots of u_F and evaluate v_F at their values, to find that $R_1 = (418t + 126, 182t + 873)$ and $R_2 = (613t + 962, 849t + 206)$ are L-rational points of $\mathscr{H}$. Those points are also zeroes of $y - v$.

Complexity analysis and improvements We use the notations of Algorithm 1. With overwhelming probability, $\deg u_1 = \deg u_2 = g$. The degree of u is maximal when $\gcd(u_1, u_2) = 1$, in which case $\deg u = 2g$. By construction, $\deg v = 2g - 1$ at worst, so that $\deg u' = 4g - 2 - 2g = 2g - 2$ at worst in the first round of the loop. This means at most $\lceil g/2 \rceil$ rounds are done. All the polynomials considered by the algorithm have degree less than g, so every operation can be done in $O(g^2)$. Improvements for special cases, characteristic and curves have been proposed, see for example [BSSC05] and [CF05, Chap. 14]. Geometric approaches have been also designed [CL11, HC14], and recent works on genus 2 curves using *theta functions arithmetic* and Kummer surfaces [Gau07, GL09, CCS15, BCHL16] have lead to very efficient arithmetics.

2.3 Elliptic Curves

2.3.1 Weierstrass models for elliptic curves

Definition 2.21 (Elliptic Curves)**.** *An elliptic curve defined over K is a projective curve of genus 1 defined over K with at least one K-rational point $\mathscr{O}$.*

The fundamental property of elliptic curves is that they identify to their Jacobian variety, and that the group law transfers from the Jacobian to the curve.

Proposition 2.22. *Let E be an elliptic curve defined over K with distinguished K-rational point $\mathcal{O}$. The application $\varphi_{\mathcal{O}} : E \longrightarrow \mathrm{Jac}(E)$ defined by $\varphi_{\mathcal{O}}(P) = P - \mathcal{O}$ is a bijection, and E is an abelian group with neutral element $\mathcal{O}$.*

Proof. Surjectivity comes from Corollary 2.13. Let P, P' in E such that $P - \mathcal{O} \sim P' - \mathcal{O}$. By definition there exists a function f such that $\mathrm{div}\, f = P - P'$, or in other words, there is a function in $\mathscr{L}(P')$. From Riemann-Roch theorem, this space has dimension 1, and since it contains the constants, it is in fact the space of constant functions. But if f is constant, then $\mathrm{div}\, f = 0$ or equivalently, $P = P'$ and $\varphi_{\mathcal{O}}$ is one-to-one. Using again Corollary 2.13 it is straightforward to verify that the application $\sigma : E \times E \longrightarrow E$ defined by $\sigma(P,Q) = \varphi_{\mathcal{O}}^{-1}(\varphi_{\mathcal{O}}(P) + \varphi_{\mathcal{O}}(Q))$ transfers the group structure of $\mathrm{Jac}(E)$ on E. □

Definition 2.21 is enough to find an affine equation for any elliptic curve. Let K be a field and E an elliptic curve defined over K. From Riemann-Roch's Theorem 2.12, and any integer $n \geq 1$, we have $l(n\mathcal{O}) = n$. From this we deduce that $\mathscr{L}(\mathcal{O}) = K$, that there is a nonconstant function x such that $\{1, x\}$ is a K-basis of $\mathscr{L}(2\mathcal{O})$, and a nonconstant function $y \neq x$ such that $\{1, x, y\}$ is a K-basis of $\mathscr{L}(3\mathcal{O})$. This implies that $\{1, x, y, x^2\}$ is a K-basis of $\mathscr{L}(4\mathcal{O})$ and that $\{1, x, y, x^2, xy\}$ is a K-basis of $\mathscr{L}(5\mathcal{O})$. Now the space $\mathscr{L}(6\mathcal{O})$ has dimension 6 and contains the seven functions $1, x, x^2, x^3, xy, y, y^2$. This means there is a linear relation among them, whose coefficients in x^3 and y^2 cannot vanish. Up to scalar multiplication on x and y, the relation has the form of

$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6, a_i \in K. \tag{2.1}$$

This defines an affine plane curve E', with projective closure $\overline{E'} \subset \mathbb{P}^2$ given by

$$\overline{E'} \ : \ Y^2Z + a_1XYZ + a_3XZ^2 = X^3 + a_2X^2Z + a_4XZ^2 + a_6Z^3.$$

It has a single point at infinity $[0:1:0]$, which is in fact $\mathcal{O}$. It is then possible to show that $\overline{E'}$ is isomorphic to E. As we mainly consider affine models, we may assume any elliptic curve admits an equation as (2.1). Smoothness of the curve is achieved if the partial derivatives $2y + a_1x + a_3$ and $a_1y - (3a_2x^2 + a_4)$ do not vanish simultaneously at a point $(x_1, y_1) \in E$.

Remark 2.23. *From Equation 2.1 and its affine version and Definition 2.21, it seems that we can see elliptic curves as hyperelliptic curves of genus 1. While this point of view is not problematic for any of our contributions, it hides a fundamental structural difference between hyperelliptic and elliptic function fields, see [CF05, Chap.4-4, p.73].*

The general equation for an elliptic curve is not the most convenient for computational purpose. Sparser equations are well-known. We omit the case of characteristic 3 as we never consider it in our applications.

Proposition 2.24 (Short Weierstrass equations for Elliptic curve)**.** *Let $E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ be an elliptic curve defined over a field K with $Char(K) \neq 3$. There exist linear changes of variables such that E admits an equation in the form of*

- $y^2 = x^3 + a_4x + a_6$ *if* $Char(K) \neq 2$;
- $y^2 + xy = x^3 + a_2x^2 + a_6$ *if* $Char(K) = 2$ *and* $a_1 \neq 0$
- $y^2 + a_3y = x^3 + a_4x + a_6$ *if* $Char(K) = 2$ *and* $a_1 = 0$

Elliptic curves with equation like the last one are called *supersingular*. They are known to be weak for any Discrete logarithm purpose [FMR94] but were popular in pairing-based cryptography because their Tate pairing is easy to compute. The elliptic involution is often denoted $[-] : E \longrightarrow E$ and defined as $[-](x,y) = (x,-y)$ in odd characteristic or $[-](x,y) = (x, y+x)$ or $(x, y+a_3)$ in even characteristic. For a short Weierstrass equation in odd characteristic, the roots of $x^3 + a_4x^2 + a_6$ give three points of order 2, that is to say points fixed by the elliptic involution. Such points are also called 2-torsion points.

2.3.2 Arithmetic of Elliptic Curves

The geometric interpretation of the group law on an elliptic curve is well-known as the *chord-tangent* method. We first briefly recall how it is done. Let E be an elliptic curve with point at infinity $\mathscr{O}$, and let P_1, P_2 be two points of E. Because E has an equation of degree 3, the line through P_1, P_2 must cut E in a third point Q. In term of divisor, $P_1 + P_2 + Q - 3P_\infty$ is principal, hence 0. As $Q \sim [-]Q$, it is natural to define an addition on E as $P_1 \oplus P_2 = [-]Q$. There are some particular cases — see also Figure 2.2 for a visualization over the reals:

- when $P_1 = \mathscr{O}$, we define $P_1 \oplus P_2 = P_2$;
- when $P_1 = [-]P_2$, we define $P_1 \oplus P_2 = \mathscr{O}$;
- When $P_1 = P_2$, the line is the tangent at P_1 for E, and $P_1 \oplus P_2 = 2P_1$.

It is known (and proved for example in [Sil13, Chap.3]) that this geometric law is the same as the group law from $\mathrm{Jac}(E)$, when it is transferred on E as in Proposition 2.22. For this reason, we often denote $P_1 \oplus P_2$ directly as $P_1 + P_2$ and $-P$ stands either for the image by the elliptic involution or the opposite for the group law.

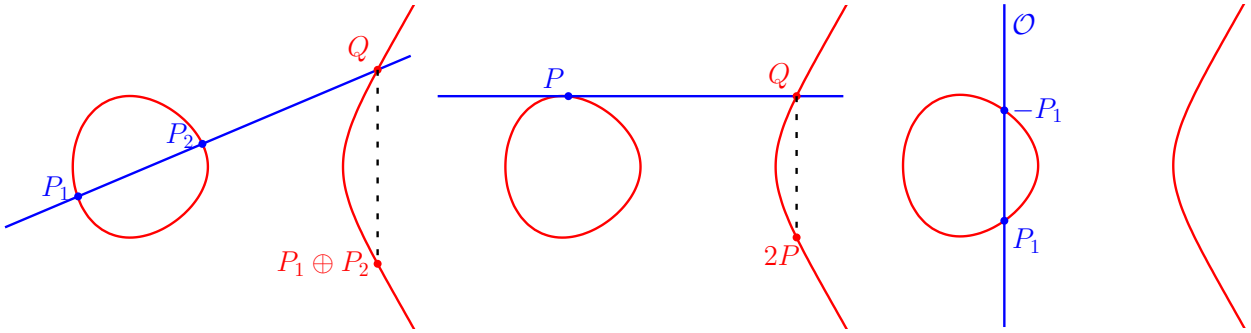


Figure 2.2: Group law for $y^2 = x^3 + Ax + B$ over the reals.

Formulae can be deduced once an equation $E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$ is given. The computations are identical to that of Cantor's algorithm 1 for genus 1 curve if we consider that a point has a Mumford representation as $(x - x(P), y(P))$. We now explicit those formulae, omitting the special cases. Let $P_1(x_1, y_1)$ and $P_2(x_2, y_2)$ be points of E, and $P_3 = P_1 + P_2 = (x_3, y_3)$. We also let λ be the slope of the line through P_1, P_2 or the slope of the tangent of E at P:

$$\text{Addition: } \lambda = \frac{y_2 - y_1}{x_2 - x_1} \quad \text{Doubling: } \lambda = \frac{3x_1^2 + 2a_2x_1 + a_4 - a_1y_1}{2y_1 + a_1x_1 + a_3}$$

Then the coordinates of the sum are given by

$$\begin{cases} x_3 = \lambda^2 + a_1\lambda - a_2 - x_1 - x_2 \\ y_3 = \lambda(x_1 - x_3) - y_1 - a_1x_3 - a_3. \end{cases}$$

Improvements, special models Several elliptic curves have been selected as standards for cryptographic primitives. Indeed, the litterature on their arithmetic is extremely rich and various approaches have been proposed to give faster, more efficient and secured arithmetic. A close-to-exhaustive (practical) listing can be found on the url `http://hyperelliptic.org/EFD/`, and we give some explanations on what can be found there.

Fastest arithmetic can be designed using (twisted) Edwards curves, see [Edw07, BBJ+08, BLF08]. These models have complete addition law (in the sense that there are no particular cases), and require fewer operations than curves in short Weierstrass models. Adding torsion points and computing involution amount to sign changes. These models are in fact really

close to some theta models. In particular, this link was used in even characteristic to define binary Edwards curves [Fou13]. Another interesting family of curves is the class of (twisted) Hessian curves [Sma01, BCKL15], which come close to Edwards curves in term of speed.

For side-channel resistant implementations of the scalar multiplication, that is to say, computations of mP for some fixed P, the Montgomery ladder [Mon87] can be used. While the algorithm does not depend on the model of the curve, an efficient implementation requires some properties on the addition law. Curves with said properties are called Montgomery Curves. Any Montgomery curve can be transformed into a (twisted) Edwards curve and reciprocally.

Chapter 3

The Discrete Logarithm Problem in Jacobian Varieties

This Chapter gives a State-of-the-Art on the Discrete Logarithm Problem. This problem can be stated as such: given an abelian group $(G, \times)$ and $g, h \in G$, find, if it exists, an integer x such that $g^x = h$. Variants of the problem exist, for example if G is known to be cyclic with order N and g is a generator — it is the case in most practical applications. The first Section presents the DLP as the inverse problem of the group exponentiation. It also introduces applications of the DLP in cryptography, with the *Diffie-Hellman key exchange*, the *ElGamal encryption* and the *Digital Signature Agorithm (DSA)*. The next Sections all deal with known methods to compute Discrete Logarithms. In Section 3.2, *generic* attacks are presented. In this context, it is assumed that only the group law can be used, which is precised in Definition 3.4. In the generic case, it is known that a generic algorithm needs a number of arithmetic operations at least proportional to the square root of the group order to solve a DLP instance ([Nec94, Sho97], see Theorem 3.5 for a precise statement). As the size of a group can be carefully chosen, the DLP is generically a hard problem.

To achieve better asymptotic complexities and practical run time for DLP computations, *specialized* approaches must be used. The term "specialized" implies that the richer structure (algebraic equations, underlying geometry, ...) surrounding the group is exploited to compute Discrete Logarithms. Indeed, in practice, the target group is the group of rational points of an elliptic curve defined over a finite field. This second situation is a particular case of the Discrete Logarithm Problem over the Jacobian Variety of an algebraic curve, the main topic of this thesis and therefore of the remaining Sections. A popular specialized approach to the DLP is given by the family of *Index Calculus* algorithms. Such algorithms run in mainly two phases: in a first one, an overdetermined linear system linking discrete logarithms of selected elements of the group is built. As a particular focus of this work, we refer to this phase as the *harvesting* or *relations collection*. The linear system is then solved in a second phase. Section 3.3 gives more details.

Several variants of the harvesting exist, and for certain families of group, have culminated in subexponential algorithms [ADH99], with proper analysis in [Eng02, EG02, ES02], and [EGT11], detailed in Section 3.3. However, the parameters targeted by those algorithms are not practical. In fact, practical applications only consider elliptic curves ($g = 1$), which are already widely used [LM10, NIS99, Res10]. Recent works on Jacobian arithmetics [Gau07, GL09, LR16] suggest that genus 2 curves can be competitive with elliptic curves, and could therefore be considered for future standards. Still, other algebraic curves must not be ruled out of the picture: *Transfer attacks* [Die03, GHS02] can transfer DLP instances on certain Elliptic Curves to equivalent instances on potentially weaker curves, usually with higher genus and defined on different finite fields. Such attacks are briefly covered in Section 3.4.1. The

main idea is to use the concept of Weil Restriction of an algebraic variety defined over an extension of a finite field (defined in Section 1.5.2). This approach leads to another variant of the harvesting, called *Decomposition Attacks* and pioneered by Gaudry and Diem in [Die11, Gau09]. Decomposition attacks are concerned with the major part of our contributions, and are therefore emphasized in this Chapter.

3.1 Discrete Logarithm Problem and Cryptography

3.1.1 Exponentiation and Discrete Logarithm

Public-key cryptography is based on *one-way functions*. A function is said to be one-way when computing its value with given inputs can be done quickly in term of arithmetic operations while, on the contrary, recovering the inputs from the result cannot be done easily. Among public key cryptosystems, the two most famous one-way functions are the Discrete Logarithm [DH76] and RSA [RSA78]. For the latter, computing the product of two (very large) prime numbers can be done very efficiently, whereas obtaining the factorization when the product is given is usually a challenging task. The Discrete Logarithm involves an Abelian group $(G,\times)$. For a given $g \in G$ and $x \in \mathbb{N}$, the *exponentiation* g^x can be computed very efficiently, thanks to the *Square-and-Multiply* algorithm and assuming x can itself be computed very efficiently. We give one version in Algorithm 2.

Algorithm 2 Square-and-Multiply

Input: A element g of an abelian group $(G,\times)$, and $x \in \mathbb{N}$.
Output: The exponentiation g^x in G.

$l \leftarrow \log_2(x)$ and $i \leftarrow l-1$;
Get binary expansion of $x = (x_{l-1}\dots x_0)_2$;
$a \leftarrow 1_G$.
while $i \geq 0$ **do**
  $a \leftarrow a^2$;
  if $n_i = 1$ **then**
    $a \leftarrow a \times g$;
  end if
  $i \leftarrow i-1$;
end while
return a

It is straightforward to see that the loop is entered $\log_2(x)$ times, and that the worst case happens when all bits of x are 1. If G has known order N of bitsize t, we can assume for simplicity that the exponent x has at most size t. If an operation of the group law is computed in $M(t)$ operations, Algorithm 2 needs at most $2tM(t)$ operations in the group to terminate. This gives a complexity of $O(tM(t))$, which is linear in term of group multiplications, and at most cubic in bitsize t for practical groups. Several variants exists (see for example [CF05, Chap. 9]) depending on the constraints.

Definition 3.1 (Discrete Logarithm)**.** *Let $(G,\times)$ be an abelian group, and $g \in G$ of order N. Let $h \in \langle g \rangle$. The discrete logarithm of h in base g is the unique integer $0 \leq x \leq N-1$ such that $g^x = h$. It is denoted by $x = \log_g(h)$.*

Thanks to Definition 3.1, the Discrete Logarithm Problem can now be seen as the inverse problem of the Exponentiation.

Problem 3.2 (Discrete Logarithm Problem (DLP))**.** *Let $(G,\times)$ be an abelian group, and g,h be in G. The General Discrete Logarithm Problem is to find, if it exists, an integer x such that $g^x = h$.*

Remark 3.3. *In practice, we usually have $G = \langle g \rangle$ of order N. The DLP is then to find the unique integer $0 \leq x \leq N-1$ such that $g^x = h$.*

Before turning to applications in cryptography, it remains to understand why the Discrete Logarithm Problem is (generically) considered a difficult problem to solve.

Definition 3.4 (Generic Algorithm)**.** *An algorithm performing operation over a group is said to be generic when it can only performs the following operations:*

- *computing the product of two group elements;*
- *computing the inverse of a group element;*
- *checking if two group elements are equal.*

Considering only generic algorithms over abelian groups, the next theorem assesses the hardness of the DLP in general. The statement implicitly uses Pohlig-Hellman reduction ([PH78], see also Section 3.2.1).

Theorem 3.5 ([Nec94, Sho97])**.** *Let $G = \langle g \rangle$ be an abelian group of order N, $h \in G$, and let p be the greatest prime divisor of N. Then a generic algorithm needs at least $\Omega(\sqrt{p})$ group operations to compute the Discrete Logarithm of h in base g.*

Hence, over a group with a prime cardinality of 160 bits, a generic algorithm needs at least 2^{80} operations to compute a discrete logarithm. If we consider only generic algorithms, and $\log_2(p) = 256$, then recommended cryptographic security is achieved. This has to be compared with RSA key-size, for which the recommandation is now at least 4096 bits. Theorem 3.5 is already enough to understand the generic difficulty of the DLP, but the given bound is in fact tight, as it is reached by several algorithms — see Section 3.2 — and so the conclusion could be reformulated with a $\Theta(\sqrt{p})$ instead.

The special case $G = (\mathbb{Z}/N\mathbb{Z}, +)$ Here the DLP rewrites as $xg \equiv h\ [N]$, and we assume for simplicity that g is a generator. This also means that $\mathrm{GCD}(g, N) = 1$, equivalently that g is invertible modulo N. Its inverse g^{-1} can be computed in quadratic time in $\log_2(N)$ using Extended Euclidean Algorithm and satisifies $x \equiv hg^{-1}\ [N]$. Hence solving the Discrete Logarithm Problem in $(\mathbb{Z}/N\mathbb{Z}, +)$ amounts to computing an inverse modulo N. This is an example of a non-generic algorithm because the ring structure has been used in the computation of the inverse. Recall that if a given group G is known to be cyclic of order N with generator g, then it is isomorphic to $(\mathbb{Z}/N\mathbb{Z}, +)$. From $\mathbb{Z}/N\mathbb{Z}$ to G this is simply the exponentiation $i \mapsto g^i$. However, describing the inverse map amounts to solve the discrete logarithm in G as we want to define $h \mapsto \log_g(h)$. Hence describing the isomorphism is equivalent to solve the discrete logarithm problem in G.

3.1.2 DLP Based Cryptosystems

Diffie-Hellman's key exchange Presented in 1976 and following Merkle's ideas, the Diffie-Hellman Key Exchange [DH76] allows two parties to share a common secret — typically a key for a symmetric cipher. This started asymmetric cryptography and gave a first answer to the problem of key distribution between users. It is used as an essential component in cryptographic protocols such as SSL or TLS. Traditionally, the two parties are named Alice and Bob, and an ennemy Eve is spying them. Alice and Bob first agree publicly on a cyclic group $G = \langle g \rangle$ or order N, hence the tuple (G, g, N) is known. The exchange is then described by the scheme below.

Alice	(G,g,N)	*Bob*
$a = Rand(N)$		$b = Rand(N)$
$A = g^a$		$B = g^b$
	$A \rightarrow$	
	$\leftarrow B$	
$K = B^a$		$K = A^b$

Table 3.1: Diffie-Hellman Key Exchange

In Table 3.1 the function $Rand(N)$ stands for a cryptographic Pseudo-Random Number Generator (PRNG) returning an positive integer smaller than N. The *Computational Diffie-Hellman Problem* (CDHP) is the following: given (G,g,N) and g^a, g^b, compute $K = g^{ab}$. It is straightforward to see that if Eve has an efficient algorithm to solve the DLP, then she can solve as efficiently the CDHP. The converse is not known, but in special cases both problems are equivalent [dB88, Mau94]. In practice, groups of rational points of standardized elliptic curves are used and the protocol is then called *Elliptic Curve Diffie-Hellman* (ECDH).

El-Gamal Encryption El-Gamal encryption is a scheme that allows a user to send encrypted data to the owner of a public key. To decrypt the data it is necessary to know the secret key associated with the public key. Alice chooses again a cyclic group $G = \langle g \rangle$ of order N, a random integer $x \leq N$ and compute $h = g^x$. Then she publishes her public key $\text{PK} = (G,g,N,h)$ and keeps her secret key $\text{SK} = x$. Assuming that Bob has a way of mapping data to elements of G, he can now send data encrypted using the public function ENC_{PK}. Alice can read such data using her decryption function DEC_{SK}. Both functions are described in Algorithms 3 and 4.

Algorithm 3 El-Gamal Encryption ENC_{PK}

Input: A message $m \in G$.
Output: A ciphertext $c = (c_1, c_2)$.

$t \leftarrow Rand(N)$; /* t: ephemeral key */
$c_1 \leftarrow g^t$;
$s \leftarrow h^t$;
$c_2 \leftarrow m \times s$;
return (c_1, c_2)

Algorithm 4 El-Gamal Decryption DEC_{SK}

Input: A tuple (c_1, c_2) in $G \times G$.
Output: A message $m \in G$.

$s \leftarrow c_1^x$;
$i \leftarrow s^{-1}$;
$m \leftarrow c_2 \times i$;
return m

For any $m \in G$, we have $\text{DEC}_{\text{SK}}(\text{ENC}_{\text{PK}}(m)) = m$, since $c_2 \times i = (m \times (g^x)^t) \times (g^{-t})^x = m$. If Eve can efficiently solve the DLP, then she can decrypt all data sent to Alice, as she can recover the secret key x. Notice that $s = h^t = (c_1)^x = g^{xt}$ is a shared value between Alice and Bob during decryption of (c_1, c_2) and that Eve knows $h = g^x$ and $c_1 = g^t$. Hence, if Eve can efficiently solve the CDHP, she can decrypt the message contained in c_2. Using only El-Gamal encryption, Alice has no way to make sure that Bob is indeed the sender of data. Achieving this is known as *Signature* and can be done with a Discrete Logarithm based protocol that we now describe.

Digital Signature Algorithm (DSA) The Digital Signature Algorithm allows to certify the sender of a message. It is currently widely used (for example in SSL/TLS protocols), in particular in its Elliptic Curve variant (ECDSA). Alice and Bob first agree publicly on a cyclic group $G = \langle g \rangle$, of prime order p, and a *cryptographic hash function* H — definitions

can be found for example in [CF05, Chap. 1, p. 12] or [Jou09, Chap 1. p. 21-22]. For the sake of simplicity we assume that the ouput of H is an integer modulo p. If Bob wants to send a message m to Alice and certify that the message is his, he chooses a random integer x and compute $h = g^x$. He then publishes his public key $\text{PK} = h$ and keeps the secret key $\text{SK} = x$. Overall, a third party Eve knows (G, g, p, H, h). Before sending his message, he generates a *signature* using Algorithm 5.

Algorithm 5 DSA Signature Generation

Input: A message m and the public key (G, g, p, H, h).
Output: A signature $(r, s) \in \mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/p\mathbb{Z}$.

$r \leftarrow 0, s \leftarrow 0$;
repeat
  $z \leftarrow \text{H}(m)$;
  $t \leftarrow Rand(p)$;
  $r \leftarrow \text{H}(g^t)$;
  $s \leftarrow t^{-1}(z + rx) \mod p$;
until $r \neq 0$ **and** $s \neq 0$
return (r, s)

Now Bob sends his message m together with the signature (r, s). Alice can verify that Bob is inded the sender by using the next Algorithm. For the sake of simplicity Algorithm 6 does not check if its inputs are valid elements.

Algorithm 6 DSA Signature Verification

Input: A message m, a public key (G, g, p, H, h) and a couple $(r, s) \in \mathbb{Z}/p\mathbb{Z} \times \mathbb{Z}/p\mathbb{Z}$.
Output: ACCEPT or REFUSE.

$z \leftarrow \text{H}(m)$;
$w \leftarrow s^{-1} \mod p$;
$u_1 \leftarrow zw \mod p$;
$u_2 \leftarrow rw \mod p$;
$T \leftarrow \text{H}(g^{u_1} \times h^{u_2})$;
if $T = r$ **then**
  return ACCEPT
else
  return REFUSE
end if

As H is a cryptographic hash function, to check the validity of Algorithm 6 it is enough to check that $g^{u_1} \times h^{u_2} = g^t$. This is the case, as we have $s^{-1} = t(z + rx)^{-1}$ so that

$$\begin{aligned} g^{u_1} \times h^{u_2} &= g^{s^{-1}z} \times g^{s^{-1}rx} \\ &= g^{s^{-1}(z+rx)} \\ &= g^t. \end{aligned}$$

3.2 Generic Algorithms to compute Discrete Logarithms

Since the target group is finite, the most naive way to compute a discrete logarithm is to enumerate the power of the given generator until the challenge is found. For a group G of order N, in the worst case this needs N operations in the group, which gives a complexity

exponential in the bitsize of N. This approach can therefore only be used when the target group is really small. This can happen when N admits small prime factors, using Pohlig-Hellman reduction described in the next Section. Starting from Section 3.2.2 we assume that G is cyclic of prime order p. We then present two *Square root* algorithms, respectively *Baby-Step-Giant-Step* and Pollard's ρ*-Method*. The terminology describes their asymptotic time complexity in $O(\sqrt{p})$. In particular, those two algorithms tell that Shoups's generic lower bound of Theorem 3.5 is tight.

3.2.1 Pohlig-Hellman reduction

Pohlig-Hellman reduction [PH78] allows to reduce the asymptotic complexity of the DLP in an abelian group G of order N to the asymptotic complexity of DLP instances in a subgroup of order p, where p is the biggest prime factor of N. This is done in two steps. First, let w.l.o.g[1] $G = \langle g \rangle$ be a cyclic group of order $N = p_1^{\alpha_1} \dots p_n^{\alpha_n}$, with $p_1 < \dots < p_n$, where the p_i are all prime numbers. It is known that there is a group isomorphism $G \simeq \prod_{i=1}^{n} G_i$, with $G_i = \left\langle g^{N/p_i^{\alpha_i}} \right\rangle \simeq \mathbb{Z}/p_i^{\alpha_i}\mathbb{Z}$. Thus for all i, we can let $g_i = g^{N/p_i^{\alpha_i}}$ be a generator of G_i. Let now $h \in G$, so that there exists $x \leq N-1$ such that $h = g^x$. Let $h_i = h^{N/p_i^{\alpha_i}}$. Since h_i has order a power of p_i, then h_i must belong to G_i. Moreover we have $h_i = g^{xN/p_i^{\alpha_i}} = g_i^x$, and since G_i has order $p_i^{\alpha_i}$, this gives $h_i = g_i^{x_i}$ with $x \equiv x_i \ [p_i^{\alpha_i}]$. If all the x_i are known, that is to say, if the discrete logarithms of all h_i's in base g_i are known, then x can be recovered by solving the Chinese Remainder System

$$\begin{cases} x \equiv x_1 \ [p_1^{\alpha_1}], \\ \vdots \\ x \equiv x_n \ [p_n^{\alpha_n}]. \end{cases}$$

Asymptotically, the dominant part of this reduction is the computation of $\log_{g_n}(h_n)$, and overall we have reduced the DLP instance in G to a DLP instance in a cyclic group of order a power of a prime.

A second reduction can be obtained. We now assume that $G = \langle g \rangle$ has order p^α, and that we are given $h = g^x$ for some $x \leq p^\alpha - 1$. The idea is to write the expansion of x in base p, that is to say, to consider $x = x_0 + x_1 p + \dots + x_{\alpha-1} p^{\alpha-1}$, with $0 \leq x_i < p$. Next we observe that $h^{p^{\alpha-1}} = g^{x_0 p^{\alpha-1}}$ belongs to the subgroup $\left\langle g^{p^{\alpha-1}} \right\rangle$ of order p. Hence we can find x_0 by solving a DLP instance in a cyclic subgroup of order p. The previous observation leads to $h^{p^{\alpha-2}} = g^{x_0 p^{\alpha-2} + x_1 p^{\alpha-1}}$, therefore we have $h^{p^{\alpha-2}} g^{-x_0 p^{\alpha-2}} = (g^{p^{\alpha-1}})^{x_1}$, so that x_1 can be found once x_0 is known by solving a DLP instance in the same subgroup of order p. We can continue inductively until all x_i's have been found. Overall, α DLP instances in a group of order p have to be solved; asymptotically, to find x thus amounts to solving a DLP instance in a subgroup of order p. The next theorem sums up this section.

Theorem 3.6 (Pohlig-Hellman, [PH78])**.** *Let G be a cyclic group of order $N = p_1^{\alpha_1} \dots p_k^{\alpha_k}$, with $p_1 < \dots < p_k$ primes numbers. Let $c(p)$ be the of operations to solve a DLP instance in a cyclic group of order p. Then the number of operations to solve DLP instance on G can be bounded by $O(\alpha_k c(p_k))$.*

Practical impact Because of Theorem 3.6, the biggest prime factor of a practical DLP group must be long enough bitwise. The next Sections shows that the generic bound given by Theorem 3.5 is tight, with $c(p) = \sqrt{p}$.

[1] We assume a practical DLP context, where the challenge lies in the the group generated by the input g.

3.2.2 Baby-Step-Giant-Step

The Baby-Step-Giant-Step (BSGS) method is a "smart brute force" attack proposed by Shanks, and more precisely a time-memory tradeoff that uses collisions between two lists to compute a discrete logarithm in the target group. From the previous Section, we can now assume that we are given $G = \langle g \rangle$ of prime order p and a challenge $h = g^x$, even if the method works for any finite abelian groups. The key idea is to take a integer $y < p$ and to write the Euclidean division of x by y. Indeed, there exists two integers q, r such that $x = qy + r$, and we also know that $0 \leq r < y$. Since G has order p, we know that $x \leq p - 1$, so this implies that $0 \leq q \leq \lfloor \frac{p}{y} \rfloor$. Turning now to elements of G, we have that

$$h = g^x = g^{qy+r} \Leftrightarrow hg^{-r} = (g^y)^q. \tag{3.1}$$

To find the quotient q and rest r, we can enumerate all possibilities in two lists:

- the list of *baby-steps*: $l = \{h, hg^{-1}, \ldots, hg^{-(y-1)}\} = \{hg^{-i} \ : \ 0 \leq i \leq y-1\}$.
- the list of *giant-steps*: $L = \{1, g^y, \ldots, g^{\lfloor p/y \rfloor y}\} = \{(g^y)^i \ : \ 0 \leq i \leq \left\lfloor \frac{p}{y} \right\rfloor\}$.

Equation 3.1 guarantees that $l \cap L$ has one element, which can then be used to find the discrete logarithm. To balance the construction of both lists, we notice that g^{-1} and g^y can be precomputed once y is fixed, and that they should have the same number of elements so that building one is not way longer than building the other. This is achieved when $y = \lceil \sqrt{p} \rceil$. Building l and L then costs $2\sqrt{p}$ multiplications in the group, and costs $2S\sqrt{p}$ in memory size, where S is the bitsize of a group element. Finding the collision can be done naively by sorting both lists (in $O(\sqrt{p} \log \sqrt{p})$) and a naive complexity is then given by the cost of the sorting.

It is clear that a more efficient approach can be used. First, it is enough to build only one list. Elements of the other can be computed (but not stored) until the collision is found. This saves $S\sqrt{p}$ in memory. The sorting can be avoided by using hash tables indexed with the elements of the list: testing if an element is already in the built list is then done in constant time. This leads to an overall complexity of $O(\sqrt{p})$, which is the generic bound (Theorem 3.5), and the steps are summed up in Algorithm 7. Several variants exist, see for example [CF05, Chap. 19]

3.2.3 Pollard's ρ-Method

We now describe a probabilistic algorithm that achieves an asymptotic complexity of $O(\sqrt{p})$ for computing discrete logarithms in a cyclic group of prime order p. It is based on the forced periodicity of a *random walk* among a finite set — here, because it lies in the target group — and the square root complexity is achieved with the *birthday paradox*. The name originates from the classic representation of a periodic sequence. Assume for a moment that we are given the target group $G = \langle g \rangle$ together a function F which maps "uniformly at random" G to itself and a challenge $h = g^x$. A random-walk can be defined by $u_{i+1} = F(u_i)$. Since the group is finite, there exist an integer i_0 and $t > 0$ such that $u_i = u_{i+t}$ for all $i \geq i_0$, that is to say, the random-walk is periodic after a certain time. This can be used to deduce the discrete logarithm of h by storing each steps of the random walk until a collision is achieved, as we now describe.

Pollard suggested the following function to emulate a random walk, and experimentally its behaviour is indeed close enough to a random behaviour. First partition G into three subsets G_1, G_2, G_3. A close to uniform way to partition the group can be obtained by considering fixed bits in the machine representation of elements of G. Starting from $u_0 = g^{a_0} h^{b_0}$ for randomly

Algorithm 7 Computing Discrete Logarithms with Baby-Step-Giant-Step in $O(\sqrt{p})$

Input: A cyclic group $G = \langle g \rangle$ of prime order p and $h \in G$.
Output: The discrete logarithm $x = \log_g(h)$.

```
y ← ⌊√p⌋;
P ← g^y;
B ← g^{-1};
Initialize L as a Hash Table;
t ← 1;
L[t] ← 0;
for i = 1 to y do
   t ← t × P;
   L[t] ← i;
end for
t ← h;
j ← 0;
while t ∉ L do
   t ← t × B;
   j ← j + 1;
end while
return L[t]y + j
```

selected a_0, b_0, we define directly F as

$$u_{i+1} = F(u_i) = \begin{cases} g \times u_i \text{ if } u_i \in G_1 \\ u_i^2 \text{ if } u_i \in G_2 \\ h \times u_i \text{ if } u_i \in G_3 \end{cases} \quad \text{so that } (a_{i+1}, b_{i+1}) = \begin{cases} (a_i + 1, b_i) \text{ if } u_i \in G_1 \\ (2a_i, 2b_i) \text{ if } u_i \in G_2 \\ (a_i, b_i + 1) \text{ if } u_i \in G_3. \end{cases}$$

For simple experiments the behaviour of F is random enough. When efficiency is important, for example in record computations, it is better to partition the group in a larger number r of sets. The larger r is, the closer to a random behaviour F is. We do not discuss parallelization and more efficient variants of cycle detection. Details can be found in [CF05, Chap. 19] and [Jou09, Chap. 7.3].

Deducing $\log_g(h)$ from this random-ish walk is straightfroward. As mentioned previously, since the group is finite there exist $i < j$ such that $u_i = u_j$, or equivalently, such that $g^{a_i} h^{b_i} = g^{a_j} h^{b_j}$ which rewrites as $g^{a_i - a_j} = g^{x(b_j - b_i)}$ so that

$$x = \log_g(h) = \frac{a_i - a_j}{b_j - b_i} \mod p.$$

If $b_j = b_i$, we start the random walk again with other a_0, b_0. From this description, the algorithm still requires $O(\sqrt{p})$ in memory. This can be adressed in several ways using cycle-detections algorithms that requires constant memory space. A simple method is given by Floyd's algorithm, which can be summed up as follow: we look for a collision between "the regular walk" $(u_i)_{i \in \mathbb{N}}$ and "the speedy walk" $(u_{2i})_{i \in \mathbb{N}}$. Therefore, instead of storing every steps of the walk, we only keep track of the current step until $u_{2i} = u_i$. With good assumptions on the randomness of φ, this approach detects a cycle in $O(\sqrt{p})$ evaluation of φ, with $O(1)$ needed memory.

3.3 Index-Calculus in Jacobian Varieties

This Section is dedicated to a popular family of *specialized algorithms*, which use the additional structures surrounding the target group. The outline of the algorithm is simple: an overdetermined linear system between the discrete logarithms of selected elements of the group is built in a first phase called *harvesting*. It is solved in a second phase with *sparse linear algebra* algorithms, and the target discrete logarithm is computed from the solution. The harvesting is extremely context dependant and it is the main focus of this thesis. Section 3.3.1 dives into more details of Index Calculus and presents several variants, including the so-called *Large Primes* variants which aim at balancing the asymptotic complexity of the phases of the algorithm. The next Sections focus on Jacobian Varieties. Section 3.3.2 presents the subexponential variants achieved first by [ADH99] for hyperelliptic curves "of large genus", and of [EGT11] for "low degree" curves. The harvesting is done by looking for *smooth divisors* in the Jacobian Variety. This approach was presented for "small genus" hyperelliptic curves by Gaudry [Gau00], see Section 3.3.3. While the algorithm is exponential in the field's size, it is still asymptotically faster than generic algorithms as soon as $g \geq 4$, and as soon as $g \geq 3$ if large primes variants are used, see Table 3.2. The next Section deals with non-hyperelliptic curves and presents Diem's geometric approach [Die06]. Surprisingly, Diem's work revealed that such curves are weaker than hyperelliptic curves of the same genus. A complexity analysis is given and comparisons with other curves are shown in Table 3.3, which concludes this Section.

3.3.1 General algorithms

We assume for simplicity the target group is additive $(G,+)$, cyclic of prime order generated by g and that we are given a challenge $h = x \cdot g$. A first step is to select the *Factor Base*, which is a subset $\mathscr{B}$ of G. The terminology of a factor base can be found *e.g.* in the Number Field Sieve [LL93] in the context of integer factorization. The algorithm then mainly follows the following phases:

1. *Harvesting* consists in building a (large and sparse) determined linear system linking the discrete logarithms of elements of $\mathscr{B}$. Finding the linear equations, also called *relations*, is where additional structures surrounding the group are used. It is the central focus of this work.

2. *Linear Algebra* is the solving of the linear system. The wanted discrete logarithm is deduced from the solution, depending on the variant, see below for their description.

Let $\mathscr{B} = \{B_1, \ldots, B_m\}$, assume that $G = \langle P \rangle$ has order N and that $Q = x \cdot P$. Let also $k \geq N+1$ be an integer.

Variant 1:

Harvesting: We need k relations in the form of

$$a_i \cdot P + b_i \cdot Q = \sum_{j=1}^{m} c_{ij} B_j.$$

For all i, a_i, b_i and the c_{ij}'s are integers modulo N. Such relations give equations $a_i + b_i x = \sum_{i=1}^{m} c_{ij} \log_P(B_j)$ between the discrete logarithms of the elements in the factor base and P, Q. When enough relations have been collected, this gives the matrices $A = (a_i \ b_i)_{1 \leq i \leq k}$ and $M = (c_{ij})_{\substack{1 \leq i \leq k \\ 1 \leq j \leq m}}$.

Linear Algebra: A system $vM = 0$ is solved. If $k \geq N+1$ it usually yields a solution $v = (v_1, \ldots, v_k)$ in the left kernel of M. It must be checked that $vA \neq (0\ 0) \mod N$.

DL computation: For all $1 \leq i \leq k$ we have $a_i v_i + b_i v_i x = v_i \sum_{j=1}^{m} c_{ij} \log_P(B_j)$. Now we sum on i all these equations and use the fact that v is in the left kernel of M:

$$\begin{aligned}\sum_{i=1}^{k} a_i v_i + x\left(\sum_{i=1}^{k} b_i v_i\right) &= \sum_{i=1}^{k} v_i \left(\sum_{j=1}^{m} c_{ij} \log_P(B_j)\right) \\ &= \sum_{j=1}^{m} \log_P(B_j) \left(\sum_{i=1}^{k} v_i c_{ij}\right) \\ &= 0 \mod N.\end{aligned}$$

From this we recover $x = -\left(\sum_{i=1}^{k} a_i v_i\right)\left(\sum_{i=1}^{k} b_i v_i\right)^{-1} \mod N$. If $\sum_{i=1}^{k} b_i v_i$ is not invertible, other relations are found and another v is computed.

This variant only computes $x = \log_g(h)$. If other discrete logarithms in this group have to be computed, then the whole algorithm has to be started again.

Variant 2:

Harvesting: Relations are sought in the form of

$$\sum_{i=1}^{m} c_{ij} B_j = 0,$$

with all c_{ij} integer modulo N. Such relations rewrite as $\sum_{i=1}^{m} c_{ij} \log_P(B_j) = 0$. We let $M = (c_{ij})_{\substack{1 \leq i \leq k \\ 1 \leq j \leq m}}$.

Linear Algebra: A right kernel system $Mv = 0 \mod N$ is solved, and yields a solution $v = (v_1, \ldots, v_k)$ that describes the discrete logarithms of the B_i's up to a constant λ.

Descent Phase: For integers a, b modulo N, another two relations of the form

$$a_1 P + b_1 Q = \sum_{i=1}^{m} d_i B_i, \; a_2 P + b_2 Q = \sum_{i=1}^{m} e_i B_i$$

are needed. These relations give $a_1 + x b_1 = \lambda \sum_{i=1}^{m} d_i v_i$ and $a_2 + x b_2 = \lambda \sum_{i=1}^{m} e_i v_i$.

DL computation: Upon invertibility, the previous item leads to

$$x = \left(a_1 \sum_{i=1}^{m} e_i v_i - a_2 \sum_{i=1}^{m} d_i v_i\right)\left(b_2 \sum_{i=1}^{m} d_i v_i - b_1 \sum_{i=1}^{m} e_i v_i\right)^{-1} \mod N.$$

Else, another descent phase is done. A simpler form for x is obtained if the descent relations are sought with $b_1 = a_2 = 0$.

Since the discrete logarithms of all elements of $\mathscr{B}$ is obtained once the linear algebra is done, another DL computation in the same group needs only a new descent phase.

Comments on Linear Algebra: The matrix of relations is usually very sparse, and in our applications it is easy to bound the number of non-zero entries of a row in the matrix. In practical computations, a filtering step [Bou13, Tea15] is first performed to reduce the size of the matrix before a vector in the kernel is computed. For sparse matrices, computing kernels is usually done with Lanczos' or (Block-)Wiedemann's [Wie86] algorithms — short descriptions can be found in [CF05, Chap. 20.3, p.501] or [Jou09, Chap 3.4, p.105].

Large Primes Variants: In general not all elements of G can be written as a simple combination of elements of the factor base $\mathscr{B}$, so any time we want to find a relation, there is only a probability of success. The larger the factor base is, the higher the probability to find a relation is. However, the larger the factor base is, the harder the linear algebra is since the matrix' size depends on the cardinal of $\mathscr{B}$. Usually the linear algebra dominates both asymptotically and practically the run time for the computation. Hence improvements could come from *balancing* both phases, that is to say to reduce the size of the factor base without decreasing too much the probability of successfully finding a relation. Thériaut [Thé03] investigated an idea from Harley and gave an analysis of a balanced Index-Calculus algorithm with a "one large prime" harvesting. Then a common effort of Diem, Gaudry, Thériault and Thomé [GTTD07] gave a complete Index-Calculus using a "two large prime" harvesting. Here we only describe how the harvesting can be done. Complexity analysis are context dependant, and thus are postponed to Sections 3.3.3 and 3.3.4.

The factor base $\mathscr{B}$ is built, and then a subset $\mathscr{B}'$ of *small primes* is chosen — size is discussed in Sections 3.3.3 and 3.3.4. The elements of $\mathscr{B} \setminus \mathscr{B}'$ are called *large primes*. The word "prime" stems again from integer factorization and number field sieve. Then we search for three types of relations:

- *full relations*, which are sums $\sum c_{ij} B'_i$ involving small primes only;
- *one large prime relations*, which are sums $\sum c_{ij} B'_i + B$, where B is a large prime;
- *two large primes relations*: sums as $\sum c_{ij} B'_i + B_1 + B_2$, and B_1, B_2 are large primes.

The idea is that if enough relations like this are collected, then more than $\#\mathscr{B}'$ full relations can be obtained. The linear algebra is therefore done on a matrix with $\#\mathscr{B}'$ columns. Building full relations from large prime relations can be modelled by cycle-detection in a graph. The graph is built with the following steps:

1. It is initialized with a "root" $*$. Edges represent large primes relations: one large prime relation have vertices $*$ and the large prime, two large primes relations have the two large primes as vertices. Edges are labelled with the information of the relation.
2. If a full relation is found, it is directly added to the matrix.
3. If a large prime relation is found, we check if adding the corresponding edge to the graph creates a cycle.
   - If no cycle is created, the relation is added to the graph.
   - If a cycle is created, then there is a linear combination of the relations involved in the cycle that gives either a one large prime relation or a full relation. In the first case, the relation is added to the graph. In the second, the relation is added to the matrix.

Heuristically, $\#\mathscr{B} + \#\mathscr{B}'$ relations of the three types are needed in average to obtain enough full relations and to start the linear algebra.

It is not clear if such variants are indeed faster in practical discrete logarithm computations. In particular, an efficient elimination of large prime relations (*i.e.* management of the graph) must be implemented. To our knowledge, only one efficient implementation has been proposed and analyzed for non-hyperelliptic curves of genus 3 ([DT08, LL15]). In [LL15], the subset $\mathscr{B}'$ is built progressively while finding relations, in order to simplify the elimination of the large primes and improve linear algebra. A drawback of their method is the huge amount of needed memory.

3.3.2 Subexponential Approaches

Adleman-DeMarrais-Huang: smooth divisors

Let $\mathscr{H}$ be an hyperelliptic curve of genus g and defined over $\mathbb{F}_q$. For simplicity we assume that we are in the odd characteristic case, but everything can be easily adapted to the characteristic two case. The original algorithm of Adleman, DeMarrais and Huang uses the concept of *smooth divisors* to find relations between elements of the factor base. Their presentation and analysis show that the algorithm achieves a heuristic subexponential time when g is small compared to $\log q$. Further work of Enge, Gaudry and Stein [Eng02, EG02, ES02] rigorously obtained subexponentiality.

We give a short and informal description on Gaudry's formulation for this algorithm. The smoothness of a divisor is then defined using the Mumford representation.

Definition 3.7. *Let $\mathscr{H}$ be a genus g hyperelliptic curve defined over a field $\mathbb{F}$. Let $D \in \mathrm{Jac}(\mathscr{H})$ be the unique reduced divisor in the class $[D]$, and let (u,v) be its Mumford representation. Let $B \geq 1$ be an integer. We say that D is B-smooth if all the irreducible factors of u have degree at most B.*

Assume that $\mathrm{Jac}(\mathscr{H}) = \langle D \rangle$ and that we look for the discrete logarithm of $E \in \mathrm{Jac}(\mathscr{H})$. A smooth bound B is fixed, and the factor base is selected as

$$\mathscr{B} = \{D(u,v) \in \mathrm{Jac}(\mathscr{H}) \ : \ \deg u \leq B\}.$$

To find one relation, first the Mumford representation (u,v) of some $R = aD + bE$ is computed, then the factorization of u is obtained with standard algorithms. This factorization leads to a relation when all the irreducible factors have degree at most B.

While the method is relatively simple, the subexponentiality when g is small compared to $\log q$ was only heuristic in [ADH99].

Definition 3.8. *Let $0 \leq \alpha \leq 1$ and $N, c > 0$. The subexponential function is defined as*

$$L_N(\alpha, c) = \exp(c(\log N)^{\alpha}(\log\log N)^{1-\alpha}).$$

Usually, the constant c is omitted for the sake of clarity. Assume an algorithm runs in $O(L_N(\alpha))$. When $\alpha = 1$, the algorithm has exponential complexity in the size of N, subexponential for any $0 < \alpha < 1$ and polynomial if $\alpha = 0$. This function was defined to estimate time complexities of several functions in the Integer Factorization context, and in Index-Calculus algorithms for finite fields as well. It is also used to estimate the number of B-smooth elements in $\mathrm{Jac}(\mathscr{H})$.

Theorem 3.9 (Enge-Stein, [ES02])**.** *Let $\mathscr{H}$ be a hyperelliptic curve of genus g, defined over $\mathbb{F}_q$. Let $B = \lceil \log_q L_{q^g}(1/2, \rho) \rceil$ for some constant $\rho > 0$, and $N(B)$ the number of B-smooth elements in $\mathrm{Jac}(\mathscr{H})$. When g grows to infinity, we have*

$$N(B) \geq \frac{q^g}{L_{q^g}(1/2, 1/2\rho + o(1))}.$$

Hence an average of $O(L_{q^g}(1/2))$ tries are needed to find a relation. When the "genus is large", or in more precise terms, when g is large compared to $\log q$, or when q is fixed and g goes to infinity, the analysis in [EG02] shows that the optimal choice is $\rho = 1/\sqrt{2}$, hence $B = \lceil \log_q L_{q^g}(1/2, 1/\sqrt{2}) \rceil$. This gives the next theorem.

Theorem 3.10 (Enge-Gaudry, [EG02]). *Let $\mathscr{H}$ be a hyperelliptic curve of genus g defined over $\mathbb{F}_q$, and assume that $\log q = o(g)$. Then computing discrete logarithms in $\mathrm{Jac}(\mathscr{H})$ when g grows to infinity can be done with an Index-Calculus algorithm in*

$$O\left(L_{q^g}\left(\frac{1}{2},\sqrt{2}\right)\right).$$

Sketch of Proof. Fix $B = \lceil \log_q L_{q^g}(1/2, 1/\sqrt{2}) \rceil$. Since $\#\mathscr{B} = O(q^B)$, this means that around $L_{q^g}(1/2,1/\sqrt{2})$ relations have to be found to start linear algebra. As B-smooth elements in $\mathrm{Jac}(\mathscr{H})$ are found with probability $L_{q^g}(1/2,-1/2\sqrt{2})$, the harvesting runs in $L_{q^g}(1/2,1/\sqrt{2}) \times L_{q^g}(1/2,\sqrt{2}/2) = L_{q^g}(1/2,\sqrt{2})$. Linear algebra runs in $O(q^{2B}) = O(L_{q^g}(\frac{1}{2},\sqrt{2}))$. □

Enge-Gaudry-Thomé: low degree curves

A similar approach on a $\mathscr{C}_{ab}$ curves can be proposed. A $\mathscr{C}_{ab}$ curve is defined by an equation

$$\mathscr{C} : Y^a + X^b + f(X,Y) = 0,$$

with $\gcd(a,b) = 1$ and for all monomial X^iY^j in $\mathrm{Supp}(f)$, then $ai + bj < ab$. If we let $w(X) = a$ and $w(Y) = b$, it is the same as asking that $\deg_w(X^iY^j) < ab$. The genus is given by $g = \frac{(a-1)(b-1)}{2}$. They behave relatively similarly as hyperelliptic curves: the non-singular model of a $\mathscr{C}_{ab}$ curve has a single point at infinity and the elements of their Jacobian variety enjoy a nice description by polynomial ideals leading to a potentially efficient arithmetic. For this reason, they have been studied in [Ari03, Bas03], before Diem's work (see Section 3.3.4) revealed that they were in fact weaker than hyperelliptic curves of same genus.

To achieve a complexity of $L(1/3,\rho)$ for some constant ρ, the basic idea is to choose parameters a,b as well as the type of relations. The main part of Enge, Gaudry and Thomé's work [EGT11] is to analyze the constant ρ, and to give a heuristic proof that there are enough smooth divisors to start linear algebra. Here, we only present how the parameters can be chosen so that the complexity is $L_{q^g}(1/3)$ and refer to their article for additional details. First let $\mathscr{C}$ be a $\mathscr{C}_{ab}$ curve defined over $\mathbb{F}_q$, with $\deg_Y \mathscr{C} = a \approx g^\alpha$ and $\deg_X \mathscr{C} = b \approx g^{1-\alpha}$, for some $1/3 \leq \alpha \leq 2/3$. The smooth bound is set as $B = \lceil \log_q L_{q^g}(1/3) \rceil$, and q and the factor base is the set of all elements in $\mathrm{Jac}(\mathscr{H})$ that are B-smooth, so that $\#\mathscr{B} = O(L_{q^g}(1/3))$.

Relations are searched by intersections between $\mathscr{C}$ and polynomial functions $\varphi(X,Y) \in \mathbb{F}_q(\mathscr{C})$ with $\deg_X \varphi \approx g^{2/3-\alpha}$ and $\deg_X \varphi \approx g^{\alpha-1/3}$. This intersection can be described by $N(\varphi) = \mathrm{Res}_Y(\varphi, Y^a + X^b + f(X,Y))$. The notation comes from the fact that this is also the norm of φ relatively to the field extension $\mathbb{F}_q(\mathscr{C}) | \mathbb{F}_q(X)$. If $N(\varphi)$ has irreducible factors with degree at most B over $\mathbb{F}_q$, then it describes a relation. Because of the choice of the parameters, the degree of $N(\varphi)$ is

$$\begin{aligned}
\deg N(\varphi) &= \deg \mathrm{Res}_Y(\varphi, Y^a + X^b + f(X,Y)) \\
&= a \deg_X \varphi + b \deg_Y \varphi \\
&\approx g^\alpha g^{2/3-\alpha} + g^{1-\alpha} g^{\alpha-1/3} \\
&\approx 2g^{2/3} = O(g^{2/3}).
\end{aligned}$$

Assuming enough "genericity" in the behaviour of $N(\varphi)$ as a polynomial of degree $g^{2/3}$, the probability that it is B-smooth is $L_{q^g}(1/3)^{-1}$ — see the formulation of Theorem 3.9 in [ES02]. The harvesting then takes roughly $L_{q^g}(1/3) \times L_{q^g}(1/3) = L_{q^g}(1/3,\rho)$ operations, for some constant $\rho > 0$ and the linear algebra runs in $L_{q^g}(1/3) \times L_{q^g}(1/3) = L_{q^g}(1/3,2\rho)$.

3.3.3 Gaudry's approach for Hyperelliptic Curves of small genus

When the genus is "small" compared to the characteristic, or equivalently, when the genus is fixed and q grows to infinity, the optimal smooth bound becomes $B = 1$. If we write $D = (P_1) + \cdots + (P_{\deg u}) - \deg u(P_\infty)$, we see that D is smooth when the polynomial u splits over $\mathbb{F}_q$, that is to say that, for each i, the point P_i is $\mathbb{F}_q$-rational. The class of $(P_i) - (P_\infty)$ then defines an element of $\mathrm{Jac}_{\mathbb{F}_q}(\mathscr{H})$. Gaudry's algorithm [Gau00] stems from this observation, and its harvesting phase can be summarized as follows.

- The factor base $\mathscr{B}$ is the set $\{(P) - (P_\infty) : P \in \mathscr{H}(\mathbb{F}_q)\}$, or rather a set of representatives of its quotient by the hyperelliptic involution, accounting for the trivial relations $(\imath P) - (P_\infty) \sim -((P) - (P_\infty))$. It contains $O(q)$ elements.

- At each step, we compute (using a semi-random walk, see *e.g.* Section 3.2.3) the Mumford representation (u, v) of a reduced divisor $D \sim aD_0 + bD_1$, where D_0 and D_1 are the entries of the DLP challenge.

- If u splits over $\mathbb{F}_q$ as $\prod_i (x - x_i)$ then a relation is found since we have $aD_0 + bD_1 \sim \sum_i ((P_i) - (P_\infty))$ where $P_i = (x_i, v(x_i))$.

We see that each step of the harvesting phase requires a few operations in $\mathrm{Jac}_{\mathscr{H}}$ followed by the factorization of u, which is generically a degree g polynomial. The probability that u actually splits over $\mathbb{F}_q$ is about $1/g!$, so we need about $g!$ trials before finding a relation.

A precise analysis of the complexity is given by the author in [Gau00]. For the sake of completeness, we give the main steps and focus on time-complexity. The genus is fixed and the field size q grows to infinity. Enumerating the factor base $\mathscr{B}$ is done in $O(q)$, and in practice it is way faster than the next phases. We do not consider the problems of storing all the elements and of memory accesses.

Cost for one try: The Mumford Representation (u, v) of a divisor $aD + bE$ is computed. Following the pseudo-random walk of Section 3.2.3, this can be reduced to the cost of one addition or one doubling. We assume for simplicity that both share the same cost $c(g)$. Testing if u is split over $\mathbb{F}_q$ can be done in the first step of the root-finding algorithm: u is split over $\mathbb{F}_q$ if and only if $\mathrm{GCD}(X^q - X, u) = u$. The computation is done by fast exponentiation (see *e.g.* Algorithm 2) of X to the power q in the quotient ring $\mathbb{F}_q[X]/\langle u \rangle$ in $O(g^2 \log q)$ field operations. The total cost for a try is then $O(c(g) + g^2 \log q)$. Swan's criterion can also be used as a first splitting test: when a polynomial is split, then its discriminant is a square in the coefficients' field. Experimentally Gaudry [Gau00] observed nice speed-ups, but it does not help the analysis.

Cost for one relation: If u splits, then its roots can be computed in $\tilde{O}(g^2 \log q)$ by finishing the root-finding algorithm — the next step is the so-called Cantor-Zassenhaus' "Equal Degree Factorization" algorithm, see [CZ81, vzGG13]. The roots gives the abscissae for the P_i's, and the corresponding y-coordinates are computed by evaluation of a degree $g - 1$ polynomial in $g - 1$ multiplications. The probability that u splits is $1/g!$, so in average $g!$ tries are needed. Hence finding one relation costs roughly $O(g!(c(g) + g^2 \log q) + g(g \log q + 1))$.

Cost for the harvesting: At least $\#\mathscr{B} = O(q)$ relations are needed to build a determined matrix, so the harvesting is done in roughly $O(q(g!(c(g) + g^2 \log q) + g(g \log q + 1)))$. As g is fixed and q goes to infinity, the time complexity of the harvesting is $\tilde{O}(q)$.

Cost of the linear algebra: Computing the kernel of the matrix is done for example with Wiedemann's algorithm [Wie86] in $\tilde{O}(\#\mathscr{B}^2) = \tilde{O}(q^2)$ operations.

Asymptotically, the linear algebra phase dominates the complexity, but with a two large primes variant [GTTD07], the complexity can be balanced. Let $0 \leq \alpha < 1$ and select $\mathscr{B}' \subset \mathscr{B}$ of size $\#\mathscr{B}' \approx q^\alpha$. We now accept relations involving elements of $\mathscr{B}'$ and at most 2 elements of $\mathscr{B}$ (see Section 3.3.1). The probability that such a relation happens is heuristically

$$\frac{\#\mathscr{B}'^{g-2}}{(g-2)!} \cdot \frac{\#\mathscr{B}^2}{2\#\mathrm{Jac}(\mathscr{H})} \approx \frac{q^{\alpha(g-2)}}{2(g-2)!q^{g-2}} = \frac{q^{(\alpha-1)(g-2)}}{2(g-2)!}.$$

In general around $q + q^\alpha = O(q)$ relations are needed to have enough full relations, so the cost of this phase is $\tilde{O}(q^{(1-\alpha)(g-2)}q)$. Linear algebra is done in $\tilde{O}(q^{2\alpha})$, and as the goal is to balance the complexity, we want $1-(\alpha-1)(g-2) = 2\alpha$. This is achieved if $\alpha = 1 - \frac{1}{g}$, and leads to an overall complexity of $\tilde{O}(q^{2-2/g})$.

	$g=2$	$g=3$	$g=4$
Generic	$\tilde{O}(q)$	$\tilde{O}(q^{3/2})$	$\tilde{O}(q^2)$
Gaudry	$\tilde{O}(q^2)$	$\tilde{O}(q^2)$	$\tilde{O}(q^2)$
Gaudry+2LP	$\tilde{O}(q)$	$\tilde{O}(q^{4/3})$	$\tilde{O}(q^{3/2})$

Table 3.2: Asymptotic complexities for small genus hyperelliptic curves

Recall that $\#\mathrm{Jac}(\mathscr{H}) \approx q^g$, so that generic algorithms run in $\tilde{O}(q^{g/2})$. From Table 3.2, Gaudry's approach is better than generic algorithms as soon as $g > 4$, and with a two large prime variant, as soon as $g \geq 3$.

3.3.4 Diem's approach for Small Degree Plane Curves

Gaudry's algorithm can be adapted to non-hyperelliptic curves. Most divisors can still be represented by Mumford coordinates, but computations in the Jacobian variety are not as tractable and checking for 1-smoothness is less obvious. However, all these operations are in $\tilde{O}(1)$ when g is fixed, so that the asymptotic complexity is still in $\tilde{O}(q^{2-2/g})$, albeit with a larger hidden constant than in the hyperelliptic case.

In [Die06], Diem designed a different harvesting technique for plane curves whose degree d is really close to the genus. Harvesting is done by considering relations coming from principal divisors corresponding to equations of lines. For any couple of rational points P_1 and P_2 on the (affine) curve $\mathscr{C}$, we consider the affine function $f = ax + by + c \in \mathbb{F}_q(\mathscr{C})$ such that the equation of the line L passing through P_1 and P_2 is $f = 0$. Since the curve has degree d, the intersection of L and $\mathscr{C}$ contains up to d rational points. Two of them are already known, and to determine the other $d-2$ points amounts to finding the roots of a degree $d-2$ univariate polynomial. If there are exactly $d-2$ other rational intersection points $P_3, \ldots, P_d$ then we obtain a relation of the form

$$\mathrm{div}(f) = (P_1) + \cdots + (P_d) - D_\infty \sim 0,$$

where D_∞ is the divisor corresponding to the intersection of $\mathscr{C}$ with the line at infinity. This happens with probability $1/(d-2)!$, which is better than the $1/g!$ probability for hyperelliptic curves as soon as $d \leq g+1$.

We can summarize Diem's harvesting technique as follows. The curve $\mathscr{C}$ is defined by an affine plane equation $F(x,y) = 0$. We no longer consider only (classes of) degree 0 divisors, so technically we are working in the full divisor class group and not only its degree 0 part, but in practice it makes no difference.

- The factor base is $\mathscr{B} = \{(P) : P \in \mathscr{C}(\mathbb{F}_q)\} \cup \{D_\infty\}$.
- We choose two points $P_1 = (x_1, y_1)$ and $P_2 = (x_2, y_2)$ in $\mathscr{B}$ such that $x_1 \neq x_2$ (for simplicity) and compute $\lambda = (y_2 - y_1)/(x_2 - x_1)$ and $\mu = y_1 - \lambda x_1$.
- We test if the degree $d-2$ polynomial $\frac{F(x,\lambda x+\mu)}{(x-x_1)(x-x_2)}$ splits over $\mathbb{F}_q$. If it is the case, we compute its roots $x_3, \ldots, x_d$ and the associated y-coordinates $y_3 = \lambda x_3 + \mu, \ldots$, and we store the relation
$$(P_1) + (P_2) + (P_3) + \cdots + (P_d) - D_\infty \sim 0$$
where $P_i = (x_i, y_i)$, provided these points are non-singular.
- We go back to the second step until enough relations are found.

Note that a descent phase is needed to express the entries of the DLP challenge in terms of elements of the factor base.

The cost to find one relation now depends on the degree d of the curve, hence this approach should be efficient for curves of small degree, in particular for curve with degree $d \leq g+1$. The main result of [Die06] states that, if a curve of genus $g \geq 3$ is general enough (which rules out hyperelliptic curves), a plane model of expected degree $d \leq g+1$ can be found for the curve in polynomial time using a probabilistic algorithm. This means that Diem's algorithm applies to almost all non-hyperelliptic curves, with the (then unexpected) consequence that the DLP is easier on non-hyperelliptic curves than on hyperelliptic ones.

The whole routine is particularly well-suited for two large prime variation (for another version differing mainly on the construction of the factor base and the large prime graph, see also [LL15]). Instead of selecting two points in $\mathscr{B}$, we pick them in the small factor base $\mathscr{B}'$ and keep only relations involving at most two large primes. The main advantage (as compared to the hyperelliptic case) is that we ensure in this way that each potential relation contains already two small primes; this greatly increases the probability of finding relations with only two large primes. In particular if $d = 4$, every relation found by the above method automatically satisfies the two large prime condition.

We now follow the analysis[2] of Section 3.3.3, and use a two large prime approach. The degree d is fixed and q goes to infinity. First, a symbolic polynomial $f_{\lambda,\mu,x_1,x_2} = \frac{F(x,\lambda x+\mu)}{(x-x_1)(x-x_2)}$ of degree $d-2$ can be precomputed. Enumerating the factor base $\mathscr{B}$ is done in $O(q)$, and in practice it is way faster than the next phases. Again we do not consider storing and memory accesses problems. Let $0 \leq \alpha < 1$ and select the small prime set $\mathscr{B}' \subset \mathscr{B}$ of size $\approx q^\alpha$.

Cost for one try: Computing λ and μ is done in 1 multiplication and 1 inversion. Testing if f_{λ,μ,x_1,x_2} splits over $\mathbb{F}_q$ can be done in at worst $(d-2)^2 \log q$ field operations as in Section 3.3.3. Swan's criterion can also be used as a first splitting test. The total cost for a try is then essentially $(d-2)^2 \log q$.

Cost for one relation: If f_{λ,μ,x_1,x_2} splits, its roots are found in $\tilde{O}((d-2)^2 \log q)$. The roots gives abscissae for the P_i's, and the corresponding y-coordinates are computed with the line's equation in 1 multiplication. The probability that the rational intersection is complete is heuristically
$$\frac{\#\mathscr{B}'^{d-4}}{(d-4)!\#\mathscr{B}^{d-2}} \cdot \frac{\#\mathscr{B}^2}{2} \approx \frac{q^{(\alpha-1)(d-4)}}{2(d-4)!}$$
Usually $O(q)$ large primes relations are enough to start linear algebra on $\mathscr{B}'$, so the cost of this phase is $\tilde{O}(q^{(1-\alpha)(d-4)}q)$.

[2] A more precise analysis is given in [Die06].

Linear algebra: It is done in $\tilde{O}(q^{2\alpha})$ operations. To balance the complexity, we want $1-(\alpha-1)(d-4)=2\alpha$, achieved by $\alpha=1-\frac{1}{d-2}$. The final complexity is then $\tilde{O}(q^{2-2/d-2})$.

Since $\mathscr{C}$ admits a plane model of degree $g+1$, the asymptotic run time is $\tilde{O}(q^{2-2/(g-1)})$, which improves over the $\tilde{O}(q^{2-2/g})$ complexity of the hyperelliptic case. Note however that the size of the small factor base is such that in order to find enough relations, almost all lines going through pairs of points of $\mathscr{B}'$ have to be considered. This is troublesome because each line, and thus each relation, can be obtained several times, namely $n(n-1)/2$ times if it contains n small factor base points. This is not really an issue if $d=4$, but for higher d some extra care has to be taken in order to prevent duplicate relations.

	$g=3$	$g=4$	$g=5$
Generic	$\tilde{O}(q^{3/2})$	$\tilde{O}(q^{2})$	$\tilde{O}(q^{5/2})$
Gaudry+2LP	$\tilde{O}(q^{4/3})$	$\tilde{O}(q^{3/2})$	$\tilde{O}(q^{8/5})$
Diem+2LP	$\tilde{\mathbf{O}}(\mathbf{q})$	$\tilde{O}(q^{4/3})$	$\tilde{O}(q^{3/2})$

Table 3.3: Asymptotic complexities for small genus non-hyperelliptic curves

Table 3.3 shows that Index-Calculus on a non-hyperelliptic curve of genus g is as hard as Index-Calculus on a hyperelliptic curve of genus $g-1$, and that it also outperforms generic algorithms as soon as $g\geq 3$. A notable fact is that, for approximately 18.5% of genus 3 hyperelliptic curves [Smi08], a DLP instance can be mapped to an equivalent instance on a non-hyperelliptic curve of same genus, where Diem's algorithm is particularly efficient.

3.4 Attacks based on Weil Restrictions

3.4.1 Transfer attacks

Using the Weil Restriction of an Abelian Variety Let $\mathscr{A}$ be an Abelian Variety (Definition 2.15) defined over $\mathbb{F}_{q^n}$, for some $n\geq 2$, and let $W=\mathscr{W}_n(\mathscr{A})$ be its Weil Restriction (as defined in Section 1.5.2). The natural identification between $\mathscr{A}(\mathbb{F}_{q^n})$ and the $\mathbb{F}_q$-rational points of W allows to transfer the group law from $\mathscr{A}$ to W. Being an Abelian Variety, the group law on $\mathscr{A}$ can be expressed by rational functions with coefficients in $\mathbb{F}_{q^n}$. Hence the group law on W can be expressed by rational functions, with coefficients in $\mathbb{F}_q$. This means that W naturally inherits an Abelian Variety structure.

Transfer attacks rely on this property to map a DLP instance on $\mathscr{A}$ to a "convenient" (to be discussed) subvariety of W. Assume that we are given a curve $\mathscr{C}$ defined over $\mathbb{F}_q$ in W, or in other words, a regular map $\psi:\mathscr{C}\longrightarrow W$. Further assume there is at least one $\mathbb{F}_q$-rational point $P\in\mathscr{C}$. Up to translation, we can assume $\psi(P)$ is the neutral element in W. Recall that $\mathrm{Jac}_{\mathbb{F}_q}(\mathscr{C})$ identifies to the g^{th} symmetric product $\mathscr{C}^g/\mathfrak{S}_g$ (Section 2.1.3). Abusing notations, we let $\psi:\mathrm{Jac}(\mathscr{C})\longrightarrow W$ be defined by $\psi(P_1,\ldots,P_g)=\sum_{i=1}^{g}\psi(P_i)$ using the law of W. This map sends the neutral element in the Jacobian Variety to the neutral of W: a result on Abelian Varieties (see for example [Mil86b]) then ensures that ψ is also a group homomorphism. An instance of DLP in $\mathscr{A}$ can then be pulled back to $\mathrm{Jac}_{\mathbb{F}_q}(\mathscr{C})$. As practical applications mainly deal with Elliptic Curves, we now focus on this setting.

Coverings of Elliptic Curves and the GHS technique We now assume that $\mathscr{A}$ has dimension 1, or equivalently that it is an Elliptic Curve E (defined over $\mathbb{F}_{q^n}$). Finding $\mathscr{C}|_{\mathbb{F}_q}$ amounts to finding a *covering* $\pi:\mathscr{C}\longrightarrow E$, where π is a $\mathbb{F}_{q^n}$-morphism of curves. A group homomorphism $\tau:\mathrm{Jac}_{\mathbb{F}_{q^n}}(E)=E\longrightarrow\mathrm{Jac}_{\mathbb{F}_q}(\mathscr{C})$ can then be obtained (called *conorm-norm map* in [GHS02]), and a DLP on E can be transferred to a DLP in $\mathrm{Jac}_{\mathbb{F}_q}(\mathscr{C})$, provided that the kernel of τ is not too big. Several difficulties now have to be considered. For the point of

view of our work, the main concern is the construction of the curve $\mathscr{C}$. For this transfer to be interesting in term of practical attacks, we want that the genus of $\mathscr{C}$ is not too high. However, there are no known general method to obtain a curve of prescribed genus in a variety. Since $\dim \mathscr{W}_n(E) = n$, a curve can be obtained with $n-1$ "generic" hyperplane sections, and the remaining difficulty is to estimate its genus $g(\mathscr{C})$. A first rough estimation can be done, with the assumption that the kernel of the conorm-norm map is trivial: as $\#E(\mathbb{F}_{q^n}) \approx q^n$, we must have $q^n \leq q^{g(\mathscr{C})}$, so that $n \leq g(\mathscr{C})$.

A general analysis can be done using function field theory, thanks to the work of Gaudry, Hess and Smart in even characteristic [GHS02], and Diem's in odd characteristic [Die03]. We give a brief and informal description. The function field of E is a quadratic extension of $\mathbb{F}_q(x)$. If σ is any element in $\mathrm{Gal}(\mathbb{F}_{q^n} | \mathbb{F}_q)$, so are the function fields of E^{σ^i}. The idea is now to build a common extension $\mathbb{F}$ for all those function fields and to study their embeddings in $\mathbb{F}$. As all $\mathbb{F}_{q^n}(E^\sigma)$ are quadratic, it is necessary that $[\mathbb{F} : \mathbb{F}_{q^n}(x)] = 2^m$, for some integer m. Its value highly depends on the choice of a model for E, equivalently on the choice of an extension $\mathbb{F}_{q^n}(x) \longrightarrow \mathbb{F}_{q^n}(E)$. The curve $\mathscr{C}$ is then a curve such that $\mathbb{F}_{q^n}(\mathscr{C}) = \mathbb{F}$, and its genus depends exponentially in m.

Transfer attacks on "small" field extensions Because of this last observation, few values of n can lead to interesting transfers. For example, for an Elliptic Curve E defined over $\mathbb{F}_{q^6}$, q odd, hyperelliptic coverings of genus 2 and 3 can be obtained defined respectively over $\mathbb{F}_{q^3}$ and $\mathbb{F}_{q^2}$ [JV12]. For Hyperelliptic Jacobian Varieties, the attacks previously presented may perform better than a direct attack on E. If a suitable covering of E can be found, an efficient attack on E can then be designed by first tranferring the DLP to the Jacobian Variety of the cover, then solving this new DLP instance, provided an efficient algorithm exists for this class of Jacobian Variety. The original DLP is then mapped back to E. When the degree of the extension considered is small and the target curve has small genus, *Decomposition attacks* can be such efficient attacks. They are the main concern of this thesis, and therefore deserve the next section.

3.4.2 Decomposition attacks

In this Section we focus on Jacobian Varieties $\mathrm{Jac}(\mathscr{H})$ of hyperelliptic curves defined over field extensions. Here we consider elliptic curves as hyperelliptic curves of genus 1. Relations harvesting in this setting can be done by solving multiple instances of the following problem.

Definition 3.11 (Point m-Decomposition Problem (PDP_m))**.** *Given an element R and a subset $\mathscr{B}$ of $\mathrm{Jac}(\mathscr{H})$, find, if they exist, $D_1, \ldots, D_m$ with $D_i \in \mathscr{B}$ such that:*

$$R = D_1 + \cdots + D_m.$$

When the curve is an elliptic curve E, i.e. $g = 1$, an algebraic modelling of PDP_m instance, involving Summation Polynomials [Sem04], was introduced by Diem [Die11] and Gaudry [Gau09]. Usually, the factor base is selected as $\mathscr{B} = \{P \in E(\mathbb{F}_{q^n}) \ : \ x(P) \in \mathbb{F}_q\}$. Their similar approaches rely on the linear structure of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$ to describe a decomposition by a 0-dimensional system — see also Section 1.5.2. Such systems are solved with Gröbner Bases strategy, see Section 1.3.3. Description of Summation Polynomials and their impact for PDP_m solving is analyzed in Section 3.4.3, and the situation is illustrated by a toy example. More general details on the usage of Elliptic Summation Polynomials are given in the survey of [GG16].

When $g \geq 2$, Nagao proposed in [Nag10] to solve instances of this problem by using a geometric description of decompositions involving bases of Riemann-Roch spaces, together with a Weil Descent strategy. When $\mathscr{H}$ has genus g and is defined over $\mathbb{F}_{q^n}$, he also selected

the factor base as $\mathscr{B} = \{(P) : P \in \mathscr{H},\ x(P) \in \mathbb{F}_q\}$. Systems arising from this method are also generally 0-dimensional and solved with Gröbner basis methods. We introduce this approach in Section 3.4.4.

3.4.3 Semaev's summation polynomials and Weil Descent

Semaev introduced in [Sem04] a multivariate polynomial that describes vanishing sums of points of an elliptic curve. We omit details for characteristic 3 fields, as we never explicitely consider this situation in our contributions. Anyway, the following presentation can be extended to this characteristic.

Definition-Proposition 3.12 (Summation Polynomials for Elliptic Curve)**.** *Let E be an elliptic curve defined over a field $\mathbb{F}$ with algebraic closure $\overline{\mathbb{F}}$, and $\mathscr{O}$ its point at infinity. For $m \geq 3$, the m-th summation polynomial associated to E is a polynomial $S_m \in \mathbb{F}[X_1, \ldots, X_m]$ defined by the Summation Property:*

$$S_m(x_1, \ldots, x_n) = 0 \ \Leftrightarrow\ \exists\ y_1, \ldots, y_m \in \overline{\mathbb{F}} \text{ such that } P_i(x_i, y_i) \in E(\overline{\mathbb{F}}), 1 \leqslant i \leqslant m, \text{ and } P_1 + \cdots + P_m = \mathscr{O}.$$

If $Char(\mathbb{F}) \neq 2, 3$, and E is given by a Weierstrass equation $y^2 = x^3 + ax + b$, we have:

$$S_3(X_1, X_2, X_3) = (X_1 - X_2)^2 X_3^2 - 2((X_1 + X_2)(X_1X_2 + a) + 2b)X_3 + (X_1X_2 - a)^2 - 4b(X_1 + X_2),$$

and if $Char(\mathbb{F}) = 2$ with $E : y^2 + xy = x^3 + ax^2 + b$ we have:

$$S_3(X_1, X_2, X_3) = (X_1X_2 + X_1X_3 + X_2X_3)^2 + X_1X_2X_3 + b.$$

For $n \geq 3$, S_m is symmetric in $X_1, \ldots, X_m$, is irreducible, and has degree 2^{m-2} in each variable.

Since the addition in an elliptic curve is commutative, the order in which the points are considered does not matter. This gives an informal intuition as to why S_m is a symmetric polynomial. The idea behind Summation Polynomials is to project the elliptic curve's group law on the x-line. The formula for S_3 can be derived by hand [Sem04]. For larger sum size, Semaev also gave a recursive way to compute the polynomials:

Proposition 3.13 (Recursive expression of summation polynomials)**.** *Let E be an elliptic curve, $m \geq 4$ be an integer and $Res_T(P, Q)$ be the resultant of two polynomials with respect to T. For all $2 \leqslant k \leqslant m-2$, the m^{th} Summation Polynomial associated with E can be computed as:*

$$S_m(X_1, \ldots, X_m) = Res_T(S_{k+1}(X_1, \ldots, X_k, T), S_{m-k+1}(X_{k+1}, \ldots, X_m, T)).$$

Sketch of proof. Checking that S_3 vanishes exactly on sum of size 3 can be done by tedious but direct computations, see [Sem04]. A geometric proof can be derived from our new modelling of Summation Polynomials (Propositions 5.3 and 5.6). The recursive formula comes from the rewriting of a sum of size m as two smaller sums and induction. Fix $m \geq 4$ and $2 \leq k \leq m-2$. For $P_i \in E$ let $x_i = x(P_i)$. Then we have:

$$\begin{aligned} P_1 + \cdots + P_m = \mathscr{O} \Leftrightarrow \exists\ Q \in E\ :\ & \begin{cases} P_1 + \cdots + P_k = Q \\ P_{k+1} + \cdots + P_m = -Q \end{cases} \\ \Leftrightarrow \exists\ Q \in E\ :\ & \begin{cases} S_{k+1}(x_1, \ldots, x_k, x(Q)) = 0 \\ S_{m-k+1}(x_{k+1}, \ldots, x_m, x(Q)) = 0 \end{cases} \end{aligned}$$

The last statement is true if and only if $S_{k+1}(x_1, \ldots, x_k, T)$ and $S_{m-k+1}(x_{k+1}, \ldots, x_m, T)$ have a common root. This is equivalent to asking that their resultant with respect to T vanishes. □

Using only Proposition 3.13, computations revealed to be already hard for $n > 5$. More efficient ways to compute summation polynomials are given in [JV13] for $n < 7$, relying on the symmetry of the polynomial. With additional symmetries, it culminated in [FHJ+14] where authors computed up to the 8-th summation polynomial, with a dedicated method for this computational challenge. The method takes advantage of the fact that the representation of a summation polynomial using a set of fundamental invariants is usually much sparser.

Summation Polynomials and PDP$_n$ instances Diem and Gaudry showed independently [Die11, Gau09] how to use the summation polynomials in an Index-Calculus context. Let E be an elliptic curve defined over $\mathbb{F}_{q^n}$, and select the factor base as $\mathscr{B} = \{P \in E(\mathbb{F}_{q^n}) : x(P) \in \mathbb{F}_q\}$. Assume we are given $R \in E(\mathbb{F}_{q^n})$. In practice, such R is typically obtained by a linear combination $R = aP + bQ$ of the generator P and the challenge Q. We look for decompositions of R as a sum of exactly n points:

$$R = P_1 + \cdots + P_n, \ P_i \in \mathscr{B}.$$

Let $x_i = x(P_i)$ be the abscissae of P_i. This PDP$_n$ instance translates algebraically with Definition 3.12 as

$$\sum_{i=1}^{n} P_i = R \Leftrightarrow S_{n+1}(x_1, \ldots, x_n, x(R)) = 0, \tag{3.2}$$

with $x_i \in \mathbb{F}_q$. Let $1, t, \ldots, t^{n-1}$ be a $\mathbb{F}_q$-basis of $\mathbb{F}_{q^n}$ to write $x(R) = \sum_{i=0}^{n-1} r_i t^i$ where $r_i \in \mathbb{F}_q$, and $X_i = \sum_{i=0}^{n-1} X_{ij} t^j$. Collecting $S_{n+1}(X_1, \ldots, X_n, x(R))$ with respect to t, we obtain an expression

$$S_{n+1}(X_1, \ldots, X_n, x(R)) = \sum_{i=0}^{n-1} s_i(X_{1,0}, \ldots, X_{1,n-1}, \ldots, X_{n,0}, \ldots, X_{n,n-1}) t^i,$$

where $s_i \in \mathbb{F}_q[X_{1,0}, \ldots, X_{n,n-1}]$ for all i. Asking that $x_i \in \mathbb{F}_q$ amounts to asking that $X_{ij} = 0$ for all i and all $j > 0$. Hence Equation (3.2) can be rewritten as a multivariate polynomial system

$$\mathscr{S} = \{s_i(X_1, \ldots, X_n) = 0, 1 \leq i \leq n\}$$

defined over $\mathbb{F}_q$. A solution of this system corresponds to a solution of the PDP$_n$ instance, provided the corresponding y_i are in $\mathbb{F}_q$ and not in some $\mathbb{F}_{q^2}$. It never happened in our experiments, and if it happens, then it can be shown that this gives a shorter decomposition of a 2-torsion point (see [Vit11, Chap. 7,p. 130]). Heuristically[3] the probability for such a relation to happen is:

$$\frac{\#\left(\mathscr{B}^n / \mathfrak{S}_n\right)}{\#E^n} \approx \frac{q^n}{n!} \frac{1}{q^n} = \frac{1}{n!},$$

where $\mathfrak{S}_n$ denotes the symmetric group of order n. In general, the ideal generated by $\mathscr{S}$ has dimension 0 and has degree $n! 2^{n(n-1)}$, which makes computations untractable when $n > 4$. This degree can be reduced by considering actions of symmetries.

Recall that S_n is symmetric and can thus be expressed with the elementary symmetric functions in $X_1, \ldots, X_n$ before solving the system. This reduces the number of solutions by a factor $n!$. Moreover, if the curve has rational torsion points of small order, a larger group of symmetry can be considered. The basic idea is the following: let $T \in E(\mathbb{F})[k]$. If we fix a size n for sums of points, then adding a multiple of k times T to this sum does not change the

[3] It can be rigourously proven [Die11] using intersection theory that this estimation holds.

result. For example if $k = 2$:

$$\begin{aligned} R &= P_1 + \cdots + P_n \\ &= (P_1 + T) + (P_2 + T) + P_3 + \cdots + P_n \\ &= (P_1 + T) + P_2 + (P_3 + T) + \cdots + P_n \\ &= \dots \\ &= (P_1 + T) + (P_2 + T) + (P_3 + T) + (P_4 + T) + P_5 + \cdots + P_n \\ &= \dots \end{aligned}$$

Therefore S_n must be invariant under the action of the two-torsion group and the permutation of variables. The underlying group is known as a Coxeter group, and has order $n! \times 2^{n-1}$. The first results in this direction were given for binary (twisted) Edwards and Hessian curves in [FGHR14], where authors gained an additional factor of 2^{n-1}. These results were generalized in [FHJ+14] to any characteristic and any elliptic curve. Taking into account a possible 4-torsion point, the number of solutions can be further reduced by a factor $2^{2(n-1)}$. A similar approach to the Weil descent for binary Edwards curves over prime extensions was used in [GG14].

We now assume that $S_{n+1}(X_1, \dots, X_n, x(R))$ is expressed with the elementary symmetric functions in $X_1, \dots, X_n$ as a polynomial $\tilde{S}_{n+1}$, or in other words, that we take in account the action of $\mathfrak{S}_n$ over $S_{n+1}(X_1, \dots, X_n, x(R))$. It can be shown [Die11] that $\deg \tilde{S}_{n+1} = 2^{n-1}$ in general. Since we are considering a subvariety of dimension 0 in the Weil Restriction of $\mathbf{V}(\tilde{S}_{n+1})$, the description in Section 1.5.2 shows that, in general, a PDP$_n$ system has $\deg \mathscr{S} = (\deg \tilde{S}_{n+1})^n = 2^{n(n-1)}$ solutions. We denote by $c(n)$ the cost of solving such a sytem. Following Section 1.3.3, the solving strategy starts by computing a DRL order basis for $\mathscr{S}$, whose complexity depends on the degree of regularity $\delta(n) = d_{a,reg}(\mathscr{S})$ of the ideal generated by $\mathscr{S}$. To simplify the analysis, we assume that $(s_0, \dots, s_{n-1})$ is an affine regular sequence — looking at Proposition 1.49, this assumption seems reasonable as $\mathscr{S}$ is usually 0-dimensional. Then, an upper bound on $\delta(n)$ is given by the Macaulay bound $\Delta = \sum_{i=1}^{n} (\deg s_i - 1) + 1$ (see also Proposition 1.50). Since $\deg s_i \leq 2^{n-1}$ for each i, then $\Delta \sim n2^{n-1}$. The following estimates hold:

$$\begin{aligned} \binom{n + \delta(n)}{\delta(n)} &\leq \frac{(\Delta + 1) \dots (\Delta + n)}{n!} \\ &\sim \frac{n^n 2^{n(n-1)}}{n!} \end{aligned}$$

With Stirling's formula, we obtain:

$$\binom{n + \delta(n)}{\delta(n)} \sim 2^{n(n-1)} e^n \sqrt{2\pi n}.$$

With Theorem 1.53, computing a Gröbner basis for the DRL order costs $O\left((2^{n(n-1)} e^n \sqrt{2\pi n})^\omega\right)$. Proposition 1.58 shows that computing a lexicographical basis can be done in $O(n 2^{\omega n(n-1)})$. Lastly, the lexicographical basis is in Shape Position with very high probability, so that solutions are found once the roots of a univariate polynomial of degree $\deg \mathscr{S} = 2^{n(n-1)}$ are found. It can be done in $O(2^{2n(n-1)} \log q)$ (see *e.g.* [vzGG13]). Overall we find

$$c(n) = \tilde{O}\left(2^{\omega n(n-1)} (e^{\omega n} \sqrt{2\pi n}^\omega + n) + 2^{2n(n-1)}\right).$$

Remark 3.14. *While asymptotically the DRL computation dominates the complexity, the change-ordering dominates the total run time in almost all experiments.*

The factor base has around q elements, so the harvesting runs in $O(n!c(n)q)$ and linear algebra in $O(q^2)$. A two large prime variant can be used [Gau09], and an analysis similar to that of Sections 3.3.3 and 3.3.4 shows that the set of small primes should have size $O(q^{1-1/n})$, leading to a linear algebra in $O(q^{2-2/n})$. As generic algorithms run in $O(q^{n/2})$ over an elliptic curve defined over $\mathbb{F}_{q^n}$, this method performs asymptotically better as soon as $n > 2$. However, the cost of finding one relation is in average $n!c(n)$, which is exponential in n.

Example: Let $\mathbb{F} = \mathbb{F}_{1031^2} = \mathbb{F}_{1031}[t]/\langle t^2 - 2t + 14\rangle$ and $E : y^2 = x^3 + (98t + 202)x + (769t + 711)$. We want to solve PDP_2 instances, so we need the 3rd Summation Polynomial, which is given directly in Definition 3.12. Let $R(211t + 341, 539t + 528) \in E$. We try to write it as

$$R = P_1 + P_2,\ P_i \in \mathscr{B} = \{P \in E : x(P) \in \mathbb{F}_{1031}\}.$$

Evaluating S_3 at $x(R) = 211t + 341$, we find the bivariate polynomial

$$\begin{aligned} S_3(X_1, X_2, x(R)) = {} & X_1^2X_2^2 + (609t + 349)X_1^2X_2 + (969t + 239)X_1^2 + (609t + 349)X_1X_2^2 + (959t + 149)X_1X_2 \\ & + (293t + 201)X_1 + (969t + 239)X_2^2 + (293t + 201)X_2 + 590t + 837. \end{aligned}$$

Collecting wrt. t, we obtain the following system of 2 equations in 2 variables over $\mathbb{F}_{1031}$:

$$\begin{cases} 609X_1^2X_2 + 969X_1^2 + 609X_1X_2^2 + 959X_1X_2 + 293X_1 + 969X_2^2 + 293X_2 + 590, \\ X_1^2X_2^2 + 349X_1^2X_2 + 239X_1^2 + 349X_1X_2^2 + 149X_1X_2 + 201X_1 + 239X_2^2 + 201X_2 + 837 \end{cases}$$

A Gröbner basis computation for a lex order with $X_1 > X_2$ gives

$$\begin{cases} X_1 + T_2(X_2) = X_1 + 304X_2^7 + 607X_2^6 + 300X_2^5 + 459X_2^4 + 865X_2^3 + 956X_2^2 + 119X_2 + 280, \\ T_1(X_2) = X_2^8 + 506X_2^7 + 572X_2^6 + 797X_2^5 + 706X_2^4 + 1015X_2^3 + 781X_2^2 + 782X_2 + 117 \end{cases}$$

so we check that the degree is $n! \times 2^{n(n-1)} = 2! \times 2^{2\cdot 1} = 8$. The polynomial T_1 has roots 130 and 585 in $\mathbb{F}_{1031}$, which leads to 585 and 130 as values for X_2. Indeed, the points $P_1(130, 154t + 161)$ and $P_2(585, 910t + 635)$ of E are in $\mathscr{B}$ and such that $R = P_1 + P_2$.

It is clear that $(130, 585)$ and $(585, 130)$ are in fact the same solutions. The action of the symmetric group can indeed be taken into account. As Summation Polynomials are symmetric, so is $S_3(X_1, X_2, x(R))$. If e_1, e_2 are variables for the elementary symmetric functions in X_1, X_2, the symmetrized expression is

$$\widetilde{S_3}(e_1, e_2) = (969t + 239)e_1^2 + (609t + 349)e_1e_2 + (293t + 201)e_1 + e_2^2 + (52t + 702)e_2 + 590t + 837.$$

This polynomial is sparser than $S_3(X_1, X_2, x(R))$ and has lower total degree, so we expect the ideal resulting of the Weil Descent to have reduced degree as well. Collecting wrt. t, a new system is obtained:

$$\begin{cases} 969e_1^2 + 609e_1e_2 + 293e_1 + 52e_2 + 590, \\ 239e_1^2 + 349e_1e_2 + 201e_1 + e_2^2 + 702e_2 + 837 \end{cases}$$

For lex order $e_1 > e_2$, it admits the Gröbner basis:

$$\begin{cases} e_1 + 468e_2^3 + 44e_2^2 + 735e_2 + 734, \\ e_2^4 + 767e_2^3 + 462e_2^2 + 206e_2 + 117 \end{cases}$$

The degree of the ideal is $2^{n(n-1)} = 4$ as the action of the symmetric group is now encoded. Solutions over $\mathbb{F}_{1031}$ are $(715, 787)$ and $(1002, 437)$, and an additional step is to factor over $\mathbb{F}_{1031}$ the two univariate polynomials

$$\begin{aligned} F_1(x) = {} & x^2 - 715x + 787, \\ F_2(x) = {} & x^2 - 1002x + 437. \end{aligned}$$

The polynomial F_1 has familiar roots 130 and 585, for which we recover the same decomposition of R, while F_2 has none over $\mathbb{F}_{1031}$.

3.4.4 Nagao's approach using Riemann-Roch coordinates

Let $\mathscr{H}$ be an imaginary hyperelliptic curve of genus g, defined over a field $\mathbb{F}_{q^n}$, $n \geq 2$, by a Weierstrass equation $y^2 + h_1(x)y = h_0(x)$, $\deg h_1 \leqslant g, \deg h_0 = 2g+1$ and a single point at infinity P_∞. Recall that elements of the Jacobian Variety $\mathrm{Jac}(\mathscr{H})$ are called *reduced divisors*. The zero class in the Jacobian variety is given by principal divisors, i.e. describing the zeroes and poles of a rational function on $\mathscr{H}$. We denote by (P) the usual embedding $P \mapsto P - P_\infty$ of a hyperelliptic curve into its Jacobian variety.

In an Index Calculus context, the factor basis is $\mathscr{B} = \{(P) : x(P) \in \mathbb{F}_q\}$ and we solve PDP_{ng} instances: given $R \in \mathrm{Jac}(\mathscr{H})$, try to decompose it as

$$R = (P_1) + \cdots + (P_{ng}), \text{ where } P_i \in \mathscr{B} \text{ for all } 1 \leqslant i \leqslant ng. \tag{3.3}$$

This is equivalent to:

$$\exists\, f \in \mathscr{L}(ngP_\infty - R) : \mathrm{div}\, f + R = (P_1) + \cdots + (P_{ng}). \tag{3.4}$$

From Riemann-Roch theorem 2.12, $l(ngP_\infty - R) = (n-1)g + 1$, and, if we let $d_1 = \lfloor (n-1)g/2 \rfloor, d_2 = \lfloor ((n-1)g-1)/2 \rfloor$ and $d = (n-1)g = d_1 + d_2 + 1$, a natural basis of this space is

$$\{u, ux, \ldots, ux^{d_1}, y - v, (y-v)x, \ldots, (y-v)x^{d_2}\}.$$

Then any function $f \in \mathscr{L}(ngP_\infty - R)$ can be written

$$f(x,y) = u(x)\left(\sum_{i=0}^{d_1} a_{2i+1}x^i\right) + (y - v(x))\left(\sum_{i=0}^{d_2} a_{2i+2}x^i\right).$$

and we can set the coefficient of the a_{d+1} to 1. Since this describes whether ux^{d_1} or $(y-v)x^{d_2}$ has the pole of highest order at infinity, we also say that we normalize f at infinity. The goal is to determine the coefficients $a_1, \ldots, a_d$, such that expression (3.3) is verified. To achieve this we symbolically compute

$$\frac{\mathrm{Res}_y(f, \mathscr{H})}{u(x)} = F(x) = x^{ng} + \sum_{i=0}^{ng-1} N_{ng-i}(a_1, \ldots, a_d)x^i, \tag{3.5}$$

where $N_i \in \mathbb{F}_{q^n}[a_1, \ldots, a_d]$ and $\deg N_i = 2$. Let (x_i, y_i) be the coordinates of P_i, so that the roots of this polynomial are exactly the x_i. As $P_i \in \mathscr{B}$ for each i, a necessary condition is that F has coefficients in $\mathbb{F}_q$, or equivalently, to find values $a_1^*, \ldots, a_d^* \in \mathbb{F}_{q^n}$ such that all $N_i(a_1^*, \ldots, a_d^*)$ are in $\mathbb{F}_q$.

We can now use a Weil Descent. Let $1, t, \ldots, t^{n-1}$ be a $\mathbb{F}_q$-basis of $\mathbb{F}_{q^n}$, and write $a_i = \sum_{j=0}^{n-1} a_{i,j}t^j$ with $a_{i,j} \in \mathbb{F}_q$. We let $\mathbf{a} = (a_{0,0}, \ldots, a_{0,n-1}, \ldots, a_{d-1,0}, \ldots, a_{d-1,n-1})$ to simplify the notations, and get

$$N_i(a_0, \ldots, a_{d-1}) = \sum_{i=0}^{n-1} N_{i,j}(\mathbf{a})t^j, \tag{3.6}$$

with $N_{i,j} \in \mathbb{F}_q[\mathbf{a}]$. Values for which all N_i belong to $\mathbb{F}_q$ are exactly solutions of the multivariate polynomial system

$$\mathscr{N} = \{N_{i,j}(\mathbf{a}) = 0,\ 1 \leqslant i \leqslant ng,\ 1 \leqslant j \leqslant n-1\} \tag{3.7}$$

Such systems have $n(n-1)g$ quadratic equations in $n(n-1)g$ variables, and are generally of dimension 0. They are solved by the usual Gröbner basis strategy.

If this system has a solution $\mathbf{a}^*$ over $\mathbb{F}_q$, we have to check in addition that the specialized polynomial $F^*(x) = x^{ng} + \sum_{i=0}^{ng-1} N_i(\mathbf{a}^*)x^i$ is split over $\mathbb{F}_q$. When it is the case, its roots are the

abscissae of the P_i's, giving a decomposition of R. Heuristically, the probability of finding such decompositions is given by

$$\frac{\#\left(\mathscr{B}^{ng}/\mathscr{S}_{ng}\right)}{\#\mathrm{Jac}(\mathscr{H})^n} \approx \frac{q^{ng}}{(ng)!}\frac{1}{q^{ng}} = \frac{1}{(ng)!}.$$

Notice that the whole modelling adapts to the elliptic case, i.e. when $g = 1$. However the summation method presented in Section 3.4.3 needs less variables and gives better results in practice and is thus always preferred. We also see that the degree of the ideal grows very fast with the genus and the extension degree, so the probability of finding a PDP_{ng} solution drops exponentially fast as well. In practice, computations in this setting generally take too long as soon as $g > 4$ or $n > 2$ to even find relations in experiments.

Toy example for $\mathbf{n = 2, g = 2}$ in even characteristic. Let the fields be $\mathbb{F}_4 = \mathbb{F}_2[s]/\langle s^2+s+1\rangle$ and $\mathbb{F}_{16} = \mathbb{F}_4[t]/\langle t^2+t+s\rangle$. Select the curve as $\mathscr{H} : y^2+xy = x^5+x^3+(s+1)+st$. Fix $R(u,v) \in \mathrm{Jac}(\mathscr{H})$ with $u(x) = x^2+((s+1)+st)x+st$ and $v(x) = ((s+1)t+1)x$ and search for $f \in \mathscr{L}(4P_\infty - R)$. A generic function in this space looks like

$$f(x,y) = (a_1+x)u(x)+a_2(y-v(x)),$$

with $a_i = a_{i,0}+a_{i,1}t$, $a_i \in \mathbb{F}_4$ for $i = 0,1$, so the R-Decomposition polynomial is

$$\begin{aligned} F(x) = x^4+(a_2^2+st+s+1)x^3+(a_1^2+(st+s+1)a_2^2+a_2+st)x^2+\\ ((st+s+1)a_1^2+a_1a_2+(t+s)a_2^2)x+sta_1^2+ta_2^2. \end{aligned}$$

The Weil Descent leads to the following system:

$$\mathscr{N} : \begin{cases} a_{2,1}^2+s, \\ a_{1,1}^2+sa_{2,0}^2+sa_{2,1}^2+a_{2,1}+s, \\ sa_{1,0}^2+a_{1,0}a_{2,1}+sa_{1,1}^2+a_{1,1}a_{2,0}+a_{1,1}a_{2,1}+a_{2,0}^2+a_{2,1}^2, \\ sa_{1,0}^2+a_{1,1}^2+a_{2,0}^2+(s+1)a_{2,1}^2 \end{cases}$$

A Gröbner basis wrt a lexicographical ordering e.g. $a_{1,0} > a_{1,1} > a_{2,0} > a_{2,1}$ is then

$$\begin{cases} a_{1,0}+(s+1)a_{1,1}a_{2,0}a_{2,1}+a_{1,1}+(s+1)a_{2,0}^2a_{2,1}+s+1, \\ a_{1,1}^2+sa_{2,0}^2+a_{2,1}+1, \\ a_{2,0}^4+sa_{2,0}^2a_{2,1}+sa_{2,0}^2+a_{2,1}+s, \\ a_{2,1}^2+s \end{cases}$$

so that the ideal has degree 16. In this small example, we already see interesting properties. Indeed the first equation in the system is univariate, and the last one is a square because of the characteristic. In Chapter 6, Section 6.1, we will see that such properties are keys to reduce the degree of the ideals.

Part III

Contributions

Chapter 4

A Sieving approach to the Harvesting

The results of this Chapter have been published at LatinCrypt 2015: V. Vitse, A. Wallet, Improved Sieving on Algebraic Curves *[VW15].*

We focus on improving the known harvesting methods, dedicated to Jacobian varieties of algebraic curves defined over finite fields. We emphasize that all the other aspects of the index calculus method (such as the choice of the factor base, the processing of large prime relations and the linear algebra phase) are not modified. Sarkar and Singh proposed in [SS14] a sieving technique for harvesting relations in the hyperelliptic case, instead of the standard approach of Gaudry [Gau00] based on smooth reduced divisors. A very similar sieve had actually been used before by Joux and Vitse in [JV12], but in the different context of curves defined over extension fields and Weil restrictions. A sieving approach is usually a time-memory trade-off: instead of running numerous (expensive) computations, the results of (cheap) computations are stored for later use. This is precisely the goal of Sarkar and Singh's approach, where relations are sought by incrementing a table of counter depending on the result of some multiplications in the field, instead of testing the splitness of polynomials of fixed degree over the base field. It turns out that Sarkar and Singh's sieve has a simpler interpretation, which allows to generalize it to the Index Calculus introduced by Diem [Die06] for non-hyperelliptic curves, or more exactly small degree planar curves. In our experiments, the new non-hyperelliptic sieve improves Diem's original method, as well as its development by Diem and Kochinke [DK13], by a factor ranging from 3 to 7 approximately. Our reformulation does not rely on any list sorting and is more memory-efficient than [SS14] in hyperelliptic context. Additionaly, our approach prevents duplicated relations in a non-hyperelliptic setting by exhausting the whole "pencil" of lines passing through a base point before switching to a new one.

The presentation follows these steps. We begin by the case of hyperelliptic curves with a presentation and analysis of the sieving variant of Sarkar and Singh. We then introduce its simpler reformulation. The next section deals with algebraic curves of genus g admitting a plane model of degree $d \leq g+1$. We start by the classical ideas of using principal divisors associated to equations of lines to generate relations [Die06]. We then give the adaptation of our sieve to small degree curves, and compare it to Diem's method. We also briefly present the singularity-based technique of Diem and Kochinke and show that our sieve adapts again to this setting. Experiments and timings are reported in the last section.

4.1 Sieving for Hyperelliptic Curves

4.1.1 Sarkar and Singh's Sieve

A recent result of Sarkar and Singh [SS14] proposes a sieving approach to the relation search for hyperelliptic curves. In this method, with the same factor base $\mathscr{B}$ as in Sect. 3.3.3, we start from a weight g reduced divisor written as $D = [u,v] = \sum_{i=1}^{g}(P_i) - g(P_\infty)$, usually related to the challenge. We then consider all the weight $g+1$ semi-reduced divisors $D' = [u',v']$ that are linearly equivalent to $-D$; a relation is obtained each time u' is split (the factor base $\mathscr{B}$ is the same as in the previous version). The set of all the decompositions of $-D$ as

$$-D \sim \sum_{i=1}^{g+1}(Q_i) - (g+1)(P_\infty),$$

i.e. the set of all weight $g+1$ semi-reduced divisors linearly equivalent to $-D$, is in one-to-one correspondence with the set of divisors of functions in the Riemann-Roch space $\mathscr{L}(-D+(g+1)(P_\infty)) = \mathscr{L}(-\sum_{i=1}^{g}(P_i)+(2g+1)(P_\infty))$. This space is equal to $Span(u(x), y-v(x))$ (since functions in this space have poles at P_∞ only, of order at most $2g+1$, and vanish at the support of D), and thus the decompositions of $-D$ can be parametrized by an element $\lambda \in \mathbb{F}_q$.

We begin with the non-large-prime, non-sieving version of the algorithm. The relation search consists of two main loops, the outer one being simply a semi-random walk iterating through reduced divisors $D = [u,v] \sim aD_0 + bD_1$. The inner loop iterates over the value of the parameter $\lambda \in \mathbb{F}_q$. For each λ, we consider the function $f_\lambda = y - v(x) + \lambda u(x)$ and the corresponding semi-reduced divisor $D_\lambda = -D + \mathrm{div}(f_\lambda)$. The Mumford representation $[u_\lambda, v_\lambda]$ of D_λ is given by the formulae

$$\begin{cases} u_\lambda = c\frac{(\lambda u - v)^2 - h}{u} = c(\lambda^2 u - 2\lambda v + \frac{v^2-h}{u}) \\ v_\lambda = v - \lambda u \mod u_\lambda \end{cases},$$

where $c \in \mathbb{F}_q$ is the constant that makes u_λ monic. We obtain a relation each time D_λ is 1-smooth, i.e. when u_λ is split over $\mathbb{F}_q$; this happens heuristically with probability $1/(g+1)!$.

The main advantage of this relation search is that it admits a sieving version, in the spirit of [JV12]. The idea is to replace the inner loop in λ by an inner loop in $x \in \mathbb{F}_q$.

For each value of x, we compute the expression

$$S(x,\lambda) = \lambda^2 u(x) - 2\lambda v(x) + \frac{v(x)^2 - h(x)}{u(x)},$$

which becomes a quadratic polynomial in λ, and find the corresponding roots (for simplicity we can skip the values of x for which $u(x) = 0$). There are two distinct roots λ_0 and λ_1 if and only if $h(x)$ is a square in $\mathbb{F}_q$, and those roots are given by:

$$\lambda_0 = \frac{v(x) + h(x)^{1/2}}{u(x)}, \quad \lambda_1 = \frac{v(x) - h(x)^{1/2}}{u(x)}.$$

As explained by the authors, this step is very fast if a table containing a square root of $h(x)$ (if it exists) for each $x \in \mathbb{F}_q$ has been precomputed. We then store the corresponding couples (λ_0, x) and (λ_1, x). At the end of the inner loop, we look for the values of λ that have appeared $g+1$ times: this means that the corresponding polynomial u_λ has $g+1$ distinct roots, so that D_λ yields a relation, i.e. a decomposition of $-D$. In practice, we can either store each value of x in an array L of lists indexed by λ; each time a value of λ is obtained as a root of the quadratic expression, we append x to $L[\lambda]$. When $\#L[\lambda] = g+1$, we directly have the x-coordinates of the points in the support of $\mathrm{div}(f_\lambda) - D$, and a last step is then to compute back the y-coordinates using f_λ. Alternatively, we can simply maintain a counter array `Ctr`

indexed by λ and increment $\texttt{Ctr}[\lambda]$ each time λ is obtained as a root. When this counter reaches $g+1$, we factorize the corresponding split polynomial u_λ. This variant has the merit of saving memory at the expense of some duplicate computations, but is more interesting when g increases since the proportion of λ's yielding a relation becomes small.

The main speed-up is provided by the fact that at each iteration of the inner loop, we replace the splitting test and the eventual factorization of either the degree g polynomial u (in Gaudry's version) or the degree $g+1$ polynomial $S(\lambda,x)$ evaluated in λ, by the resolution of the degree 2 equation $S(\lambda,x)=0$, evaluated in x. This comes at the expense of a slightly lower decomposition probability, namely $1/(g+1)!$ instead of $1/g!$, and higher memory requirement. As already noticed in [JV12], a second advantage of this sieve is its compatibility with the double large prime variation. Indeed, once the "small prime" factor base $\mathscr{B}_s$ is constructed, it is sufficient to sieve among the values of $x \in \mathbb{F}_q$ corresponding to abscissae of its elements (the full sieving as described above can still be used in the construction steps of $\mathscr{B}_s$ if necessary). Since the cardinality of $\mathscr{B}_s$ is in $\Theta(q^\alpha)$ with $\alpha = 1-1/g$, this shortened sieving only costs $\tilde{O}(q^\alpha)$ instead of $\tilde{O}(q)$. We then look for the values of λ that have been obtained at least $g-1$ times. The corresponding polynomials u_λ have at least $g-1$ roots corresponding to small primes, and it just remains to test if it is indeed split, which happens with heuristic probability $1/2$ (in the case where λ has been obtained exactly $g-1$ times). Note that we cannot simply scan the array L or $\texttt{Ctr}$, as it would cost $\tilde{O}(q)$ (even with a very small hidden constant) and defeat our purpose. So additional care must be taken in the implementation in order to recover the interesting values of λ in only $\tilde{O}(q^\alpha)$, for instance using associative arrays, see [SS14] for details. Although it is not specified in the original paper, one can show that the asymptotic complexity of this variant is still in $\tilde{O}(q^{2-2/g})$ for fixed g, as in the work of Gaudry, Thomé, Thériault and Diem [GTTD07], but it is more efficient in practice, and the authors report a significant speed-up.

4.1.2 Sarkar and Singh's Sieve Revisited

As mentioned above, precomputing a table containing an eventual square root of $h(x)$ for each $x \in \mathbb{F}_q$ can significantly speed up the sieving phase (for a $\tilde{O}(q)$ overhead). But this table is actually nothing more than a list of the rational points of $\mathscr{H}$. Indeed, if y is a square root of $h(x)$ then (x,y) and $(x,-y)$ are exactly the two points in $\mathscr{H}(\mathbb{F}_q)$ with abscissa x, and this precomputation is actually performed when the factor base $\mathscr{B} = \{(P)-(P_\infty) : P \in \mathscr{H}(\mathbb{F}_q)\}$ is enumerated.

This means that we can modify Sarkar and Singh's sieve as follows. Recall that we are looking for functions $f_\lambda = y - v(x) - \lambda u(x)$ such that $-D + \mathrm{div}(f_\lambda)$ is 1-smooth. Instead of sieving over the value of $x \in \mathbb{F}_q$, or in a small subset corresponding to small primes, we directly sieve over $P = (x_P, y_P) \in \mathscr{B}$ or $\mathscr{B}_s$, and the corresponding value of λ is simply recovered as $\frac{y_P - v(x_P)}{u(x_P)}$. We give a pseudo-code of this sieve in Alg. 8.

The pseudo-code corresponds to the non-large-prime version. Details like the management of the list or associative array L and the update of M are omitted. As mentioned above, a simple counter array $\texttt{Ctr}$ can be used instead of L, requiring the factorization of $S(x,\lambda)$ for the update of M. If the double large prime variation is used, then the first inner loop iterates only over the elements of the small factor base $\mathscr{B}_s$, and in the second we test if $\#L[\lambda] \geq g-1$ and subsequently if the remaining factor of $S(x,\lambda)$ splits.

An easy improvement, not included in the pseudo-code for the sake of clarity, is to use the action of the hyperelliptic involution to divide by two the size of the factor base. We can then compute simultaneously the values of λ corresponding to $P = (x_P, y_P)$ and $\imath P = (x_P, -y_P)$. This saves one evaluation of u and of v at x_P, and one inversion of $u(x_P)$, although it is also possible to precompute all inverses. It is clear that this is basically a rewriting of Sarkar and

Singh's original sieve, so that the performances of both should be similar. However, we will now see that it is easier to adapt to the non-hyperelliptic case.

Algorithm 8 Sieving in the hyperelliptic case

Input: the set of rational points $\mathscr{B}$ of $\mathscr{H}$.
Output: the relation matrix M.

```
n_rel = 0.
repeat
  Choose a random reduced divisor D = [u,v] ~ aD_0 + bD_1.
  Initialize an array of lists L.
  for P = (x_P, y_P) ∈ B do
    Compute u(x_P) and v(x_P).
    if u(x_P) ≠ 0 then
      Compute λ = (y_P − v(x_P))/u(x_P).
      Append P to L[λ].
    end if
  end for
  for λ ∈ F_q do
    if #(L[λ]) = g + 1 then
      Update M.
      Increment n_rel.
    end if
  end for
until n_rel > #B
return the matrix M.
```

4.2 Sieving for Small Degree Curves

4.2.1 The Sieving Technique

We can easily adapt our sieving formulation to Diem's setting. The factor base remains the same set of points. Basically, in a first loop we iterate over points $P_1 = (x_1, y_1) \in \mathscr{C}_0(\mathbb{F}_q)$. The equation of a non-vertical line passing through P_1 is given by $(y - y_1) - \lambda(x - x_1) = 0$. The task is now to find the values of λ such that the line has d rational points of intersection with $\mathscr{C}$ without checking for smoothness. For this we then loop over $P_2 = (x_2, y_2) \in \mathscr{C}_0(\mathbb{F}_q)$ and compute the corresponding $\lambda = (y_2 - y_1)/(x_2 - x_1)$. But instead of looking for the intersection of the line with $\mathscr{C}$, we just append P_2 to the list $L[\lambda]$, where L is an array of lists; alternatively, we can simply increment a counter $\texttt{Ctr}[\lambda]$. If this counter reaches $d-1$, or if $L[\lambda]$ contains $d-1$ elements, we know that the line contains enough points and yields a relation. This is made precise in the pseudo-code of Alg. 9.

Note that in the inner loop we do not iterate over the elements of $\mathscr{B}$ that have already been considered in the outer loop. Indeed, after an iteration of the outer loop all the lines passing through the given point $P_1 = \mathscr{B}[i]$ have been surveyed, so there is no reason to scan this point again. In this way no line can be considered twice, and we avoid completely having to check for duplicate relations.

In Diem's version, each step requires the computation and factorization of $\frac{F(x,\lambda x+\mu)}{(x-x_1)(x-x_2)}$. The probability of finding a relation is $1/(d-2)!$, so that after q steps about $q/(d-2)!$ relations are harvested. By comparison, in our sieving each step requires a single division (or multiplication if the inverses are tabulated). The inner loop ends after about $\#\mathscr{B} \approx q$ steps, and yields $q/(d-1)!$ relations approximately: all the lines through $P_1 = (x_1, y_1)$ have been explored, and contain $d-1$ other points with probability $1/(d-1)!$. Thus we need $d-1$ times as many steps to obtain the same number of relations, but each step is much simpler,

Algorithm 9 Sieving for small degree curves

Input: the list of rational non-singular affine points $\mathscr{B} = \mathscr{C}_0(\mathbb{F}_q)$.
Output: the relation matrix M.

```
n_rel = 0.
for i = 1 to #B do
  (x_1, y_1) ← B[i]
  Initialize an array of lists L.
  for j = i+1 to #B do
    (x_2, y_2) ← B[j].
    if x_2 ≠ x_1 then
      Compute λ = (y_2 − y_1)/(x_2 − x_1).
      Append (x_2, y_2) to L[λ].
    end if
  end for
  for λ ∈ F_q do
    if #L[λ] = d − 1 then
      Update M.
      Increment n_rel.
    end if
    if n_rel > #B then
      return the matrix M.
    end if
  end for
end for
```

and the experiments of the next section confirm the important speed-up.

This sieve can be adapted straightforwardly to the double large prime variation: we just have to restrict both loops to the small factor base $\mathscr{B}_s$ (once it is constructed, if the version of [LL15] is followed), then we recover the values of λ such that $\#L[\lambda] \geq d-3$. When $\#L[\lambda] = d-3$, we still have to check if the remaining two points on the line are rational, which amounts to factorising a degree 2 polynomial. If $d = 4$, in Diem's version there are at most two remaining points on any line anyway; our new sieve is thus basically equivalent and does not provide a significant speed-up when using double large primes. However as soon as $d \geq 5$ it outperforms Diem's version, but the asymptotic complexity remains in $\tilde{O}(q^{2-2/(d-2)})$.

4.2.2 Sieving with Singularities

An article of Diem and Kochinke [DK13] tries to improve on the asymptotic complexity of the above method. The basic idea is to consider singular small degree plane models, and use a singular point as one of the points defining the lines cutting out the curve. Indeed, a singular point appears with a multiplicity greater than one in any line passing through it, so that there are fewer remaining points of intersection with $\mathscr{C}$, and the degree of the polynomial to test for smoothness is less than when two regular points are used. Unfortunately in general there are not enough singular points on a given planar curve to obtain sufficiently many relations. Thus an important part of Diem and Kochinke's work is to find a way to compute new singular plane models of degree $d \leq g+1$ for a given genus g curve, but this is outside of the scope of the present article; furthermore, the computation of the maps between the different models is not asymptotically relevant. Using Brill-Noether theory and considerations on special linear systems, they show that this method works for "general enough" non-hyperelliptic curves, of genus $g \geq 5$.

So we assume that we are given a degree d curve $\mathscr{C}$, of equation $F(x,y) = 0$, with a rational

singular point P_1 of multiplicity $m \geq 2$ (in most cases $m = 2$). The factor base is given by the rational points of the desingularization $\tilde{\mathscr{C}}$ of $\mathscr{C}$, i.e. $\mathscr{B} = \{(P) : P \in \tilde{\mathscr{C}}(\mathbb{F}_q)\}$. In the original version, for each other point P_2 in $\mathscr{B}$ or in the small factor base $\mathscr{B}_s$, the intersection of $\mathscr{C}$ with the line passing through P_1 and P_2 is computed as before. This amounts to finding the roots of the polynomial

$$\frac{F(x, \lambda x + \mu)}{(x - x_1)^m (x - x_2)},$$

which has degree $d - m - 1$. If it splits, which happens with probability about $1/(d-m-1)!$, we compute the intersection points $P_3, \ldots, P_{d-m-1}$ and obtain a relation that we can write as

$$D \sim (P_2) + (P_3) + \cdots + (P_{d-m-1}),$$

where D involves the singularity and the points at infinity. In the double large prime variation we keep this relation only if it involves no more than two large primes. To get rid of the divisor D on the left-hand side we would like to subtract one such relation from all the other ones. But in order to do this (using large primes) we need one relation involving only small primes ; if it does not exist a solution is then to add some points to $\mathscr{B}_s$. Since there are less points on the right-hand side than in Diem's first algorithm, the probability of finding a relation increases, and one can show that the overall complexity becomes $\tilde{O}(q^{2-2/(g-2)})$. Note that here again, some care must be taken to avoid duplicate relations, and in particular not all points P_2 but only a fraction of the factor base should be considered.

Now it is clear that our sieve can be naturally adapted to this new setting. Indeed, we can keep the inner loops of Alg. 9 ; the point (x_1, y_1) is now the singular point P_1, and we look for the values of λ that have been obtained $d - m$ times, or $d - m - 2$ times in the double large prime variation. Once again, this replaces the factorization of a degree $d - m - 1$ polynomial by a single division, and avoids checking for duplicate relations.

4.3 Experiments

We have experimented the harvesting techniques presented in this article for several curves of different genera, defined over different finite fields. All computations have been done using the computer algebra system Magma [BCP97] on an AMD Opteron™ 6176 SE@2.3GHz processor. We only implemented the non-large-prime version of the algorithms, the main reason being that we wanted the tests to be as simple as possible[1]. The curves have been generated with the command `RandomCurveByGenus`, which always returned a degree g curve (instead of $g+1$) for $g \geq 6$; for this reason the results in genus 6 are very close to those in genus 5 and we did not report them. For the non-sieving versions, we used associative arrays and sets to automate the check for duplicate relations, but this is more and more costly as the number of relations grows.

We give in Table 4.1 the comparison between Diem's method and our sieve; the values are the timings in seconds (on an Intel© Core i5@2.00Ghz processor) to obtain $p \approx \#\mathscr{F}$ relations, averaged over several curves.

In Table 4.2 we give timings comparing the new sieve with Diem and Kochinke's method. We did not implement the change of plane models; instead, we simply chose random curves possessing rational singular points, and used one of them as the base point for the relation search. In the sieving version all the relations involving lines passing through the singularity were computed, whereas in the non-sieving case we only iterated through half of the basis, as suggested in [DK13]. For this reason the values correspond to the timings in seconds needed to obtain 1000 relations, again averaged over several curves.

[1] More fundamentally, large prime variations are interesting for the asymptotic complexity analysis, but are not always well-suited in practice ; other methods such as the Gaussian structured elimination [LO91] can be more efficient.

p		78137	177167	823547	1594331
Genus 3, degree 4	Diem	11.57	27.54	135.1	266.1
	Diem + sieving	3.65	9.38	46.96	94.60
	Ratio	3.16	2.95	2.88	2.81
Genus 4, degree 5	Diem	51.85	122.4	595.8	1174
	Diem + sieving	15.58	40.01	195.1	387.6
	Ratio	3.33	3.06	3.05	3.03
Genus 5, degree 6	Diem	229.4	535.8	2581	5062
	Diem + sieving	75.66	199.0	969.3	1909
	Ratio	3.03	2.69	2.66	2.65
Genus 7, degree 7	Diem	1382	3173	14990	29280
	Diem + sieving	458.5	1199	5859	11510
	Ratio	3.02	2.65	2.56	2.54

Table 4.1: Comparisons of the new sieve with Diem's classical method

p		78137	177167	823547	1594331
Genus 5, degree 6	Diem & Kochinke	1.58	1.60	1.69	1.76
	DK + sieving	0.43	0.45	0.52	0.61
	Ratio	3.67	3.60	3.23	2.90
Genus 7, degree 7	Diem & Kochinke	8.59	8.68	8.97	9.20
	DK + sieving	1.21	1.25	1.56	1.93
	Ratio	7.13	6.96	5.74	4.77

Table 4.2: Comparisons of the new sieve with Diem and Kochinke's method

4.4 Conclusion

We have shown in this work that a reformulation of Sarkar and Singh's sieve [SS14], namely sieving over points instead of x-coordinates, gives a simpler presentation of the harvesting phase of the index calculus algorithm on hyperelliptic curves. More importantly, it can be naturally adapted to Diem and Kochinke's index calculus for non-hyperelliptic curves [Die06, DK13]. Our experiments show that the new sieve clearly outperforms the relation search of the other methods in all circumstances and should always be preferred.

Chapter 5

Summation Ideals

The results of this Chapter are part of an article which has been submitted at Design, Codes and Cryptography journal.

This section focuses on an alternate modelling of PDP_m instances in all genera, derived from Gaudry and Diem's propositions for elliptic curves defined over extension fields. Recall that their approach is practical only if the extension degree is "small", which means that the degree of the extension admits a small factor k (≤ 5). The overall idea is to describe the algebraic dependance of points' abscissae, when the points form a vanishing sum like

$$P_1 + \cdots + P_m = \mathscr{O}.$$

Once the size m of the sum is fixed, the set of such points forms an algebraic variety, whose projection over the x coordinates is an hypersurface generated by the Summation Polynomial; this is why computing the Summation Polynomial is sometime referred as "projecting the group law on the x-line". If we denote by S_m the Summation Polynomial for sums of size m, then we have

$$\sum_{i=1}^{m} P_i = \mathscr{O} \;\Leftrightarrow\; S_m(x(P_1), \ldots, x(P_m)) = 0,$$

which can be taken as a defining property. As showed by Diem [Die11] and Gaudry [Gau09], Summation Polynomials can be used to solve PDP_n instances in Decomposition attacks over $\mathbb{F}_{q^n}$. In this setting an additional condition is that the points belong to the factor base, i.e. that $x(P_i) \in \mathbb{F}_q$ (up to a linear change of variables). The Weil Descent on S_n's coefficients then leads to 0-dimensional systems. As those systems involve less variables than their Nagao counterparts, it is hoped that they are easier to solve. This proved to be experimentaly true for elliptic curves.

In Section 5.1 we propose a general definition of Summation Polynomials for higher genus curves by focusing on the geometric description of vanishing sums. With Index-Calculus in mind, we focus on hyperelliptic curves, but the whole modelling can be adapted to any type of curves. Assume $\mathscr{H}$ is given by an imaginary model, and let (P) stands for the canonical embedding $\mathscr{H} \longrightarrow \mathrm{Jac}(\mathscr{H})$ using the point at infinity. We introduce the m-Summation Variety

$$\mathscr{V}_m = \{(P_1, \ldots, P_m) : \sum_{i=1}^{m} (P_i) = \mathscr{O}\},$$

and we give a polynomial parametrization of $\mathscr{V}_m$ using Riemann-Roch spaces. This allows us to compute the associated ideal for its projection over the abscissae: we define this ideal as the *m-th Summation Ideal* and any set of generators as *Summation Sets of Polynomials*. The projection of $\mathscr{V}_m$ is no longer an hypersurface when $g \geq 2$, since its codimension is at least g in general. It is noteworthy that we recover the classical elliptic Summation Polynomial of

Section 3.4.3 when $g = 1$, as already noted in [JV13]. We give examples of such sets for the smallest parameters $g = 1, 2$ and $m = 3, 5$ in positive characteristic, completed by computation timings and informations for several couples of (g, m).

Section 5.4 deals with practical usage of Summation Sets for Index Calculus purpose. When modelling PDP_m instances there are two approaches that can be used. On the one hand we can first compute a Summation set, then specialize it to the coordinates given by the input $R \in \mathrm{Jac}(\mathscr{H})$: we call this approach *Project-then-specialize*. On the other hand it is possible at each new input to compute a Summation Set already evaluated using a modification of $\mathscr{V}_m$'s parametrization: we refer to this approach as *Specialize-then-project*. Classically in the elliptic setting, the first one is used but it makes no difference to use the other. This is specific to the case $g = 1$: indeed, using the Project-then-specialize approach when $g \geq 2$ leads to ideals with degree $2^{(ng-1)n} = 2^{n(g-1)} \cdot d_{\mathrm{Nag}}$ after Weil Descent, where $d_{\mathrm{Nag}} = 2^{n(n-1)g}$ is the degree obtained with a classical Nagao approach. We show that the Specialize-then-project method leads to ideals with degree d_{Nag}. To this effect we chose a more geometry-flavoured description, to highlight the importance of the projection of $\mathscr{V}_m$ over the abscissae. Next the *Specialized Summation Variety* is introduced as

$$\mathscr{V}_{m,R} = \{(P_1, \ldots, P_m) : \sum_{i=1}^{m} (P_i) = R\},$$

for a fixed $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. We also give a parametrization of this variety, and introduce *Specialized Summation Sets* as generators of its projection over the abscissae. Finally we analyze the PDP_{ng} solving process using Project-then-Specialize and Specialize-then-project methods to conclude that the second gives ideals of degree $2^{n(n-1)g}$ and is thus better as soon as $g \geq 2$. The analysis shows that, when $g = 1$, it makes no difference to use Summation Polynomials or Specialized Summation Polynomials: the latter is only the evaluation of the former at $x(R)$. The Chapter concludes with some more general answers: a recursive method to compute Elliptic Summation Polynomials is known (Proposition 3.13). In higher genus, we explain why such a method cannot be expected, which may be summed up by the fact that the cover $x : \mathscr{H} \longrightarrow \mathbb{P}^1$ over degree 2 induces a morphism $\mathrm{Jac}(\mathscr{H}) \longrightarrow (\mathbb{P}^1)^g$ of degree 2^g. For more or less the same reason, unfortunately, the natural group of symmetries acting on the Jacobian Variety does not preserve the Summation Variety that we defined. Therefore it seems unlikely that such symmetries can be exploited to reduce the degree of the target variety, as in the work of [FGHR14, FHJ+14, GG14].

5.1 A geometric description of PDP_m instances

Following [Die11] and [JV13], we want to generalize Semev's summation polynomials to any hyperelliptic curve. The elliptic case is covered up when $g=1$. To simplify the presentation we assume for the moment that the base field $\mathbb{F}$ is algebraically closed, but the whole presentation extends easily to arbitrary fields. For any hyperelliptic curve $\mathscr{H}$, recall that (P) stands for the classical embedding $P-P_\infty$ of $\mathscr{H}$ into $\mathrm{Jac}(\mathscr{H})$. We now describe the variety of all sums like

$$(P_1)+\cdots+(P_m)=\mathscr{O},\ P_i(x_i,y_i)\in\mathscr{H}. \tag{5.1}$$

To find a relation as (5.1), we need an $f\in\mathscr{L}(mP_\infty)$ such that $\operatorname{div} f=\sum_{i=1}^m(P_i)$.

Remark 5.1. *If $m<2g+1$ then no basis of $\mathscr{L}(mP_\infty)$ can contain a function involving y, since it has a pole of order $2g+1$ at P_∞. But if f involves only monomials in x and vanishes at P, then it also vanishes at $-P$ and thus $(P)+(-P)$ is in the support of $\operatorname{div} f$. Any such divisor reduces to $\mathscr{O}$ in the Jacobian, and therefore we need at least $m\geq 2g+1$. We will always assume that this is the case in this section.*

Definition 5.2. *Let $m\geq 2g+1$. Let $\mathscr{H}:y^2+h_1(x)y=h_0(x)$ be an hyperelliptic curve of genus g in imaginary model defined over a field $\mathbb{F}$, and set $\mathscr{O}$ to be the neutral element of $\mathrm{Jac}(\mathscr{H})$. The set $\mathscr{V}_m=\{(P_1,\ldots,P_m):\sum_{i=1}^m(P_i)=\mathscr{O}\}$ is called the m-Summation Variety of $\mathscr{H}$.*

We now find generators for the (ideal associated to) $\mathscr{V}_m$; this shows that $\mathscr{V}_m$ is an algebraic variety. Using Definition 5.7, this set can also be seen as the kernel of a morphism of Abelian variety, which gives an alternate proof, less computationally-flavoured. The space $\mathscr{L}(mP_\infty)$ has dimension $m-g+1$ and a natural basis is the set

$$\{1,x,\ldots,x^{d_1},y,yx,\ldots,yx^{d_2}\},$$

with $d_1=\lfloor m/2\rfloor$ and $d_2=\lfloor(m-2g-1)/2\rfloor$. Let $d=m-g$ and $\mathbf{a}=(a_0,\ldots,a_d)$ the coordinates in this basis, so that a generic function in $\mathscr{L}(mP_\infty)$ can be written as an element in $(\mathbb{F}[\mathbf{a}])[x,y]$ as:

$$f(x,y)=p(x)+q(x)y, \tag{5.2}$$

with $p(x)=\sum_{i=0}^{d_1}a_ix^i, q(x)=\sum_{i=0}^{d_2}a_{i+d_1+1}x^i$. We normalize f at infinity — see Section 3.4.4, the reader may assume that $a_d=1$. In those coordinates, the generic norm of f is then the monic polynomial in $(\mathbb{F}[\mathbf{a}])[x]$ given by

$$N(f)=p(x)^2+p(x)q(x)h_1(x)-q(x)^2h_0(x) \tag{5.3}$$

$$=(-1)^m(x^m+\sum_{i=0}^{m-1}N_{m-i}(\mathbf{a})x^i), \tag{5.4}$$

with $\deg N_i=2$ for $1\leq i\leq m$. Assume now that f describes a sum $(P_1)+\cdots+(P_m)=\mathscr{O}$ and let $x_i=x(P_i),\mathbf{x}=(x_1,\ldots,x_m)$. The generic norm $N(f)$ vanishes exactly at all the x_i's so we have

$$N(f)=(-1)^m\prod_{i=1}^n(x-x_i) \tag{5.5}$$

$$=(-1)^m(x^m+\sum_{i=0}^{m-1}(-1)^{n-i}e_{n-i}(\mathbf{x})x^i), \tag{5.6}$$

where e_i stands for the i-th elementary symmetric polynomial in m variables $\mathbf{X}=(X_1,\ldots,X_m)$. Equating coefficients of (5.4) and (5.6) we obtain a polynomial system

$$\begin{cases} N_1(\mathbf{a})=e_1(\mathbf{X}),\\ \quad\vdots\\ N_m(\mathbf{a})=(-1)^{m+1}e_m(\mathbf{X}),\end{cases} \tag{5.7}$$

of m equations in $2m-g$ variables. We claim that the projection of $\mathscr{V}_m$ onto the x-line can be parametrized by the ideal generated by those polynomials.

Proposition 5.3. *Let $\mathscr{H}$ be a hyperelliptic curve in imaginary model of genus g with canonical embedding $P \mapsto (P)$ in its Jacobian $\mathrm{Jac}(\mathscr{H})$. For any $m \geq 2g+1$, define $\mathscr{I}_m$ as the ideal in $\mathbb{F}[\mathbf{a},\mathbf{X}]$ generated by system (5.7). The projection of the m-Summation Variety on the x-line admits a parametrization by $\mathscr{I}_m$.*

Proof. Let π_x denote the projection on the x line. The description above tells us that $\pi_x(\mathscr{V}_m)$ is contained in the variety parametrized by $\mathscr{I}_m$ (in the sense of Definition 1.69). For the reverse inclusion, assume that $(a_1^*,\dots,a_{m-g}^*,x_1^*,\dots,x_m^*) \in V(\mathscr{I}_m)$. We want to show that there exist $y_1,\dots,y_m \in \mathbb{F}$ such that $P_i(x_i^*,y_i) \in \mathscr{H}$ and $\sum_{i=1}^m (P_i) = \mathscr{O}$. We start by specializing the generic function f from (5.2) with $a_1^*,\dots,a_{m-g}^*$. This gives an element $f^* \in \mathscr{L}(mP_\infty)$. Next, provided $q(x_i^*) \neq 0$, we can set $y_i = -\frac{p(x_i^*)}{q(x_i)}$ for $1 \leq i \leq m$ and we can easily check that $P_i \in \mathscr{H}$ and that $f^*(x_i^*,y_i) = 0$. Now assume $q(x_i^*) = 0$ for at least one i. By (5.4) we have $p(x_i^*) = 0$ and then $f^*(x_i^*,y) = 0$ for all $y \in \mathbb{F}$, and $f^*(x,y) = (x-x_i^*)\tilde{f}(x,y)$ with $\tilde{f} \in \mathscr{L}((m-2)P_\infty)$. Since $\mathbb{F}$ is algebraically closed[1], the polynomial $y^2 + h_1(x_i^*)y + h_0(x_i^*)$ have roots y_i and $-y_i - h_1(x_i^*)$ and thus f^* vanishes at $P_i(x_i^*,y_i)$ and $-P_i(x_i^*,-y_i-h_1(x_i^*))$. □

"Projecting on the x-line" means we want a condition involving only the abscissae of points in (5.1), or equivalently, we want to eliminate the variables coming from the coordinates of f. Geometrically, this means we want to find polynomial equations defining the projection of $\mathscr{V}_m$ onto the m last variables. Assume for simplicity that we are in a generic situation. Being described by m equations in a $2m-g$ dimensional space, $\mathscr{V}_m$ has dimension $m-g$, so that the projection on a m-dimensional subspace will have codimension g. This means that a minimal generating family of the associated ideal should have at least g elements. For $g \geq 2$, this means there is no hope to obtain a *unique* summation polynomial. Instead, we will have a *set* of summation polynomials, which we can now define:

Definition 5.4 (Summation polynomials for hyperelliptic curves)**.** *Let $\mathscr{H}$ be a hyperelliptic curve of genus g given by a Weierstrass equation $y^2 + h_1(x) = h_0(x)$, and $m \geq 2g+1$. The m-th summation ideal associated to $\mathscr{H}$ is defined as the elimination ideal $\mathscr{I}_m \cap \mathbb{F}[\mathbf{X}]$ where $\mathscr{I}_m$ is the ideal in $\mathbb{F}[\mathbf{a},\mathbf{X}]$ generated by equations (5.7). Any set $\mathbb{S}_m \subset \mathbb{F}[\mathbf{X}]$ generating $\mathscr{I}_m \cap \mathbb{F}[\mathbf{X}]$ is called a set of m-th summation polynomials, or a m-th summation set, for $\mathscr{H}$.*

Remark 5.5. *Geometrically, $\mathbb{S}_m$ is a m-th summation set if it verifies*

$$V(\mathbb{S}_m) = V(\mathscr{I}_m \cap \mathbb{F}[\mathbf{X}]).$$

Describing the projection of $\mathscr{V}_m$ can be done by computing a Gröbner basis of the elimination ideal $\mathscr{I}_m \cap \mathbb{F}[\mathbf{X}]$. For any set $\mathbb{S}$ of polynomials, we denote by $\mathbb{S}(x)$ the set of all elements in $\mathbb{S}$ evaluated at x. The next proposition generalizes the definition of Semaev's summation polynomials:

Proposition 5.6. *For any $m \geq 2g+1$, a set $\mathbb{S}_m$ of m-th summation polynomials associated to $\mathscr{H}$ exists, and it verifies:*

$$\mathbb{S}_m(\mathbf{x}) = 0 \Leftrightarrow \exists\, y_1,\dots,y_n \in \mathbb{F} \text{ such that } P_i(x_i,y_i) \in \mathscr{H}, 1 \leqslant i \leqslant m, \text{ and } (P_1) + \cdots + (P_m) = \mathscr{O}.$$

Proof. The existence of summation sets is clear from the Hilbert Basis theorem and the existence of a Gröbner basis for any ideal in $\mathbb{F}[\mathbf{X}]$. Now if $\mathbf{x}$ is in $V(\mathscr{I}_m \cap \mathbb{F}[\mathbf{X}])$, according to the Extension Theorem 1.63 we know that there exists $\mathbf{a} = (a_1^*,\dots,a_{m-g}^*)$ such that $(\mathbf{a},\mathbf{x}) \in \mathscr{V}_m$. The conclusion is basically Proposition 5.3. □

[1] In the general case, we look for y_i in the algebraic closure of $\mathbb{F}$ anyway.

When $g = 1$, the ideal is principal and we recover the fact that, in the elliptic case, the m-th summation polynomial described in Section 3.4.3 is unique (up to a constant). Using variables $e_1, \ldots, e_m$ for expressions $e_i(X_1, \ldots, X_m)$ before eliminating $\mathbf{a}$ in system (5.7) allows us to compute directly a m-th Summation Set expressed with the elementary symmetric functions. Therefore from now on, we always consider $\mathscr{V}_m$ as the m-th Summation Variety quotiented by the action of the symmetric group.

In fact, using variables for the symmetric expressions in system (5.7) gives a *polynomial parametrization* of $\mathscr{V}_m$. From Corollary 1.72, $\mathscr{V}_m$ is irreducible and the defining equations generate a radical ideal. This gives a proof that the (symmetrized) elliptic Summation Polynomial is irreducible. While the modelling focuses on the hyperelliptic case, it can be adapted straigthforwardly to non-hyperelliptic curves as well by studying bases of $\mathscr{L}(m\mathscr{O})$, where $\mathscr{O}$ is a distinguished point of the curve. We ran some experiments for superelliptic curves $(y^g = f(x))$ and $\mathscr{C}_{a,b}$ curves of small genus but we did not investigate further as such curves are not considered in practice.

5.2 Examples of Summation Sets

We first show that the third summation for an elliptic curve expressed in the elementary symmetric functions can indeed be computed with our modelling. We then present some sets of summation polynomials for genus 2 curves and discuss the expected codimension of the projected variety.

5.2.1 Elliptic Summation Polynomial revisited

In this Section we assume the characteristic of $\mathbb{F}$ to be odd and not equal to three, but the whole process can be adapted easily to every characteristic as well. In this setting it is well-known that every elliptic curve admits a Weierstrass equation of the form

$$E : y^2 = x^3 + Ax + B.$$

We want to describe the simplest non trivial sum $P_1 + P_2 + P_3 = \mathscr{O}, P_i(x_i, y_i) \in E(\mathbb{F})$ so that we need a basis of $\mathscr{L}(3\mathscr{O})$. A convenient one is given by the set $\{1, x, y\}$, so that we are in fact looking at the intersection between lines and the curve (this is also the geometric way to see the addition for elliptic curves, the so-called *chord-tangent method*). Such a line admits an equation $f(x,y) = y - (a_1x + a_0)$, so using (5.4) we get:

$$\begin{aligned} N(f) &= (a_1x + a_0)^2 - (x^3 + Ax + B) \\ &= -x^3 + a_1^2x^2 + (2a_0a_1 - A)x + a_0^2 - B \end{aligned}$$

and the norm should also have roots exactly in the x_i:

$$N(f) = -\prod_{i=1}^{3}(x - x_i) = -x^3 + e_1x^2 - e_2x + e_3 \tag{5.8}$$

with e_i being the i-th elementary symmetric polynomial in the x_i. We are led to the following polynomial system:

$$\begin{cases} a_1^2 = e_1, \\ 2a_0a_1 - A = -e_2, \\ a_0^2 - B = e_3. \end{cases} \tag{5.9}$$

Let $\mathscr{I}$ be the ideal generated by (5.9) in $\mathbb{F}[a_0, a_1, e_1, e_2, e_3]$. We compute a Gröbner Basis of $\mathscr{I} \cap \mathbb{F}[e_1, e_2, e_3]$ and find

$$\{e_2^2 - 4e_3e_1 - 2Ae_2 - 4Be_1 + A^2\}$$

which is indeed the representation of S_3 associated to E, using elementary symmetric functions. With this modelling, it is very fast to compute summation polynomials up to $n=5$. For greater n the recursive method using partial symmetrization [JV13] at each step prove to be much faster.

5.2.2 First Summation Sets in genus 2

Odd characteristic We assume for simplicity that $\mathbb{F}$ has characteristic $\neq 5$. Then an imaginary hyperelliptic curve admits a Weierstrass equation $\mathscr{H}: y^2 = x^5 + h_3x^3 + h_2x^2 + h_1x + h_0$, with $h_i \in \mathbb{F}_q$. Using Section 5.1 the smallest interesting decomposition is obtained for $m = 2g+1 = 5$:

$$(P_1) + \cdots + (P_5) = \mathscr{O}.$$

A convenient $\mathbb{F}_q$-basis for $\mathscr{L}(5P_\infty)$ is given by $\{1, x, x^2, y\}$, and describes the family of parabolas in $\mathbb{F}^2$. Keeping notations consistent with previous Sections, a generic function in this space is $f(x,y) = y - (a_2x^2 + a_1x + a_0)$, with generic norm

$$\begin{aligned} N(f) &= (a_1x^2 + a_1x + a_0)^2 - (x^5 + h_3x^3 + h_2x^2 + h_1x + h_0) \\ &= -x^5 + a_2^2x^4 + (2a_2a_1 - h_3)x^3 + (a_1^2 + 2a_2a_0 - h_2)x^2 + (2a_0a_1 - h_1)x + a_0^2 - h_0, \end{aligned}$$

and can be also expressed as

$$\begin{aligned} N(f) &= -\prod_{i=1}^{5}(x - x_i) \\ &= -x^5 + e_1x^4 - e_2x^3 + e_3x^2 - e_4x + e_5. \end{aligned}$$

Equating coefficients gives the following system:

$$\begin{cases} a_0^2 - h_0 = e_5, \\ 2a_0a_1 - h_1 = -e_4, \\ a_1^2 + 2a_2a_0 - h_2 = e_3, \\ 2a_2a_1 - h_3 = -e_2, \\ a_2^2 = e_1. \end{cases}$$

To eliminate a_0, a_1, a_2 we compute a Gröbner basis for the elimination ideal. If we choose a lexicographical order with $a_0 > a_1 > a_2 > e_5 > \ldots > e_1$, we find after elimination:

$$\begin{aligned}
S_{5,1} = {} & (8e_2 - 8h_3)e_5^2 + ((4h_2 - 4e_3)e_4 - 16h_0e_2 + 16h_0h_3 - 4h_1h_2 + 4h_1e_3)e_5 + e_4^3 - 3h_1e_4^2 + \\
& (4h_0e_3 - 4h_0h_2 + 3h_1^2)e_4 - 8h_0^2h_3 + 4h_0h_1h_2 - h_1^3 - 4h_0h_1e_3 + 8h_0^2e_2, \\
S_{5,2} = {} & 16e_5^2e_1 + ((-2h_3 + 2e_2)e_4 + 2h_1h_3 - 4h_2^2 + 8h_2e_3 - 32h_0e_1 - 4e_3^2 - 2h_1e_2)e_5 + (e_3 - h_2)e_4^2 + \\
& (-2h_0e_2 - 2h_1e_3 + 2h_0h_3 + 2h_1h_2)e_4 + 4h_0e_3^2 + (-8h_0h_2 + h_1^2)e_3 + 2h_0h_1e_2 + 16h_0^2e_1 - \\
& 2h_0h_1h_3 + 4h_0h_2^2 - h_1^2h_2, \\
S_{5,3} = {} & (8e_4e_1 + (-4e_2 + 4h_3)e_3 - 4h_2h_3 - 8h_1e_1 + 4h_2e_2)e_5 + (-h_3 + e_2)e_4^2 + \\
& (-8h_0e_1 - 2h_1e_2 + 2h_1h_3)e_4 + (-4h_0h_3 + 4h_0e_2)e_3 + (-4h_0h_2 + h_1^2)e_2 + 8h_0h_1e_1 + \\
& 4h_0h_2h_3 - h_1^2h_3, \\
S_{5,4} = {} & (-2h_3e_2 + h_3^2 + e_2^2)e_5 + 2h_1e_4e_1 - h_0e_2^2 + 2h_0h_3e_2 - e_4^2e_1 - h_0h_3^2 - h_1^2e_1, \\
S_{5,5} = {} & (-8h_3e_1 + 8e_2e_1)e_5 + (-4e_3e_1 + e_2^2 - 2h_3e_2 + 4h_2e_1 + h_3^2)e_4 + 4h_1e_3e_1 - h_1e_2^2 + \\
& (-8h_0e_1 + 2h_1h_3)e_2 + (8h_0h_3 - 4h_1h_2)e_1 - h_1h_3^2, \\
S_{5,6} = {} & 16e_5e_1^2 + (2e_2e_1 - 2h_3e_1)e_4 - 4e_3^2e_1 + (h_3^2 + e_2^2 - 2h_3e_2 + 8h_2e_1)e_3 - h_2e_2^2 + (2h_2h_3 - 2h_1e_1)e_2 - \\
& h_2h_3^2 + (2h_1h_3 - 4h_2^2)e_1 - 16h_0e_1^2, \\
S_{5,7} = {} & 8e_4e_1^2 + (-4e_2e_1 + 4h_3e_1)e_3 + e_2^3 - 3h_3e_2^2 + (4h_2e_1 + 3h_3^2)e_2 - 4h_2h_3e_1 - 8h_1e_1^2 - h_3^3.
\end{aligned} \tag{5.10}$$

For $n=6$ the polynomials are already too large to be displayed on paper. For instance, once specialized to a curve, the smallest polynomial in a DRL basis for the case $n=6$ has 172

monomials in 6 variables, and there are 14 polynomials in the basis. For the lexicographical order, the smallest polynomial has 382 monomials and there are 27 polynomials in the basis.

We also see that $S_{5,7}$ in the list of generators (5.10) depends only on the $e_1,\dots,e_4$. This has the following interpretation. First assume that e_1,e_2 and e_3 are given, with $e_1 \neq 0$ and denote by e_i^* the value of the corresponding i-th symmetric function. Solving the linear equation $S_{5,7}(e_1^*,\dots,e_3^*,e_4)=0$ we find the value e_4^*. Next, as the leading coefficient e_1^2 of $S_{5,6}(e_1^*,\dots,e_4^*,e_5)$ is non zero, then this polynomial is linear in e_5; it is then straightforward to find its root and thus an element of $V(\mathbb{S}_5)$. If $e_1^*=0$, from the expression of the function we see that this means the function represents a line. It could indeed happen that its intersection with the curve has 5 points. There are approximately q^2 lines (up to a constant factor) in the space $\mathscr{L}(5P_\infty)$, hence the probability of considering one is $1/q$, which will be very low for practical fields. From a geometrical point of view it means that the variety generated by $S_{5,6}$ and $S_{5,7}$ should have the same dimension as the whole variety. Indeed, the codimension of the variety is $g=2$ in this case (see Section 5.1). In practice, it means the major part of the sums of length 5 will make $S_{5,1},\dots,S_{5,5}$ vanish as well.

Even characteristic The general case gives equations such as $\mathscr{H}: y^2+h_1(x)y=h_0(x)$, with $\deg h_1 \leq 2$ and $\deg h_0 = 5$. If h_{ij} is the j-th coefficient of h_i, we see using (5.4) that $\mathscr{V}_5$ is given by

$$\begin{cases} a_0^2+h_{10}a_0+h_{00}=e_5, \\ h_{10}a_1+h_{11}a_0+h_{01}=e_4, \\ h_{10}a_2+a_1^2+h_{11}a_1+h_{12}a_0+h_{02}=e_3, \\ h_{11}a_2+h_{12}a_1+h_{03}=e_2, \\ a_2^2+h_{12}a_2+h_{04}=e_1. \end{cases}$$

We first compute a Gröbner basis on an elimination ideal $\mathscr{I}\cap\mathbb{F}_{2^d}[\mathbf{e}]$ and a second Gröbner basis for a lexicographical ordering such that $e_5>\ldots>e_1$:

$$\begin{aligned} S_{5,1} = {} & h_{11}^3e_5+h_{11}e_4^2+(h_{10}^2h_{12}+h_{10}h_{11}^2)e_4+h_{10}^2h_{11}e_3+h_{10}^3e_2+h_{00}h_{11}^3+ \\ & h_{01}^2h_{11}+h_{01}h_{10}^2h_{12}+h_{01}h_{10}h_{11}^2+h_{02}h_{10}^2h_{11}+h_{03}h_{10}^3, \\ S_{5,2} = {} & h_{12}^3e_4+h_{11}h_{12}^2e_3+h_{11}e_2^2+(h_{10}h_{12}^2+h_{11}^2h_{12})e_2+h_{11}^3e_1+h_{01}h_{12}^3+ \\ & h_{02}h_{11}h_{12}^2+h_{03}^2h_{11}+h_{03}h_{10}h_{12}^2+h_{03}h_{11}^2h_{12}+h_{04}h_{11}^3. \end{aligned}$$

Estimations on the codimension of the projection at the end of Section 5.1 are further confirmed, as $S_{5,2}$ depends only on 4 variables e_1,e_2,e_3,e_4, and that the projected variety is generated by two polynomials. Moreover, we notice that the representation of the polynomials is much sparser than its odd characteristic counterpart, as it was already the case in genus 1.

Finally, if $E: y^2+xy=x^3+Ax^2+B$ is an ordinary elliptic curve defined over $\mathbb{F}_{2^d}$, the first summation polynomial expressed in the e_i is given by

$$S_3(e_1,e_2,e_3)=e_2^2+e_3+B.$$

This equation is very close to the canonical equation of a type II genus 2 curve over $\mathbb{F}_{2^d}$ with d odd which is $y^2+xy=x^5+f_3x^3+\varepsilon x^2+f_0$, $\varepsilon\in\mathbb{F}_2$, see 6.1.3. As a comparison, a set of 5-th summation polynomials for a type II curve is:

$$\begin{aligned} S_{5,1}(e_1,e_2,e_3,e_4,e_5) &= e_4^2+e_5+f_0, \\ S_{5,2}(e_1,e_2,e_3,e_4,e_5) &= e_2^2+e_1+f_3^2, \end{aligned}$$

and we see that their expressions are also very similar to the genus 1 case.

5.3 Timings and Experiments

Timings in odd characteristic Table 5.1 shows the details of the computations for the first sets of summation polynomials, expressed in the symmetric elementary functions $e_1, \ldots, e_m$, for hyperelliptic curves with $g = 2,3,4$. The base field is $\mathbb{F}_{65521}$ and all the curves are given by a general Weierstrass equation with randomized coefficients. The computation of the elimination ideal was carried with the Magma system [BCP97], on a Intel® Xeon®@2.93GHz processor. We only computed Gröbner bases with respect to weighted degree orderings; in the simplest cases, it is also possible to compute lexicographical bases. The time is expressed in seconds, and averaged over several curves. If all variables could not be eliminated we let the number we could achieve in parenthesis in the eliminated column, together with the time needed to do so. Next columns give the average number (rounded) of monomials and average total degree of elements in the summation set. The degree accounts for weight i on the variable e_i. When a Summation Set $\mathbb{S}_m$ can be computed, we also compute $\deg V(\mathbb{S}_m)$ using the Hilbert Series, see last column. We interrupted the computations if any of our strategies could not compute the basis in less than 8 hours or if the needed memory exceeded 120 Go.

genus g	m	#vars	#elim	Time	#set	Avg. len.	Avg. deg.	$\deg V(\mathbb{S}_m)$
2	5	8	3	0.000 s	7	13	9	4
	6	10	4	0.220 s	14	326	21	8
	7	12	5	209.810 s	58	7808.8	48	16
	8	14	6 (4)	> 5000 s	-	-	-	-
3	7	11	4	0.02 s	45	44	16	8
	8	13	5	168.6 s	210	2994	35	16
	9	15	6	-	-	-	-	-
4	9	14	5	0.75s	286	196	-	-
	10	16	5	-	-	-	-	-

Table 5.1: Computations of summation sets in odd characteristic

Timings in even characteristic In Table 5.2 we report computation times for the first summation sets for binary hyperelliptic curves of genus 2,3,4. This is done with Magma on the same processor. The base field was fixed as $\mathbb{F}_{2^{15}}$ and curves' coefficients were randomly chosen, considering the most general case. In genus 2, it has to be stated that the use of canonical forms speeds up the computation and leads to sparser sets, because less non-zero coefficients in the curve's equation means less monomials in the support of the parametrization of $\mathcal{V}_m$. The column headings in the table are the same as in the previous paragraph, and we used the same criterion to interrupt a lengthy computation.

As for Semaev's summation polynomials, computations are easier to finish in even characteristic, and the summations sets' elements are much sparser and fewer. It is known [Die11] that when $g = 1$ then $\deg V(S_m) = 2^{m-2}$ if S_m is the m-th Summation Polynomial. This fact added to the last column of Tables 5.1 and 5.2 leads to the conjecture that $\deg V(\mathbb{S}_m) = 2^{m-g-1}$. We give more details on this in the next section.

genus g	m	#vars	#elim	Time	#set	Avg. len.	Avg. deg.	$\deg V(\mathbb{S}_m)$
2	5	8	3	0.02 s	2	6	6	4
	6	10	4	0.010 s	4	32	15	8
	7	12	5	7.810 s	12	836	35	16
	8	14	6	1320 s	38	36914	71	32
	9	16	7	-	-	-	-	-
3	7	11	4	0.1 s	3	8	9	8
	8	13	5	0.2 s	9	46	20	16
	9	15	6	140.75 s	37	2401	40	32
	10	17	7	-	-	-	-	-
4	9	14	5	0.02 s	4	9	11	16
	10	16	6	0.3 s	16	60	24	32
	11	18	7	22682.87 s	81	6195	45	-
	12	20	8	-	-	-	-	-

Table 5.2: Computations of summation sets in even characteristic

5.4 Specialization of Summation Sets for Index-Calculus

In this section the discussion suits both elliptic and hyperelliptic curves so that we let $\mathscr{H}$ stand for both elliptic or hyperelliptic curves, specifying the genus as needed. We start by highlighting the projection of $\mathscr{V}_m$ to $V(\mathbb{S}_m)$ and analyze its property to give a general conjecture for $\deg \mathscr{V}_m$ and $\deg V(\mathbb{S}_m)$. Next we study two possible ways of describing PDP_m instances (Definition 3.11) using Summation Sets before the Weil Descent. Classically when $g = 1$, a summation polynomial is computed - or given as raw input - once and for all, then evaluated at every new R that we try to decompose. Alternatively it is possible to compute the summation polynomial already evaluated at $x(R)$. We refer to those approaches as *project-then-specialize* resp. *specialize-then-project*. This also prompts the definition of the *Specialized m-Summation Variety* associated to $R \in \mathrm{Jac}(\mathscr{H})$

$$\mathscr{V}_{m,R} = \{(P_1, \ldots, P_{m-g}) : \sum_{i=1}^{m-g} (P_i) = R\}.$$

A parametrization of this variety is given, as well as generators for its projection over the abscissae, that we call *Specialized Summation Polynomials*. A conjecture on the degree of $\mathscr{V}_{m,R}$ is also given depending on m, g. Then we analyze both approaches to observe that the degree of the ideal resulting from the Weil Descent after project-then-specialize is $2^{n(g-1)}$ times greater than its specialize-then-project counterpart, which is found to be equal to $d_{\mathrm{Nag}} = 2^{n(n-1)g}$. In particular this shows that project-then-specialized or specialized-then-project methods are equivalent when $g = 1$. A toy-example is presented in Section 5.4.4 to illustrate the theoretical discussion. As a conclusion to this section, we also give a negative answer to the question of a recursive computation for summation sets in the spirit of Proposition 3.13.

5.4.1 Summation Sets and Projections

For an imaginary hyperelliptic curve $\mathscr{H}$ and $P(x,y) \in \mathscr{H}$, we denote by $-P$ the image of P by the canonical (hyper)elliptic involution $[-]$. When $g = 1$, this also gives the opposite of P for the group law. It is well-known that hyperelliptic curves are degree 2 covers of the projective line, which means that there exists a projection $\pi : \mathscr{H} \longrightarrow \mathbb{P}^1$ of degree 2. Additionally it can be chosen invariant wrt. the canonical involution i.e. $\pi \circ [-] = \pi$. For example consider the projection $\pi(P) = [x(P) : 1], \pi(P_\infty) = [1 : 0]$ on the abscissae in a Weierstrass model. As $-P(x, -y + h_1(x))$, notice indeed that $\pi \circ [-] = \pi$. For all $m \in \mathbb{N}^*$ we also denote by $\pi : \mathscr{H}^m \longrightarrow (\mathbb{P}^1)^m$ the induced cover of degree 2^m.

Definition 5.7. *Let $\mathscr{H}$ be a (hyper)elliptic curve of genus $g \geq 1$ with $P \mapsto (P)$ its canonical embedding into $\mathrm{Jac}(\mathscr{H})$. For all $m \in \mathbb{N}^*$, the m-points sum is the map defined by*

$$\begin{array}{rccccc} \Sigma_m : & \mathscr{H}^m & \longrightarrow & \mathrm{Jac}(\mathscr{H})^m & \longrightarrow & \mathrm{Jac}(\mathscr{H}) \\ & (P_1,\ldots,P_m) & \longmapsto & ((P_1),\ldots,(P_m)) & \longmapsto & \sum_{i=1}^m (P_i) \end{array}$$

When $m \geq 2g+1$, the m-Summation Variety is $\mathscr{V}_m = \{(P_1,\ldots,P_m) : \sum_{i=1}^m (P_i) = \mathscr{O}\} = \Sigma_m^{-1}(\{\mathscr{O}\})$. With notations of Section 5.1, Proposition 5.6 tells us that $\pi(\mathscr{V}_m) = V(\mathscr{I}_m \cap \mathbb{F}[\mathbf{X}]) = V(\mathbb{S}_m)$ for any m-summation set $\mathbb{S}_m$. In particular, summation ideals depend on the choice[2] of the double cover π. Overall we have a commutative diagram

$$\begin{array}{ccc} \mathscr{V}_m & \hookrightarrow & \mathscr{H}^m \\ \downarrow{\scriptstyle \pi} & & \downarrow \\ V(\mathbb{S}_m) & \hookrightarrow & (\mathbb{P}^1)^m \end{array}$$

If $(x_1,\ldots,x_m) \in V(\mathbb{S}_m)$, from Proposition 5.6 there exist $P_i(x_i,y_i) \in \mathscr{H}$ such that $\sum_{i=1}^m (P_i) = \mathscr{O}$. Since $x(-P_i) = x(P_i)$, we also have $\sum_{i=1}^m (-P_i) = \mathscr{O}$. This means that $\pi : \mathscr{V}_m \longrightarrow V(\mathbb{S}_m)$ has degree at least 2, and it can be shown there are only two functions in $\mathbb{F}(\mathscr{H})$ that vanishes exactly at the $x(P_i)$'s, so $\deg \pi = 2$. If $[-] : \mathscr{H}^m \longrightarrow \mathscr{H}^m$ is induced by the canonical involution on $\mathscr{H}$, then $\mathscr{V}_m/[-] \simeq V(\mathbb{S}_m)$ as the projection π factors through this quotient in a degree 1 map. Using Table 5.1 and 5.2 as well as Diem's proof [Die11] for $g = 1$, we can formulate a conjecture on the degree of $\mathscr{V}_m$.

Conjecture 5.8. *Let $\mathscr{H}$ be an (hyper)elliptic curve of genus g, and $m \geq 2g+1$. The degree of the m-Summation variety $\mathscr{V}_m$ resp. its projection $\pi(\mathscr{V}_m) = V(\mathbb{S}_m)$ are*

$$\deg \mathscr{V}_m = 2^{m-g}, \ \deg V(\mathbb{S}_m) = 2^{m-g-1}.$$

This conjecture is strengthened by the following intuition: if we fix $\dim \mathscr{V}_m = m-g$ points in a vanishing sum $(P_1)+\ldots+(P_m) = \mathscr{O}$ of points of a genus g curve, then generally the last g points are totally determined. In other word we have $m-g$ degree of freedom for a vanishing sum of size m. Two choices can be done for each "free" point, as any point has an opposite, so there should be 2^{m-g} possible sums (provided the field is algebraically closed). As $\mathscr{V}_m/[-] \simeq V(\mathbb{S}_m)$ from the quotient of a degree 2 projection, then it is expected that $\deg V(\mathbb{S}_m) = \deg \mathscr{V}_m/2$.

5.4.2 Specialized Summations Sets

Let $\mathbf{X} = (X_1,\ldots,X_m)$ and $\overline{\mathbf{X}} = (X_1\ldots,X_{m-g})$. To solve PDP_m instances, we can on the one hand compute a m-th Summation set $\mathbb{S}_m \subset \mathbb{F}[\mathbf{X}]$ — associated to a well-chosen cover but we assume here it is given by the Weierstrass abscissae — then for a given input $R = (R_1) + \cdots + (R_g) \in \mathrm{Jac}(\mathscr{H})$ we compute the specialization $\mathbb{S}_m(\overline{\mathbf{X}}, x(R_1),\ldots,x(R_g))$. This process can be decomposed into two steps: first a projection is done to compute $V(\mathbb{S}_m)$. Then, this projection is cut by the hyperplanes $X_{m-g+1} - x(R_1),\ldots,X_m - x(R_g)$. As $\pi^{-1}(\pi(R)) = \{\pm R_1,\ldots,\pm R_g\}$, the preimage of this intersection wrt. π is

$$\overline{\mathscr{V}_{m,R}} := \{(P_1,\ldots,P_{m-g}) : \sum_{i=1}^{m-g} (P_i) = \pm(R_1) \pm \cdots \pm (R_g)\}.$$

This variety contains obviously more elements than those actually needed. Since R is given in practice, we can on the other hand compute (generators for the projection of) another variety.

[2] When $g = 1$, the authors of [FHJ+14] use that different covers can be obtained by action of $\mathrm{Aut}(\mathbb{P}^1) = \mathrm{PGL}_2$ to find a cover having a good behaviour wrt. the group of symmetry of the m-Summation variety and to compute Summation Polynomials associated to this cover.

Definition 5.9. *Let $\mathscr{H}$ be an (hyper)elliptic curve of genus g defined over $\mathbb{F}$, and fix $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. The R-Specialized m-Summation Variety, or the Specialized m-Summation Variety if the context is clear, is the algebraic variety defined by*

$$\mathscr{V}_{m,R} = \{(P_1,\ldots,P_{m-g}) : \sum_{i=1}^{m-g}(P_i) = R\}.$$

Remark 5.10. *Whenever a decomposition $R = (P_1)+\ldots+(P_{m-g})$ is found, then it also determines the "opposite" $-R = (-P_1)+\cdots+(-P_{m-g})$. This does not give an element of $\mathscr{V}_{m,R}$, rather an element of $\mathscr{V}' = \{(P_1,\ldots,P_{m-g}) : \sum_{i=1}^{m-g}(P_i) = \pm R\}$. If $[-] : \mathscr{H}^m \longrightarrow (\mathbb{P}^1)^m$ is induced by the canonical hyperelliptic involution on $\mathscr{H}$, then we check that $\mathscr{V}'/[-] \simeq \mathscr{V}_{m,R}$. For Index-Calculus this does not matter as we use the action of $[-]$ to reduce the size of the factor base anyway.*

Let R in $\mathrm{Jac}(\mathscr{H})$ with Mumford Representation (u,v). We look for points such that $R = (P_1)+\cdots+(P_{m-g})$, and we can generate a parametrization of $\mathscr{V}_{m,R}$ following the steps presented in Section 5.1 up to equation (5.4). The difference is that we consider $\mathscr{L}((m-g)P_\infty - R)$ and its natural basis $\{u, xu, \ldots, x^{d_1}u, y-v, x(y-v), \ldots, x^{d_2}(y-v)\}$. Recall that this space has dimension $m-2g+1$ and let $\mathbf{a} = (a_1,\ldots,a_{m-2g})$. Since the generic norm $N(f)$ for this sum should vanish at any root of u, we obtain a polynomial $F(x)$ in $(\mathbb{F}[\mathbf{a}])[x]$ as:

$$F(x) = \frac{N(f)}{u(x)} = x^{m-g} + \sum_{i=0}^{m-g-1} N_{m-g-i}(\mathbf{a})x^i. \tag{5.11}$$

We call this polynomial the Decomposition Polynomial associated with R, and it will be important in Section 6.2. If f describes a PDP_m instance, then F vanishes at the abscissae of the P_i's and we can write with $\mathbf{x} = (x(P_1),\ldots,x(P_m))$:

$$\begin{aligned} F(x) &= \prod_{i=1}^{m-g}(x - x(P_i)) \\ &= x^{m-g} + \sum_{i=0}^{m-g-1}(-1)^{m-g-i}e_{m-g-i}(\mathbf{x})x^i. \end{aligned} \tag{5.12}$$

If $\mathscr{I}_{m,R} \subset \mathbb{F}[\mathbf{a},\overline{\mathbf{X}}]$ is the ideal generated by equations $\{e_i(\overline{\mathbf{X}}) + (-1)^{m-g-i+1}N_i(\mathbf{a}) : 1 \le i \le m\}$ we obtain a parametrization of $\mathscr{V}_{m,R}$ by $\mathscr{I}_{m,R}$ by equating (5.11) and (5.12) — the proof is similar to that of Proposition 5.3 - and we can then extend Definition 5.4.

Definition 5.11 (Specialized Summation Polynomials)**.** *Let $\mathscr{H}$ be an (hyper)elliptic curve of genus g defined over $\mathbb{F}$, and fix $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. The m-th summation ideal specialized to R is defined as the elimination ideal $\mathscr{I}_{m,R} \cap \mathbb{F}[\overline{\mathbf{X}}]$ where $\mathscr{I}_{m,R}$ is the ideal in $\mathbb{F}[\mathbf{a},\overline{\mathbf{X}}]$ described as above. Any set $\mathbb{S}_{m,R} \subset \mathbb{F}[\overline{\mathbf{X}}]$ generating $\mathscr{I}_{m,R} \cap \mathbb{F}[\overline{\mathbf{X}}]$ is called a set of m-th summation polynomials specialized to R or a Specializeed Summation Set.*

Such summation sets can be computed by a Gröbner basis computation for an elimination order. An analysis similar as that of Section 5.1 can be done: $\mathscr{V}_{m,R}$ is generated by $m-g$ equations in a $2m-3g$-dimensional ambient space, therefore in general $\dim \mathscr{V}_{m,R} = m-2g$. The projection onto a subspace of dimension $m-g$ has generally codimension g, hence we expect $\mathscr{V}_{m,R}$ to be generated by at least g polynomials. The next proposition has a proof similar to that of Proposition 5.6.

Proposition 5.12. *Let $\mathscr{H}$ be an (hyper)elliptic curve of genus g defined over $\mathbb{F}$, and fix $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. For any $m \ge 2g+1$, a set $\mathbb{S}_{m,R}$ of m-th summation polynomials specialized to R and associated to $\mathscr{H}$ exists, and it satisfies:*

$$\begin{aligned} \mathbb{S}_{m,R}(\mathbf{x}) = 0 \Leftrightarrow\ & \exists\ y_1,\ldots,y_{m-g} \in \mathbb{F} \text{ such that } P_i(x_i,y_i) \in \mathscr{H}, 1 \leqslant i \leqslant m-g, \\ & \text{and } (P_1)+\cdots+(P_{m-g}) = R. \end{aligned}$$

As in the end of Section 5.1, using variables for symmetric functions in the parametrization of $\mathscr{V}_{m,R}$ leads to consider the action of the symmetric group. We again keep the same notations for the Specialized variety quotiented by the symmetric action. Since it is given by a polynomial parametrization, it is irreducible and the defining equations generate a radical ideal by Corollary 1.72. Recall from Remark 5.10 that $\mathscr{V}' = \{(P_1,\dots,P_{m-g}) : \sum_{i=1}^{m-g}(P_i) = \pm R\}$. The projection $\pi : \mathscr{V}' \longrightarrow V(\mathbb{S}_{m,R})$ has degree 2, and factors through the quotient $\mathscr{V}'/[-] \simeq \mathscr{V}_{m,R}$ as a map of degree 1. In particular, $\mathscr{V}_{m,R}$ is birational to $V(\mathbb{S}_{m,R})$.

Timings for Specialized Summation Sets: Below are tables summing up computations experiments for several Specialized Summation Sets with $2 \leq g \leq 4$ in both even and odd characteristics. Protocol and column headings are identical to the previous tables. The degree of $V(\mathbb{S}_{m,R})$ can also be obtained as the degree of $\mathscr{V}_{m,R}$ since both varieties are birational. This is done when a Specialized Summation Set cannot be computed in reasonable time.

genus g	m	#vars	Time	#set	Avg. len.	Avg. deg.	$\deg V(\mathbb{S}_{m,R})$
2	5	4	0.000s	2	5	4	2
	6	6	0.000s	7	28	10	4
	7	8	0.18s	13	248	21	8
	8	10	3505s	130	5901	50	16
3	7	5	0.000s	3	5	4	2
	8	7	0.000s	6	16	8	4
	9	9	0.22s	45	159	19	8
	10	11	54.3s	194	2028	36	16
	11	13	-	-	-	-	32
4	9	6	0.00s	4	5	4	2
	10	8	0.00s	7	15	8	4
	11	10	0.03s	24	80	15	8
	12	12	-	-	-	-	16

Table 5.3: Computations of Specialized Summations Sets in odd characteristic

genus g	m	#vars	Time	#set	Avg. len.	Avg. deg.	$\deg V(\mathbb{S}_{m,R})$
2	5	4	0.000s	2	5	4	2
	6	6	0.000s	3	14	8	4
	7	8	0.03s	5	89	17	8
	8	10	12.7s	15	1032	36	16
	9	12	-	-	-	-	-
3	7	5	0.000s	3	4	4	2
	8	7	0.000s	4	12	7	4
	9	9	0.1s	6	46	13	8
	10	11	0.89s	14	276	23	16
	11	13	-	-	-	-	32
4	9	6	0.00s	4	4	4	2
	10	8	0.00s	5	11	7	4
	11	10	0.01s	7	40	12	8
	12	12	0.3s	12	127	19	16
	13	14	-	-	-	-	-

Table 5.4: Computation of Specialized Summation Sets in even characteristic

This prompts the following conjecture on the degree of $\mathscr{V}_{m,R}$. As previously, the degree is expressed taking into account the action of the $m-g$-th symmetric group.

Conjecture 5.13. *Let $\mathscr{H}$ be an (hyper)elliptic curve of genus g, $R \in \mathrm{Jac}(\mathscr{H})$ of weight g and $m \geq 2g+1$. The degree of the m-Specialized Summation Variety $\mathscr{V}_{m,R}$ and its projection $V(\mathbb{S}_{m,R})$ are:*

$$\deg \mathscr{V}_{m,R} = \deg V(\mathbb{S}_{m,R}) = 2^{m-2g}.$$

Let $H = V(\langle X_{m-g+i} - x(R_i), 1 \leq i \leq g \rangle)$ to obtain the following commutative diagram

$$\begin{array}{ccccc} \mathscr{V}_{m,R} & \hookrightarrow & \overline{\mathscr{V}_{m,R}} & \hookrightarrow & \mathscr{V}_m \\ \downarrow \pi & & \downarrow \pi & & \downarrow \pi \\ V(\mathbb{S}_{m,R}) & \hookrightarrow & V(\mathbb{S}_m) \cap H & \hookrightarrow & V(\mathbb{S}_m) \end{array}$$

When $g = 1$, let S_m resp. $S_{m,R}$ be the m-th summation polynomial resp. the specialized m-th polynomial. In this case it is checked that $\mathscr{V}' = \overline{\mathscr{V}_{m,R}}$. Since $\pi : \overline{\mathscr{V}_{m,R}} \longrightarrow V(S_m) \cap H$ has degree 2 and that it factors through $\overline{\mathscr{V}_{m,R}}/[-] \simeq \mathscr{V}_{m,R}$, then $\mathscr{V}_{m,R}$ is birational to $V(S_m) \cap H$. This also means that $S_{m,R}(X_1, \ldots, X_{m-1}) = S_m(X_1, \ldots, X_{m-1}, x(R))$, and the specialized variety is indeed generated by the evaluation of S_m at $x(R)$.

If $g \geq 2$ a specialize-then-project approach means finding points in $V(\mathbb{S}_{m,R})$, while project-then-specialize accounts for finding points in $V(\mathbb{S}_m) \cap H$, which contains more solutions than needed. More precisely, observe that $\pi^{-1}(\pi(R)) = \{\pm R_1, \ldots, \pm R_g\}$ contains 2^g elements but that we are only interested in the two $\{\pm(R_1 + \cdots + R_g)\}$: this means that we have 2^{g-1} times too much solutions in $V(\mathbb{S}_m) \cap H$. In other words, we have generally

$$\deg(V(\mathbb{S}_m) \cap H) = 2^{g-1} \cdot \deg V(\mathbb{S}_{m,R}). \tag{5.13}$$

5.4.3 Specialize-then-Project better than Project-then-Specialize

Focusing on Index-Calculus purpose, we want to solve PDP_{ng} instances, i.e. find decomposition of a given $R \in \mathrm{Jac}(\mathscr{H})$ of weight g as

$$R = (P_1) + \cdots + (P_{ng}), P_i \in \mathscr{B}$$

with $\mathscr{B} = \{(P) : P \in \mathscr{H}, x(P) \in \mathbb{F}_q\}$, so we let $m = (n+1)g$.

Project-then-Specialize: Let $\mathbb{S}_m \subset \mathbb{F}_{q^n}[e_1, \ldots, e_m]$ be a m-th Summation Set for $\mathscr{H}$, and (u, v) be the Mumford Representation of R. We now describe how to specialize it at R, omitting explicit formulae for the sake of simplicity; we refer the reader to Section 5.4.4 for an example with small parameters. If $u = x^g + \sum_{i=0}^{g-1} u_{g-i} x^i$ with $u_i \in \mathbb{F}_{q^n}$, we desymmetrize $\mathbb{S}_m$ using $m - g = ng$ new variables $E_1, \ldots, E_{ng}$ for symmetric functions in ng variables and the u_i's. Algebraically it amounts to let $\mathbb{S}_m$ be in $\mathbb{F}_{q^n}[E_1, \ldots, E_{ng}, e_1, \ldots, e_m]$ and add the m linear desymmetrization relations to the generators of $\mathbb{S}_m$. Geometrically we now have $\dim\ \mathbb{S}_m = 2ng$, so that the intersection with the m hyperplanes $H = H_1 \cap \ldots \cap H_m$ given by the desymmetrization relations has in general dimension

$$\dim(V(\mathbb{S}_m) \cap H) = (n-1)g$$

A generating set $\overline{\mathbb{S}_m}$ for $V(\mathbb{S}_m) \cap H$ is then obtained by evaluation of all $S \in \mathbb{S}_m$ at the desymmetrization relations.

Using notations from the end of Section 5.4.1, the next step is to build (a generating set of) the Weil Restriction $\mathscr{W}_n(\overline{\mathbb{S}_m})$. Let $\mathbb{F}_{q^n} = \mathbb{F}_q[t]/\langle P(t) \rangle$ and write $E_i = \sum_{i=0}^{n-1} E_{i,j} t^j$ for $1 \leq i \leq ng$. Let $\mathbf{E} = (E_{1,0}, \ldots, E_{1,n-1}, \ldots, E_{ng,0}, \ldots, E_{ng,n-1})$. We can collect any $S \in \overline{\mathbb{S}_m}$ in t, writing

$$S(E_1, \ldots, E_{ng}) = \sum_{i=0}^{n-1} S_{i,j}(\mathbf{E}) t^i, \tag{5.14}$$

hence $\mathscr{W}_n(\overline{\mathbb{S}_m})$ is generated by $\{S_{i,j}(\mathbf{E}) : 1 \le i \le ng, 0 \le i \le n-1\}$. To find a decomposition of R, it is necessary that values for E_i belong to $\mathbb{F}_q$ - equivalently, that $E_{i,j} = 0$ for $1 \le i \le ng, 1 \le j \le n-1$. Geometrically, this means we try to find points in the variety

$$\mathbf{V} = \mathscr{W}_n(\overline{\mathbb{S}_m}) \cap V(\{E_{i,j} : 1 \le i \le ng, 1 \le j \le n-1\}).$$

Since $\dim \mathscr{W}_n(\overline{\mathbb{S}_m}) = n(n-1)g$, the above intersection has dimension 0 in general. Assuming Conjectures 5.8 and 5.13 are true, we have in general $\deg(V(\mathbb{S}_m) \cap H) = \deg V(\mathbb{S}_m) = 2^{m-g-1}$ from Expression (5.13) and thus

$$\deg \mathbf{V} = (\deg V(\mathbb{S}_m))^n = 2^{n\cdot(m-g-1)} = 2^{n(ng-1)}.$$

Recall that the degree obtained with a (unrefined) Nagao approach is $d_{\text{Nag}} = 2^{n(n-1)g}$, so that $\deg \mathbf{V} = 2^{n(g-1)} \cdot d_{\text{Nag}}$.

Specialize-then-Project: Now let $\mathbb{S}_{m,R} \subset \mathbb{F}_{q^n}[E_1, \ldots, E_{ng}]$ be a Specialized Summation Set for $\mathscr{H}$. In this situation we have $\dim V(\mathbb{S}_{m,R}) = m - 2g = (n-1)g$, and therefore $\dim \mathscr{W}_n(\mathbb{S}_{m,R}) = n(n-1)g$. Collecting wrt. t any $S \in \mathbb{S}_{m,R}$ as in expression (5.14), a generating set for $\mathscr{W}_n(\mathbb{S}_{m,R})$ is $\{S_{i,j}(\mathbf{E}) : 1 \le i \le ng, 0 \le i \le n-1\}$. Intersecting

$$\mathbf{V}_R = \mathscr{W}_n(\mathbb{S}_{m,R}) \cap V(\{E_{i,j} : 1 \le i \le ng, 1 \le j \le n-1\}),$$

with the $n(n-1)g$ hyperplanes $E_{i,j} = 0, 1 \le i \le ng, 1 \le j \le n-1$ generally reduces the dimension to $\dim \mathbf{V}_R = 0$. Assuming Conjecture 5.13 is true, we find that

$$\deg \mathbf{V}_R = (\deg V(\mathbb{S}_{m,R}))^n = 2^{n\cdot(m-2g)} = 2^{n(n-1)g} = d_{\text{Nag}}.$$

The whole description confirms that when $g = 1$, i.e. for elliptic curves, it makes no difference in terms of expected degree to chose one approach or another. Overall as soon as $g \ge 2$, only Specialized Summation Sets should be used to expect lower degree for the ideals resulting from the Weil Descent. In Chapter 6, Section 6.3, we give timing comparisons between the Specialize-then-Project approach and Nagao's modelling.

5.4.4 A toy-example and a new algorithm.

We now illustrate the two approaches with small parameters $n = g = 2$, and solve a PDP_4 instance over $k = \mathbb{F}_{1031}$ and $K = k[t]/(t^2 + 728t + 1005)$. We consider the genus 2 curve

$$\mathscr{H} : y^2 = x^5 + (876t + 276)x^3 + (459t + 27)x^2 + (141t + 664)x + 383t + 69.$$

In this setting the factor base is $\mathscr{B} = \{(P) : P \in \mathscr{H}, x(P) \in k\}$. Fix $R(u = x^2 + (338t + 756)x + 166t + 804, v = (179t + 133)x + 990t + 598)$. The goal is to find if possible $P_1, \ldots, P_4 \in \mathscr{B}$ such that

$$R = (P_1) + \cdots + (P_4).$$

This means we consider general cubics, and we can express them as element of $\mathscr{L}(6P_\infty)$ resp. $\mathscr{L}(4P_\infty - R)$ as

$$\begin{aligned} f(x,y) &= x^3 + a_2x^2 + a_1x + a_0 + a_3y \\ &= (\overline{a}_0 + x)u(x) + \overline{a}_1(y - v(x)). \end{aligned}$$

We let $\mathbf{a} = (a_0, \ldots, a_3)$ and $\overline{\mathbf{a}} = (\overline{a}_0, \overline{a}_1)$ so that the generic norm resp. the R-Decomposition polynomial can be written as

$$N(f) = x^6 + \sum_{i=0}^{5} N_{6-i}(\mathbf{a})x^i, \tag{5.15}$$

$$F(x) = \frac{N(f)}{u(x)} = x^4 + \sum_{i=0}^{3} \overline{N}_{4-i}(\overline{\mathbf{a}})x^i \tag{5.16}$$

We start by the Project-then-Specialized approach. A (symmetrized) 6-th Summation Set $\mathbb{S}_6 \subset K[e_1,\ldots,e_6]$ for $\mathscr{H}$ is computed, using parametrization (5.7) and expression (5.15). In DRL order we find 14 polynomials of average length 326 monomials; hence we cannot display them on paper. The next step is to evaluate this system at the coordinates given by u. With new variables $\mathbf{E} = (E_1,\ldots,E_4)$ representing the symmetric functions in 4 variables and $u(x) = x^2 - u_1 x + u_2$, the desymmetrization relations are obtained by equating coefficients in

$$(x^2 - u_1 x + u_2) \cdot (x^4 + \sum_{i=0}^{3} E_{4-i} x^i) = x^6 + \sum_{i=1}^{5} S_{6-i}(\mathbf{E}) x^i.$$

Then desymmetrizing $\mathbb{S}_6$ is done by computing $\overline{\mathbb{S}_6} = \mathbb{S}_6(S_1,\ldots,S_6)$. We now apply the Weil Descent to get a multivariate system of 28 equations in 4 variables in $k[\mathbf{E}]$. A Gröbner basis in lex order with e.g. $E_1 < \ldots < E_4$ is found in Shape Position as

$$\begin{cases} \overline{T_1}(E_1) \\ E_2 + \overline{T_2}(E_1), \\ E_3 + \overline{T_3}(E_1), \\ E_4 + \overline{T_4}(E_1). \end{cases}$$

with $\deg \overline{T_1} = 64$ and $\deg \overline{T_i} = 63, 2 \leq i \leq 4$. The degree is $2^{n(g-1)} = 4$ times larger than the expected $d_{\text{Nag}} = 2^{n(n-1)g} = 16$. Solving this system is now a matter of factoring $\overline{T_1}$ over k: we find the two solutions $(923,399,659,612)$ and $(656,323,124,790)$. To decide if the given PDP_4 instance itself has a solution, we factor the two polynomials

$$\begin{aligned} F_1(x) &= x^4 - 923x^3 + 399x^2 - 659x + 612, \\ F_2(x) &= x^4 - 656x^3 + 323x^3 - 124x + 790 \end{aligned}$$

over k; in this case we find that F_1 has only one root over k, but that $F_2 = (x-170)(x-594)(x-956)(x-998)$. This leads us to find the relation

$$R = (170,257t+744) + (594,969t+711) + (956,654t+140) + (998,55t+32).$$

We now use specialized summations sets for this R. The modelling described at the end of Section 5.1 along with expression (5.16) gives the following parametrization for $\mathscr{V}_{6,R}$:

$$\begin{cases} E_4 = (166t+804)\overline{a}_0^2 + (82t+866)\overline{a}_0\overline{a}_1 + (555t+455)\overline{a}_1^2, \\ -E_3 = (338t+756)\overline{a}_0^2 + (673t+765)\overline{a}_0\overline{a}_1 + (534t+133)\overline{a}_1^2 + (332t+577)\overline{a}_0 + (82t+866)\overline{a}_1, \\ E_2 = \overline{a}_0^2 + (338t+756)\overline{a}_1^2 + (676t+481)\overline{a}_0 + (673t+765)\overline{a}_1 + 166t + 804, \\ -E_1 = -\overline{a}_1^2 + 2\overline{a}_0 + 338t + 756. \end{cases}$$

A (symmetrized) Specialized Summation Set $\mathbb{S}_{6,R} \subset K[\mathbf{E}]$ is computed in DRL order by elimination of $\overline{\mathbf{a}}$. It contains 7 polynomials of average length 28, so that a Weil Descent gives a system of 14 equations in 4 variables over k. The reduced Gröbner basis of this system for lex order $E_1 < \ldots < E_4$ is:

$$\begin{cases} T_1(E_1) = E_1^{16} + 475E_1^{15} + 617E_1^{14} + 317E_1^{13} + 299E_1^{12} + 646E_1^{11} + 492E_1^{10} + 275E_1^9 + 680E_1^8 + 256E_1^7 + \\ 906E_1^6 + 15E_1^5 + 831E_1^4 + 954E_1^3 + 357E_1^2 + 623E_1 + 268, \\ E_2 + T_2(E_1), \\ E_3 + T_3(E_1), \\ E_4 + T_4(E_1), \end{cases}$$

again in Shape Position with $\deg T_1 = 16 = d_{\text{Nag}}$ and $\deg T_i = 15, 2 \leq i \leq 4$. We observe that T_1 divides $\overline{T_1}$, find $(656,323,124,790)$ as only solution this time and recover the previous decomposition.

5.4.5 Obstruction for a recursive computation of summations sets

The recursive approach for computing summation polynomials for a genus 1 curve E is found by decomposing a sum into two smaller sums:

$$P_1+\cdots+P_m=\mathscr{O} \Leftrightarrow \ \forall\ k\in\{2,\ldots,m-3\},\ \exists\ Q\in E(\overline{\mathbb{F}}):\begin{cases}P_1+\cdots+P_k=Q\\ P_{k+1}+\cdots+P_m=-Q\end{cases}$$

Using X as an indeterminate for the abscissae of the intermediate summand Q and $x_i=x(P_i)$, we deduce that $S_{k+1}(x_1,\ldots,x_k,X)$ and $S_{m-k+1}(x_{k+1},\ldots,x_m,X)$ have a common root. Hence their resultant with respect to X must vanish. If we see S_k and S_{m-k+1} in $\mathbb{F}[X_1,\ldots,X_m,X]$, then geometrically this corresponds to the projection of $V(S_k(X_1,\ldots,X_k,X))\cap V(S_{m-k+1}(X_{k+1},\ldots,X_m,X))$ on the m first coordinates. In general, both varieties are hypersurfaces in a $n+1$-dimensional space. Thus their intersection has dimension $n-1$. The projection on a n-dimensional subspace is then of codimension 1 and its defining ideal is indeed generated by the resultant with respect to X of both summation polynomials. This leads to Proposition 3.13.

However this observation cannot be generalized in higher genus to obtain a recursive method of computation. For the sake of simplicity we restrict ourselves to a genus 2 curve $\mathscr{H}$. Let be $\mathbb{S}_m$ a set of m-th summation polynomials associated to $\mathscr{H}$. Generally, a sum of points reduces to an element of weight 2 in $\mathrm{Jac}(\mathscr{H})$. Therefore if we split a sum of length m into two smaller decompositions, not one but two points have to be fixed in the intermediate summand $Q=(Q_1)+(Q_2)$. More explicitely assume that we have

$$\begin{cases}\mathbb{S}_{k+2}(x_1,\ldots,x_k,x(Q_1),x(Q_2))=0\\ \mathbb{S}_{m-(k+2)}(x_{k+1},\ldots,x_m,x(Q_1),x(Q_2))=0\end{cases}$$

for some $x_1,\ldots,x_m\in\mathbb{F}$. To any pair of distinct points $Q_1,Q_2\in\mathscr{H}$ correspond $2^g=4$ divisors: $Q=(Q_1)+(Q_2)$ and $-Q$, as well as $\widetilde{Q}=(Q_1)+(-Q_2)$ and $-\widetilde{Q}$. By Proposition 5.6 there exist $P_1,\ldots,P_m\in\mathscr{H}$ with $x(P_i)=x_i$ such that

$$\begin{cases}(P_1)+\cdots+(P_k)=Q \text{ or } \tilde{Q}\\ (P_{k+1})+\cdots+(P_m)=-Q \text{ or } -\tilde{Q}.\end{cases}$$

and so the decomposition into two small sums leads to the following "weaker" possibilities:

$$\begin{cases}(P_1)+\cdots+(P_m)=\mathscr{O}\\ \qquad\text{or}\\ \exists\ i\in\{1,2\}:(P_1)+\ldots+(P_m)=2(Q_i).\end{cases}$$

This means that the projection of $V(\mathbb{S}_k)\cap V(\mathbb{S}_{n-(k+2)})$ describes more than the vanishing sums of n points. As for the end of Section 5.4.2 with $\pi:\mathscr{H}^m\longrightarrow(\mathbb{P}^1)^m$ induced by the projection over the abscissae, the obstruction comes from $\#\pi^{-1}(\pi(Q))/\#\{\pm Q\}=2^{g-1}$. Consequently, there is little hope to achieve the same kind of equivalence as in the elliptic case using this approach. Still, there are several ways to model the situation as an elimination problem. Because of the above observation and the end of Section 5.1, the computation asks for the elimination of at least g variables between two *sets* of polynomials, which seems harder to do than a resultant between two polynomials. Computations indeed proved to be intractable in odd characteristic, even for the simplest case. In even characteristic, a first set of polynomials for sums of size m could be computed this way, with an already longer computational time than with the direct method. However this set of polynomials indeed vanished on sums of length m equals to $\mathscr{O}$ as well as on sums of length m equals to the double of a point.

Chapter 6

Degree Reduction in even characteristic

The results of this Chapter are part of an article which has been submitted at Design, Codes and Cryptography journal.

Let $\mathscr{H}$ be a hyperelliptic curve of genus g with equation $y^2 + h_1(x)y = h_0(x)$ and $R \in \mathrm{Jac}(\mathscr{H})$. Decompositions of R can be found by using either Nagao's approach (3.4.4) or a Summation modelling (5.4.2) to obtain a polynomial system describing the decomposition. In this Chapter, we exploit the link between the degree of the equations in both approaches and the even characteristic to reduce the number of solutions of those systems before solving them. This is a crucial step to make the computations tractable. In practice, the solving process is therefore much faster. The Chapter is divided into three parts, and we always work in even characteristic unless stated otherwise.

In Section 6.1 we focus on Nagao's modelling. The systems arising in this approach are generated by the (polynomial) coefficients of the *Decomposition Polynomial* (Definition 6.2). The shape of those coefficients is particular in even characteristic. Indeed, we show that one of them is always a univariate polynomial, and that some others are squares. Additional squares can be found when the leading coefficient of h_1 belongs to a subfield. All those properties are used in Section 6.1.2 to reduce the number of solutions of the system describing a decomposition when $\mathscr{H}$ is defined over $\mathbb{F}_{2^{dn}}$. We analyze the reduction factor and give (tight) bounds on the final degree. Using the classification from [BD04, CJ03] for genus 2 binary hyperelliptic curves, Table 6.1 sums up the degree reduction expected for $2 \leq n \leq 4$.

Section 6.2 deals with degree reductions for Summation modelling. Coefficients of the Decomposition Polynomial are involved in the parametrization of the Summation Variety. In even characteristic, their properties give the parametrization a special shape. This is a particular case of the action of the Frobenius automorphism over the ideal of relations of polynomial equations, which expresses by natural weights on some variables. We use this more general context in Section 6.2.1 to obtain general estimates for the degree of the ideal of relations. Next we instantiate the situation to a PDP_{ng} instance over $\mathscr{H}$ and analyze the reduction factor. The situation is however not as clear as in Nagao's modelling as some degrees are harder to estimate. We give a complete analysis in genus 2 and observe that the reduction factor is maximal and equal for Type I_b curve with $h_1(x) = x^2$ and Type II curve.

The last part of the Chapter presents comparisons between modellings. The case of odd characteristic is briefly presented. Next, in even characteristic, we compare running time using a Magma 2.19 implementation of our reduction for Type I_b curve with $h_1(x) = x^2$ and Type II curve defined over $\mathbb{F}_{2^{dn}}$, with $d = 15$ and $n = 3$. The times we obtain with our Magma

code prompt a simulation of Index-Calculus for realistic parameters. Using a dedicated implementation mixing code generating techniques and efficient Gröbner Basis libraries, we estimate in Section 6.3.3 the running time for a complete harvesting phase over a Type *II* curve defined over $\mathbb{F}_{2^{93}}$, whose Jacobian variety achieves a generic security bound of 2^{92} operations. A bit more than a week is needed to start linear algebra, which emphasizes that such curves are weaker than expected.

6.1 Reducing degree of ideals in Nagao's approach

Nagao's modelling (Section 3.4.4) can be used to solve the PDP_{ng} instance related to R. Such instances are described by a system generated by the (polynomial) coefficients of the *Decomposition Polynomial* (Definition 6.2). One of them is always a univariate polynomal, that is used to determine values for up to $n-1$ variables of the system, achieving a reduction factor of 2^{n-1}. The shape of those coefficients is particular and can be used to reduce the degree of the systems to solve. Some others are squares: the number of squares among the coefficients is linked to the *length* L_h (Definition 6.6) of the polynomial h_1 in the curve's equation. Indeed, we show that exactly $g+1-L_h$ coefficients are squares. Only $g-L_h$ squares become relevant for the degree reduction, as one counts the fact that the Decomposition Polynomial is monic. Any squared coefficient can be replaced by its square root: since the square root is now a linear polynomial, the degree is in general divided by 2 with each replacement. The estimated degree reduction thus depends on L_h: after the Weil Descent is done on the Decomposition Polynomial's coefficients, $(n-1)(g-L_h)$ square equations are replaced by linear ones (Proposition 6.9). Adding the $n-1$ variables that we can determine using the univariate coefficient N_1, we obtain a general reduced degree of

$$\begin{aligned} d_{\mathrm{Red}} &= \frac{d_{\mathrm{Nag}}}{2^{(n-1)(g-L_h+1)}} \\ &= 2^{(n-1)((n-1)g+L_h-1)}. \end{aligned}$$

Lower and upper bounds are respectively reached for curves with $L_h = 0$ or $L_h = g$, and give a lower bound of $d_{\mathrm{opt}} = 2^{(n-1)((n-1)g-1)}$ for the degree reduction process. As the expected degree in a classical Nagao modelling is $d_{\mathrm{Nag}} = 2^{n(n-1)g}$, the degree reduction factor is at most

$$\frac{d_{\mathrm{Nag}}}{d_{\mathrm{opt}}} = 2^{(n-1)(g+1)}.$$

This is true for binary elliptic curve as well, setting $g = 1$. Each step of the reduction process is illustrated on a toy example. The Section concludes with an exhaustive analysis of genus 2 binary curves. Such curves are classified into three types — see Section 6.1.3 for details — and have been proposed as new standards for cryptographic primitives [BD04]. We sum up all the expected degrees for the ideals in Table 6.1. We notice that, while Type II curves were particularly suggested in [BD04] because of their lower cost arithmetic, they are as weak as Supersingular curves[1] (type *III*) against Decomposition attacks, if the base field can be written as an extension of degree less than 4. This suggests additional care in the selection of parameters for new cryptographic standards.

6.1.1 Properties of Decomposition polynomials' coefficients

Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be an hyperelliptic curve of genus g defined over a field $\mathbb{F}$. The general goal is to find decompositions $(P_1), \ldots, (P_m)$ of a fixed $R = (u, v) \in \mathrm{Jac}(\mathscr{H})$ of weight g. As described in the Section 3.4.4, it is equivalent to find $f \in \mathscr{L}(mP_\infty - R)$ such that $\mathrm{div}\, f = \sum_{i=1}^{m}(P_i) - R$. The modelling uses the coefficients of a particular polynomial (3.5) related to R. We generally define this polynomial in Definition 6.2 as the *Decomposition Polynomial*. Then we give expressions for it using coordinates $(a_0, \ldots, a_d)$ for a basis of $\mathscr{L}(mP_\infty - R)$. These expressions are used to understand the shape of its coefficients $N_i \in \mathbb{F}[a_0, \ldots, a_d]$ in even characteristic, and therefore to understand the system describing a decomposition of R.

Fix $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. Riemann-Roch spaces as $\mathscr{L}(mP_\infty - R)$ for $m \in \mathbb{N}$ are of particular interest in our situation. If (u, v) is the Mumford representation of R with $\deg u = g$

[1] Those curves are known to be weak to the Frey-Rück attack [FMR94], and thus are not considered safe as basis for Discrete Logarithms Problems.

then a natural basis of $\mathscr{L}(mP_\infty - R)$ is given by

$$\{u, xu, \ldots, x^{d_1}u, y-v, x(y-v), \ldots, x^{d_2}(y-v)\}, \tag{6.1}$$

with $d_1 = \lfloor (m-g)/2 \rfloor$ and $d_2 = \lfloor (m-g-1)/2 \rfloor$, so that its dimension is $d+1$, with $d = m-g$. Thus any $f \in \mathscr{L}(mP_\infty - R)$ can be written as

$$f(x,y) = u(x)\left(\sum_{i=0}^{d_1} a_i x^i\right) + (y - v(x))\left(\sum_{i=0}^{d_2} a_{i+d_1+1} x^i\right) \tag{6.2}$$

and we can take $a_{d_1} = 1$ or $a_d = 1$ by normalizing f at infinity.

Definition 6.1. *Let $d = \dim \mathscr{L}(mP_\infty - R) - 1$, for $R = (u,v) \in \mathrm{Jac}(\mathscr{H})$ of weight g.*

- *The polynomial $f(x,y)$ in $(\mathbb{F}[a_0, \ldots, a_d])[x,y]$ in expression (6.2) is called a generic function in $\mathscr{L}(mP_\infty - R)$.*
- *The generic norm of a generic function is the polynomial $N(f) = Res_y(f(x,y), y^2 + h_1(x)y - h_0(x))$ in $(\mathbb{F}[a_0, \ldots, a_d])[x]$.*

We consider now a sum like $R = (P_1) + \cdots + (P_m)$ and assume that $\mathrm{div}\, f = \sum_{i=1}^m (P_i) - R$. The abscissae of the $m+g$ points not at infinity are roots of $N(f)$. In particular u divides $N(f)$ and we obtain this way a polynomial in $(\mathbb{F}[a_0, \ldots, a_d])[x]$:

$$\frac{N(f)}{u(x)} = F(x) = x^m + \sum_{i=0}^{m-1} N_{m-i}(a_0, \ldots, a_{d-1}) x^i, \tag{6.3}$$

with $N_i \in \mathbb{F}[a_0, \ldots, a_d]$ and $\deg N_i = 2$ for all i. Notice that F vanishes exactly at the abscissae of the P_i's, prompting the next definition.

Definition 6.2. *For $R \in \mathrm{Jac}(\mathscr{H})$ of weight g, let f be a generic function in $\mathscr{L}(mP_\infty - R)$. The polynomial $F(x) = \frac{N(f)}{u(x)} \in (\mathbb{F}[a_0, \ldots, a_d])[x]$ is called the R-Decomposition polynomial, or the Decomposition Polynomial if the context is clear.*

This polynomial has already been seen in Section 5.4 when we defined Specialized Summation Ideals.

Convenient expressions for Decomposition polynomials Using the natural basis (6.1) of $\mathscr{L}(mP_\infty - R)$ with $d_1 = \lfloor (m-g)/2 \rfloor, d_2 = \lfloor (m-g-1)/2 \rfloor$ and $d = m-g$, set $p(x) = \sum_{i=0}^{d_1} a_i x^i$ and $q(x) = \sum_{i=0}^{d_2} a_{i+d_1+1} x^i$ in order to write (6.2) as

$$f(x,y) = u(x)p(x) + (y - v(x))q(x).$$

We normalize f at infinity if needed so that its norm is a monic polynomial in $(\mathbb{F}[a_0, \ldots, a_d])[x]$ and of degree $m+g$ given by:

$$\begin{aligned} N(f) &= (vq - up)^2 + q(vq - up)h_1 - q^2 h_0 \\ &= (up)^2 - 2upvq - upqh_1 + q^2(v^2 + vh_1 - h_0) \\ &= u(up^2 - pq(2v + h_1) + q^2 w), \end{aligned}$$

where $u, h_1, w \in \mathbb{F}[x]$ and $-w$ is a monic polynomial of degree $g+1$ given by the third property of the Mumford Representation. Hence the R-Decomposition polynomial has the following general expression in $(\mathbb{F}[a_0, \ldots, a_d])[x]$:

$$F(x) = up^2 - pq(2v + h_1) + q^2 w = x^m + \sum_{i=0}^{m-1} N_{m-i}(a_0, \ldots, a_d) x^i, \tag{6.4}$$

with a_{d_1} or $a_d = 1$ depending on the parity of m. We now assume that $\mathbb{F}$ has even characteristic, unless stated otherwise.

Coefficient N_1 is univariate The next proposition generalizes that of [JV13] to every genus.

Proposition 6.3. *Let $\mathbb{F}$ be a field of even characteristic and $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ a hyperelliptic curve of genus g defined over $\mathbb{F}$. Write $h_1(x) = \sum_{i=0}^{g} H_i x^i$ with $H_i = 0$ for $i > \deg h_1$. Fix $R = (u,v) \in \mathrm{Jac}(\mathscr{H})$ with $u(x) = x^g + u_1 x^{g-1} + \cdots \in \mathbb{F}[x]$, and with R-Decomposition polynomial F written as in expression (6.4). The coefficient $N_1(a_0,\ldots,a_d)$ is always an univariate polynomial. More precisely, we have:*

$$N_1(a_0,\ldots,a_d) = N_1(a_{d_1}) = a_{d_1}^2 + H_g a_{d_1} + u_1 \text{ if } m \text{ is odd},$$
$$N_1(a_0,\ldots,a_d) = N_1(a_d) = a_d^2 + H_g a_d + u_1 \text{ if } m \text{ is even}.$$

Proof. Since arguments are similar for the even and odd cases, we restrict ourselves to the even case. In particular p is monic as a polynomial in x, and so is up^2 in expression (6.4). We also have

$$\deg_x up^2 = m, \ \deg_x q^2 w = m-1 \text{ and } \deg_x pqh_1 = m - g - 1 + \deg h_1.$$

Because of the characteristic, p^2 does not have a term of degree $m-1$ in x, so the leading coefficient in x of $up^2 - x^m$ is $\mathrm{LC}_x(up^2 - x^m) = u_1$. We conclude as $\mathrm{LC}_x(qw^2) = a_d^2$ and $\mathrm{LC}_x(pqh_1) = H_g a_d$. □

Assume now that m is even, so that $a_{d_1} = 1$. If $H_g \in \mathbb{F}_{2^k}$, which is generally the case as h_1 is monic in practice, then Proposition 6.3 gives:

$$\begin{aligned} N_1(a_d) &= a_d^2 + H_g a_d + u_1 \\ &= \left(\sum_{i=0}^{n-1} a_{d,i} t^i\right)^2 + H_g \sum_{i=0}^{n-1} a_{d,i} t^i + \sum_{i=0}^{n-1} u_{1,i} t^i \\ &= \sum_{i=0}^{n-1} a_{d,i}^2 t^{2i} + H_g a_{d,0} + \sum_{i=1}^{n-1} a_{d,i} t^i + \sum_{i=0}^{n-1} u_{1,i} t^i \\ &= N_{1,0}(a_{d,0},\ldots,a_{d,n-1}) + \sum_{i=1}^{n-1} N_{1,i}(a_{d,1},\ldots,a_{d,n-1}) t^i. \end{aligned}$$

A similar expression is obtained for the odd case. All in all we remark that the last $n-1$ coefficients in t of $N_1(a_d)$ form a system with $n-1$ equations of degree 2

$$\mathscr{S}_1 = \{N_{1,i}(a_{d,1},\ldots,a_{d,n-1}) = 0 : 1 \le i \le n-1\}.$$

Such a system is generally of dimension 0 with 2^{n-1} solutions. As $n \le 4$ in practice, solving it is quasi-instantaneous and lead to values for the variables $a_{d,i}$. What is more interesting is that $\mathscr{S}_1$ has a solution in almost every situation.

Proposition 6.4. *Let $\mathbb{F} = \mathbb{F}_{2^{kn}}$ and use the same notations as Proposition 6.3. If $H_g = 0$ or $\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}) \neq 0$, then there exist $x \in \mathbb{F}_{2^{kn}}$ such that $N_1(x) \in \mathbb{F}_{2^k}$.*

Proof. First if $H_g = 0$, then we have $N_1(a_d) = a_d^2 + u_1 = (a_d + \sqrt{u_1})^2$ because of the characteristic and $N_1(x + \sqrt{u_1}) \in \mathbb{F}_{2^k}$ for any $x \in \mathbb{F}_{2^k}$. Now if $H_g \neq 0$ with $\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}) \neq 0$, there is $x \in \mathbb{F}_{2^{kn}}$ such that $N_1(x) \in \mathbb{F}_{2^k}$ if and only if there exist $z \in \mathbb{F}_{2^k}$ such that $N_1(x) + z = 0$. In other words we look for possible roots of $N_1(a_d) + z$ for some $z \in \mathbb{F}_{2^k}$. We use the change of variable $a \leftarrow H_g a_d$ on the polynomial $N_1(a_d) + z$ to obtain $\overline{N_1}(a) = a^2 + a + H_g^{-2}(u_1 + z)$ It is well-known [LN97, prop 3.79 p.127] that polynomials such as $\overline{N_1}(a)$ are split iff $\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_2}(H_g^2(u_1 + z)) = 0$. In particular we can choose $z = \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(u_1)$ if h_1 is monic. If it is not monic, the "chain rule" for traces gives

$$\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_2}(H_g^{-2}(u_1 + z)) = \mathrm{Tr}_{\mathbb{F}_{2^k}|\mathbb{F}_2}(\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}(u_1 + z)).$$

Therefore $\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}(u_1+z))$ needs to be a root α of the 2-polynomial $\mathrm{Tr}_{\mathbb{F}_{2^k}|\mathbb{F}_2}$, which is split [MM07] over $\mathbb{F}_{2^k}$. Next, properties of the trace give

$$\begin{aligned}\alpha + \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}u_1) &= \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}z)\\ &= z\,\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}).\end{aligned}$$

With the hypothesis it is possible to write

$$z = \frac{\alpha + \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}u_1)}{\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2})} \in \mathbb{F}_{2^k}.$$

□

Since $\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}$ is a 2^k-polynomial over $\mathbb{F}_{2^k}$ of degree $2^{k(n-1)}$, the probability that $H_g^{-2} \in \mathbb{F}_{2^{kn}}$ is one of its root is $1/2^k$ which is negligible in practice and is decided once and for all when the curve is chosen.

Corollary 6.5. *If $H_g = 0$ or $\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}) \neq 0$, the system $\mathscr{S}_1$ has a solution over $\mathbb{F}_{2^k}$.*

Proof. From Proposition 6.4 we can almost always find a value $a_d^* \in \mathbb{F}_{2^{kn}}$ such that $N_1(a_d^*) \in \mathbb{F}_{2^k}$. The corollary comes since $N_1(a_d^*) \in \mathbb{F}_{2^k}$ if and only if there exists $(a_{d,1}^*, \ldots, a_{d,n-1}^*) \in \mathbf{V}(\mathscr{S}_1)$. □

Toy Example - a new start: We use the same parameters and notations as in the example of Section 3.4.4. Recall that the R-Decomposition polynomial is

$$\begin{aligned}F(x) = x^4 + (a_2^2 + st + s + 1)x^3 + (a_1^2 + (st + s + 1)a_2^2 + a_2 + st)x^2 +\\ ((st + s + 1)a_1^2 + a_1a_2 + (t + s)a_2^2)x + sta_1^2 + ta_2^2.\end{aligned}$$

Proposition 6.3 is confirmed, and linearization over $\mathbb{F}_4$ gives

$$\begin{aligned}N_1(a_2) &= a_2^2 + st + (s+1)\\ &= a_{2,0}^2 + (t+s)a_{2,1}^2 + ((s+1) + st)\\ &= a_{2,0}^2 + sa_{2,1}^2 + (s+1) + (a_{2,1}^2 + s)t.\end{aligned}$$

It is straightforward to check that $N_1(a_2) \in \mathbb{F}_4$ iff $a_{2,1} = s+1$.

Square Coefficients Keeping previous notations, we have in characteristic 2

$$\begin{aligned}F(x) &= p(x)^2u(x) + p(x)q(x)h_1(x) + q(x)^2w(x)\\ &= x^m + \sum_{i=0}^{m-1} N_{m-i}(a_0, \ldots, a_{m-g})x^i,\end{aligned}$$

with $p(x) = \sum_{i=0}^{d_1} a_i x^i$ and $q(x) = \sum_{i=0}^{d_2} a_{i+d_1+1}x^i \in (\mathbb{F}[a_0, \ldots, a_d])[x]$. In particular, if we let

$$\mathscr{M} = \{a_ia_j : 0 \leq i \neq j \leq d\} \cup \{a_0, \ldots, a_d\},$$

any monomial of $\mathscr{M}$ appearing in a $N_i \in \mathbb{F}[a_0, \ldots, a_d]$ in expression (6.3) has to come from a coefficient in x of the polynomial pqh_1. If no such monomials appears in N_i, then it is a square since the characteristic of the field is even. The number of such square coefficients depends only on h_1.

Definition 6.6 (Length of a polynomial)**.** *Let P be a univariate polynomial. Let d_P and i_p be respectively the degree of the leading and trailing term of P. The length of P is defined as $L_P = d_P - i_P$.*

Proposition 6.7. *Let $\mathbb{F}$ be a field of even characteristic and $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ a hyperelliptic curve of genus g defined over $\mathbb{F}$ with $h_1(x) = \sum_{i=i_h}^{d_h} H_i x^i$, $d_h = \deg h_1$. For a fixed $R = (u,v) \in \mathrm{Jac}(\mathscr{H})$ of weight g, let $F = up^2 + pqh_1 + q^2 w$ be its related Decomposition polynomial, with $p = \sum_{i=0}^{d_1} a_{2i+1}x^i$ and $q = \sum_{i=0}^{d_2} a_{2i+2}x^i$. Let L_h be the length of h_1. There are $g+1-L_h$ squares among the coefficients of F in x.*

Proof. Let $pq = \sum_{i=0}^{d-1} M_i x^i$ with $M_i \in \mathbb{F}[a_0, \ldots, a_d], \deg M_i = 2$ and $\deg_{a_j} M_i = 1$ for $0 \leq j \leq d$, and $h_1 = \sum_{i=0}^{d_h} H_i x^i$. Then the Cauchy product gives

$$pqh_1 = \sum_{i=i_h}^{d-1+d_h} \left(\sum_{j=0}^{i} M_j H_{i-j} \right) x^i = \sum_{i=i_h}^{d-1+d_h} c_i x^i, \tag{6.5}$$

with the convention that $M_d = \ldots = M_{d-1+d_h} = H_{d_h+1} = \ldots = H_{d-1+d_h} = 0$, and $c_i \in \mathbb{F}[a_0, \ldots, a_d]$. Denote by

$$\mathscr{M} = \{a_i a_j : 0 \leq i \neq j \leq d\} \cup \{a_0, \ldots, a_d\} \text{ and } \overline{\mathscr{M}} = \{a_i^2 : 0 \leq i \leq d\}.$$

so that $\mathrm{Supp}\, c_i \subset \mathscr{M}$ for all $i_h \leq i \leq d-1+d_h$, $\deg d_i = \deg e_i = 2$. Recall that $2d_1 = m-g$, $2d_2 = m-g-1$ and that $\deg w = g+1$. We let

$$up^2 = u\left(\sum_{i=0}^{d_1} a_{2i+1}^2 x^{2i} \right) = \sum_{i=0}^{m} d_i x^i,$$
$$q^2 w = w\left(\sum_{i=0}^{d_2} a_{2i+2}^2 x^{2i} \right) = \sum_{i=0}^{m} e_i x^i,$$

with $\mathrm{Supp}\, d_i \subset \overline{\mathscr{M}}$ and $\mathrm{Supp}\, e_i \subset \overline{\mathscr{M}}$ for all i. We can write the Decomposition Polynomial as

$$F(x) = \sum_{i=0}^{i_h-1} (d_i + e_i)x^i + \sum_{i=i_h}^{d-1+d_h} (c_i + d_i + e_i)x^i + \sum_{i=d+d_h}^{m-1} (d_i + e_i)x^i + x^m. \tag{6.6}$$

Then $\mathrm{Supp}(d_i + e_i) \subset \overline{\mathscr{M}}$ and $\mathscr{M} \cap \mathrm{Supp}(c_i + d_i + e_i) \neq \emptyset$ whenever c_i is not zero. From the definition of c_i we see that this can only happen if $H_i = 0$ for all i, which is excluded by the fact that $\mathscr{H}$ is a binary hyperlliptic curve. Now the number of squares among the coefficients of F amounts is read on Expression 6.6 as $m - (d+d_h) + 1 + i_h = g + 1 - L_h$. □

Since F is monic in general, the number of relevant squares among the coefficient of F is $g - L_h$.

Toy example - continued: We continue from the R-Decomposition polynomial

$$F(x) = x^4 + (a_2^2 + st + s + 1)x^3 + (a_1^2 + (st + s + 1)a_2^2 + a_2 + st)x^2 +$$
$$((st + s + 1)a_1^2 + a_1 a_2 + (t + s)a_2^2)x + sta_1^2 + ta_2^2.$$

Since $d_h = 1, i_h = 1$, Proposition 6.7 tells there are $g - L_h + 1 = 3$ squares among F's coefficients, one being 1 since F is monic, the two other being

$$N_1(a_2) = a_2^2 + st + s + 1,$$
$$N_4(a_1, a_2) = sta_1^2 + ta_2^2.$$

The Weil Descent leads to the following system:

$$\mathscr{N} : \begin{cases} a_{2,1}^2 + s, \\ a_{1,1}^2 + sa_{2,0}^2 + sa_{2,1}^2 + a_{2,1} + s, \\ sa_{1,0}^2 + a_{1,0}a_{2,1} + sa_{1,1}^2 + a_{1,1}a_{2,0} + a_{1,1}a_{2,1} + a_{2,0}^2 + a_{2,1}^2, \\ sa_{1,0}^2 + a_{1,1}^2 + a_{2,0}^2 + (s+1)a_{2,1}^2 \end{cases}$$

and since we solved $\mathscr{S}_1$ to find $a_{2,1} = s+1$, we can build the system $\mathscr{S}_2 = \{N_{i,1}(\overline{\mathbf{a}}, s+1) : 1 \leq i \leq 4\}$ with $\overline{\mathbf{a}} = (a_{1,0}, a_{1,1}, a_{2,0})$. As $N_{1,1}(\mathbf{a}, s+1) = 0$, we have

$$\mathscr{S}_2 : \begin{cases} a_{1,1}^2 + s a_{2,0}^2 + s, \\ s a_{1,0}^2 + (s+1) a_{1,0} + s a_{1,1}^2 + a_{1,1} a_{2,0} + (s+1) a_{1,1} + a_{2,0}^2 + s, \\ s a_{1,0}^2 + a_{1,1}^2 + a_{2,0}^2 + 1 \end{cases}$$

and we notice that the first expression has become a square once evaluated at $a_{2,1} = s+1$. This is a general fact that happens if $\mathrm{LC}(h_1)$ is in a subfield of $\mathbb{F}_{2^{kn}}$.

Additional squares depending on $\mathbf{LC}(h_1)$ Note that N_1 is a square if and only if $\deg h_1 = d_h < g$. Assume this is the case, and first that m is even; then the leading term in x of pqh_1 is $\mathrm{LT}_x(pqh_1) = H_{d_h} a_d$, and it appears in the coefficient N_{1+g-d_h} as the only one involving a monomial from $\mathscr{M}$. If m is odd, the same observation can be done with a_{d_1} instead of a_d. If $H_{d_h} \in \mathbb{F}_{2^k}$ we write

$$H_{d_h} a_d = H_{d_h} \left(a_{d,0} + \sum_{i=1} a_{d,i} t^i \right),$$

and observe that in $N_{1+g-d_h}(a_0, \ldots, a_d) = \sum_{i=0}^{n-1} N_{1+g-d_h,j}(\mathbf{a}) t^j$ the monomial $a_{d,0}$ appears only in the coefficient of degree 0 in t. If a solution $\mathbf{a}^* = (a_{d,1}^*, \ldots, a_{d,n-1}^*)$ of $\mathscr{S}_1$ is found, as the Weil Descent here deals only with the $n-1$ last coefficients, we find $n-1$ new square equations with each $N_{1+g-d_h,j}$, for $1 \leq j \leq n-1$.

Remark 6.8. *All results hold for binary elliptic curves.*

Toy example - continued: The $1+g-d_h = 2$nd coefficient of F is

$$N_2(a_1, a_2) = a_1^2 + (st + s + 1) a_2^2 + a_2 + st,$$

and we notice that indeed the only term with support in $\mathscr{M} = \{a_1, a_2, a_1 a_2\}$ is $H_1 a_2 = a_2$. The linearization over $\mathbb{F}_{2^k}$ and evaluation at s gives

$$\begin{aligned} N_2(a_{1,0}, a_{1,1}, a_{2,0}) &= a_{1,0}^2 + (t+s) a_{1,1}^2 + (st + (s+1)) a_{2,0}^2 + a_{2,0} + t + 1 \\ &= N_{2,0}(a_{1,0}, a_{1,1}, a_{2,0}) + N_{2,1}(a_{1,0}, a_{1,1}, a_{2,0}) t, \end{aligned}$$

with $N_{2,1}(a_{1,0}, a_{1,1}, a_{2,0}) = a_{1,1}^2 + s a_{2,0}^2 + s$, indeed a square in $\mathbb{F}_{2^k}[a_{1,0}, a_{1,1}, a_{2,0}]$.

6.1.2 Reducing the degree of PDP_{ng} systems

We now assume that the field is $\mathbb{F}_{2^{kn}} = \mathbb{F}_{2^k}[t] / \langle P(t) \rangle$ with P an irreducible polynomial of degree n over $\mathbb{F}_{2^k}$. If $\mathscr{H} : y^2 + h_1(x) y = h_0(x)$ is a hyperelliptic curve of genus g defined over $\mathbb{F}_{2^{kn}}$, we fix $LC(h_1)$ to 1, as it is generally the case in practice, but we do not fix its degree $d_h \leq g$. Recall that the goal is to solve as efficiently as possible PDP_{ng} instances, i.e. we fix $R = (u, v) \in \mathrm{Jac}(\mathscr{H})$ and try to decompose it as

$$R = (P_1) + \cdots + (P_{ng}), (P_i) \in \mathscr{B} = \{(P) : P \in \mathscr{H}, x(P) \in \mathbb{F}_{2^k}\}.$$

Now we show how to reduce the degree of the underlying ideal, using the previous properties of the R-Decomposition polynomial

$$F(x) = up^2 - pqh_1 + q^2 w = x^{ng} + \sum_{i=0}^{ng-1} N_{ng-i}(a_0, \ldots, a_d) x^i.$$

where $d = (n-1)g$. Again, we assume that ng is even for the sake of simplicity. Following Proposition 6.3 we obtain a first system over $\mathbb{F}_{2^k}$

$$\mathscr{S}_1 = \{N_{1,i}(a_{d,1}, \ldots, a_{d,n-1}) = 0 : 1 \leq i \leq n-1\},$$

and we let $\mathbf{a}^* = (a^*_{d,1}, \ldots, a^*_{d,n-1})$ be a solution of $\mathscr{S}_1$, see Proposition 6.4. In our experiments $\mathscr{S}_1$ always had one solution over $\mathbb{F}_{2^k}$. Let $\overline{\mathbf{a}} = (a_{0,0}, \ldots, a_{0,n-1}, \ldots, a_{d-1,n-1}, a_{d,0})$ and evaluate the remaining equations to form the system

$$\mathscr{S}_2 = \{N_{i,j}(\overline{\mathbf{a}}, \mathbf{a}^*) : 2 \leq i \leq ng, 1 \leq j \leq n-1\}.$$

with $(ng-1)(n-1)$ variables and equations. This quadratic system is generally of dimension 0 and therefore generates an ideal of degre $2^{(ng-1)(n-1)}$. It is straightforward to verify that solutions of $\mathscr{N}$ can be described by solutions of $\mathscr{S}_1$ and $\mathscr{S}_2$.

When we start the Weil Descent over $\mathbb{F}_{2^{kn}}$, we can again use the characteristic: indeed, if N_i is a square, then it can be written $N_i = \overline{N}_i^2$ with $\deg \overline{N}_i = 1$. We write

$$\begin{aligned} N_i(a_0 \ldots, a_d) = \sum_{i=0}^{n-1} N_{i,j}(\mathbf{a}) t^j &= \overline{N}_i(a_0 \ldots, a_d)^2 \\ &= \left(\sum_{i=0}^{n-1} \overline{N}_{i,j}(\mathbf{a}) t^j \right)^2 \\ &= \sum_{i=0}^{n-1} \widetilde{N}_{i,j}(\mathbf{a})^2 t^j \end{aligned}$$

with $\deg \overline{N}_{i,j} = \deg \widetilde{N}_{i,j} = 1$, and the polynomials $\widetilde{N}_{i,j}$ are linear combinations of the linear polynomials $\overline{N}_{i,j}$. As we have

$$N_{i,j}(\mathbf{a}) = 0 \Leftrightarrow \widetilde{N}_{i,j}(\mathbf{a}) = 0 \Leftrightarrow \widetilde{N}_{i,j}(\overline{\mathbf{a}}, \mathbf{a}^*) = 0,$$

we can build a new system from $\mathscr{S}_2$ by replacing any $N_{i,j}(\overline{\mathbf{a}}, \mathbf{a}^*) \in \mathscr{S}_2$ that is a square by its square root, namely the linear equation $\widetilde{N}_{i,j}(\overline{\mathbf{a}}, \mathbf{a}^*)$. We call this new system *unsquared* and denote it by $\sqrt{\mathscr{S}_2}$ from now on.

Proposition 6.9. *Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be an hyperelliptic curve of genus g defined over $\mathbb{F}_{2^{kn}}$. Let L_h be the length of h_1, and assume h_1 is monic. Let $R \in \mathrm{Jac}(\mathscr{H})$ of weight g, and let $\mathscr{S}$ be the Nagao system describing the PDP_{ng} instance related to R. The unsquared system $\sqrt{\mathscr{S}_2}$ contains $(n-1)(g-L_h)$ linear equations.*

Proof. Recall that $L_h = d_h - i_h$, where d_h resp. i_h is the degree of the leading resp. trailing term of h_1. There are two possible cases:

- If $d_h = g$, Proposition 6.7 tells us that all squares in $\mathscr{S}_2$ come from the i_h^{th} coefficients of lower degree in F, so that $\sqrt{\mathscr{S}_2}$ contains $(n-1)i_h$ linear equations.
- If $d_h < g$, N_1 counts as a square in Proposition 6.7 but we do not use it to build $\sqrt{\mathscr{S}_2}$ since it was used for $\mathscr{S}_1$, so that $g - L_h - 1$ square coefficients are used. Using the description before Remark 6.8, the Weil Descent gives us $n-1$ additional square equations in $\mathscr{S}_2$. Overall, this leads to $(n-1)(g-L_h)$ linear equations in $\sqrt{\mathscr{S}_2}$.

In any case, there are $(n-1)(g-L_h)$ linear equations in $\sqrt{\mathscr{S}_2}$. □

It never occured in our experiments that a linear equation was a combination of the others. As systems like $\mathscr{S}_2$ are generally of dimension 0 the following heuristic is therefore reasonable:

Heuristic 6.10. *The linear equations created during the "unsquaring" process are independent. In other words, the ideal generated by $\sqrt{\mathscr{S}_2}$ has dimension 0.*

Under this heuristic, the degree of $\mathscr{S}_2$ is divided by 2 in general with every linear equation replacing a quadratic one as any linear equation is used to eliminate a variable. A new system $\mathscr{S}_3$ is built that way, containing the remaining quadratic equations. If L_h is the length of h_1, there are $(n-1)((n-1)g+L_h-1)$ variables and as much quadratic equations left in $\mathscr{S}_3$. Hence it is generally of dimension 0 and has degree:

$$\deg \mathscr{S}_3 = 2^{(n-1)((n-1)g+L_h-1)}.$$

It is interesting that the length of h_1 has this impact on the PDP_{ng} instance solving. As $1 \leq i_h \leq d_h \leq g$, we see that the best case happens when $L_h = 0$ and $\mathrm{LC}(h_1) \in \mathbb{F}_{2^k}$, e.g. when h_1 has only one term with coefficient in the subfield of interest, in which case we find the sharp bound

$$d_{\text{opt}} = 2^{(n-1)((n-1)g-1)} \leq \deg \mathscr{S}_3.$$

Toy example concluded : From $\mathscr{S}_2$, we build the unsquared system

$$\sqrt{\mathscr{S}_2} : \begin{cases} a_{1,1} + (s+1)a_{2,0} + (s+1), \\ sa_{1,0}^2 + (s+1)a_{1,0} + sa_{1,1}^2 + a_{1,1}a_{2,0} + (s+1)a_{1,1} + a_{2,0}^2 + s, \\ (s+1)a_{1,0} + a_{1,1} + a_{2,0} + 1 \end{cases}$$

We have $(n-1)((n-1)g+d_h-i_h-1) = 1$, and the generated ideal has indeed degree 2. For those small parameters $\sqrt{\mathscr{S}_2}$ can be solved by hand but nevertheless we compute a Gröbner basis wrt lexicographical order $a_{1,0} > a_{1,1} > a_{2,0}$ in *Shape Position* to find

$$\begin{cases} a_{1,0} + (s+1)a_{2,0} + s + 1, \\ a_{1,1} + (s+1)a_{2,0} + s + 1, \\ a_{2,0}^2 + sa_{2,0} + 1 \end{cases}$$

6.1.3 Analysis of the degree reduction for genus 2 binary curves

Genus 2 curves have been proposed as an alternative for curve based cryptosystems [BD04]. The article also proposes a complete classification of binary genus 2 curves inspired from [CJ03] that we succintly present first. We then apply the degree reduction process and sum up the expected degree reduction for $n = 2, 3, 4$.

Classification of genus 2 binary curves Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be a genus 2 curve defined over a field $\mathbb{F}_{2^n}$, so that $\deg h_1 = d_h \leq 2$ and $\deg h_0 = 5$. We write $h_1(x) = H_2x^2 + H_1x + H_0$ and $h_0(x) = x^5 + \sum_{i=0}^{4} f_i x^i$. There are three types of binary genus 2 curves, decided by h_1.

1. **Type I curves:** A curve is a type I curve if and only if $d_h = 2$. It then falls into one of two subtypes whether h_1 has roots in the ground field or not. Let $t \in \mathbb{F}_{2^{kn}}$ be an element of (absolute) trace 1 and $\varepsilon \in \mathbb{F}_2$. See [BD04, CJ03] for details on their definition. We emphasize that if n is odd then we can set $t = 1$.

   - If h_1 is irreducible over $\mathbb{F}_{2^n}$, then $\mathscr{H}$ is type I_a and is isomorphic to the curve

   $$\mathscr{H}_{I_a} : y^2 + (x^2 + H_1x + tH_1^2)y = x^5 + t\varepsilon x^4 + f_1x + f_0.$$

   - Else h_1 has its roots in $\mathbb{F}_{2^n}$, $\mathscr{H}$ is type I_b and isomorphic to the curve defined by

   $$\mathscr{H}_{I_b} : y^2 + x(x + H_1)y = x^5 + t\varepsilon x^4 + f_1x + f_0.$$

2. **Type II curves:** If $d_h = 1$, there are two subtypes depending on the parity of the extension degree n.

- If n is odd then $\mathscr{H}$ is isomorphic to
$$\mathscr{H}_{II} : y^2 + xy = x^5 + f_3x^3 + \varepsilon x^2 + f_0.$$
- If n is even then $\mathscr{H}$ is isomorphic to
$$\mathscr{H}_{II} : y^2 + H_1xy = x^5 + \varepsilon' x^3 + t\varepsilon H_1^2 x^2 + f_0,$$
with $\varepsilon' \in \mathbb{F}_2$.

3. **Type III curves:** Lastly if $d_h = 0$ then $\mathscr{H}$ is isormorphic to the curve defined by
$$\mathscr{H}_{III} : y^2 + y = x^5 + f_3x^3 + f_1x + t\varepsilon.$$

There are subtypes for type *III* as well but as such curves are known to be *supersingular*, therefore they are considered weak because of the Frey-Rück attack [FMR94], so we do not go into further details. There are also other forms for h_0 for each type, coming at the expense of more coefficients in h_1. More non-zero coefficients implies a more expensive Jacobian arithmetic in term of elementary operations, so this is usually avoided as much as possible in practice. Hence we focus on the above forms for genus 2 binary curves, and we call them *canonical forms* in the rest of the presentation.

As the length of h_1 plays a crucial role in the degree reduction estimations, we focus briefly on curves with $h_1(x) = x^2$. Such curves are isomorphic to type II curves using the change of variables $x = 1/x'$ and $y = y'^3 + \sqrt{f_0}$.

Comparisons of degree reductions depending on canonical forms Table 6.1 shows the minimal degrees obtained after the degree reduction process applied to each canonical form of curves defined over a field $\mathbb{F}_{2^{kn}}$ with $2 \leq n \leq 4$. The d_{Nag} column shows the degree expected by a Nagao modelling without refinement while d_{red} resp. d_{opt} stands for resp. reduced degree and optimal degree as in previous section. Column Univariate gives the number of variables that can be determined by using equation N_1, and columns Square and LC(h_1) show the number of linear equations to be expected after building the system $\mathscr{S}_2$. If n is even, then $\text{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(1) = 0$. Proposition 6.4 cannot be applied and the system $\mathscr{S}_1$ may not have a solution. This is indicated by a "$\leq$" sign in the corresponding cell. Nonetheless we only indicate the minimal degree for each type of curve.

Table 6.1: Degree reduction in genus 2 for small extension fields

Type	$\deg h_1$	L_h	n	Univariate	Square	LC(h_1)	d_{red}	d_{Nag}
I_a	2	2	2	≤ 1	-	-	8	16
			3	2	-	-	1024	4096
			4	≤ 3	-	-	2^{21}	2^{24}
I_b	2	1	2	≤ 1	1	-	4	16
			3	2	2	-	256	4096
			4	≤ 3	3	-	2^{18}	2^{24}
I_b with $h_1(x) = x^2$	2	0	2	≤ 1	2	-	$2 = d_{\text{opt}}$	16
			3	2	4	-	$64 = d_{\text{opt}}$	4096
			4	≤ 3	6	-	$2^{15} = d_{\text{opt}}$	2^{24}
II	1	0	2	1	1	≤ 1	$2 = d_{\text{opt}}$	16
			3	2	2	≤ 2	$64 = d_{\text{opt}}$	4096
			4	3	3	≤ 3	$2^{15} = d_{\text{opt}}$	2^{24}
III	0	0	2	1	1	1	$2 = d_{\text{opt}}$	16
			3	2	2	2	$64 = d_{\text{opt}}$	4096
			4	3	3	3	$2^{15} = d_{\text{opt}}$	2^{24}

For type I_a the reduction comes only by using the univariate equations to find values for some variables. The type I_b has a particular subcase when $h_1(x) = x^2$, i.e. when $H_1 = 0$ where d_{opt} can be reached. The polynomial h_1 for type II depends on the extension degree and $\text{LC}(h_1)$'s base field. As we mentioned already, if $H_1 \in \mathbb{F}_{2^k}$ then additional squares can be found in the system. For type III, h_1 is always monic so we can exploit all steps of reduction. This reinforces the weakness of those curves. Finally notice that if kn is odd and as $\text{LC}(h_1) = 1$ in practice, then the degree reduction for type II curves reaches $d_{opt} = 2^{(n-1)((n-1)g-1)}$. This reveals a weakness for this type while they were suggested as potential new standards for implementation in [BD04], and we use this to design a practical Discrete Logarithm computation for realistic parameters, see Section 6.3.3. Curves of type I_b with $h_1(x) = x^2$ could also be interesting for practical purpose, as their equation is as sparse as type II's. However the reduced degree also reach d_{opt} for those curves, suggesting a greater weakness than other type I_b curves.

6.2 Ideal degree reduction in the Summation approach

This Section starts with an example: for certain types of curves, some variables in Summation Sets appear with even exponents only. We refer to them as *squared variables*, and such variables can thus be replaced by their "square root". This divides the degree of the ideal by 2 with each replacement. This was already observed in [FHJ+14] for binary elliptic curves to reduce the degree of the PDP_n systems by a factor 2^{n-1}. The number of such variables depends again on the length of the polynomial h_1 defining the curve, and as h_1 can have higher degree when $g \geq 2$, the situation becomes more complex for hyperelliptic curves. In particular, reduction of the degree can be achieved even when no squared variables are found among a Summation Set. The complete analysis requires that we consider the more general context of the action of the Frobenius automorphism over a variety given by a polynomial parametrization in positive characteristic. More precisely, in a field $\mathbb{F}$ with $\mathrm{Char}(\mathbb{F}) = p$, let $\mathbf{a} = (a_1, \ldots, a_l)$ and $\mathbf{X} = (X_1, \ldots, X_m)$ for $l < m$, and let $I \subset \mathbb{F}[\mathbf{a}, \mathbf{X}]$ be an ideal describing a polynomial parametrization as

$$I = \langle\, X_1 - P_1(\mathbf{a})^p, \ldots, X_k - P_k(\mathbf{a})^p, X_{k+1} - P_{k+1}(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a}) \,\rangle,$$

for some polynomials P_i. Implicitizing this variety means eliminating the variables $\mathbf{a}$, equivalently computing $I_e = I \cap \mathbb{F}[\mathbf{X}]$ the *ideal of relations between the P_i's*. If we let J be

$$J = \langle\, X_1 - P_1(\mathbf{a}), \ldots, X_k - P_k(\mathbf{a}), \ldots, X_m - P_m(\mathbf{a}) \,\rangle,$$

and $J_e = J \cap \mathbb{F}[\mathbf{X}]$, then we show that it is equivalent to find points in $V(I_e)$ or to find points in $V(J_e)$. Hence we can work with J instead of I.

A first advantage is that the degree of the equations defining J is lower, so we expect elimination to be faster in practice. We then estimate in general the degree reduction when working with $V(J_e)$ instead of $V(I_e)$. As natural weights appear for J_e, working with J_e means dividing the degree of I_e by the product of the weights, up to a certain constant C_1 that we call *the degree ratio* (Definition 6.15), and which seems to depend on the length L_h of the polynomial h_1. We then observe that the parametrization of $\mathscr{V}_{m,R}$ in even characteristic is a special case of polynomial parametrizations with Frobenius action: we indeed know from Section 6.1 that some N_i's defining the parametrization are squares. The lesser degree of the variety generated by the "square root" equations translates to the Weil Restriction, with an exact formula depending on the degree ratio C_1, and that mainly says that the degree is divided by the product of the weights — here, a power of 2. In our experiments, the degree ratio C_1 is always a power of 2, and in genus 2, $C_1 = 1$ for 3 types of curves over 5 possible — we count type I_b with $h_1(x) = x^2$ as a type on its own. All types of genus 2 curves are treated, and depending on the curve's type, the degree of the ideal to solve a PDP_{2n} instance can be reduced by a factor up to $2^{2(n-1)}$. Higher genus hyperelliptic curves are discussed: following our experiments, we conjecture that $C_1 = 2^{L_h} \leq 2^g$. With $d_{\mathrm{Nag}} = 2^{n(n-1)g}$, this gives a first degree reduction to an ideal $\mathbb{I}$ of degree

$$\begin{aligned} \deg \mathbb{I} &= 2^{nL_h} \cdot \frac{d_{\mathrm{Nag}}}{2^{(n-1)g+L_h}} \\ &= 2^{(n-1)((n-1)g+L_h)}. \end{aligned}$$

Finally it is possible to achieve further reduction with another use of the univariate coefficient in Section 6.1: the variable a_d can be used instead of the variable e_1. Compared to Nagao's degree reduction in Section 6.1, a difference appears: for an extension of degree n, when the polynomial h_1 is not of degree g, it is not possible to achieve an additional 2^{n-1} factor, but only a smaller power of 2. If h_1 has degree g, then in general a factor of 2^{n-1} can be obtained.

For curves with $\deg h_1 = g$ and $L_h = 0$, a second reduction to an ideal $\mathbb{I}_2$ of degree

$$\begin{aligned}\deg \mathbb{I}_2 &= \frac{\deg \mathbb{I}}{2^{n-1}} \\ &= 2^{(n-1)((n-1)g-1)} = d_{\text{opt}}\end{aligned}$$

can be achieved just as in Nagao's modelling. Table 6.3 sums all the reduction achieved with Summation modelling for genus 2 binary curves.

6.2.1 Polynomial Parametrizations in Positive Characteristic

Squared variables in genus 2 Let $\mathbb{F} = \mathbb{F}_{32}$ with primitive element t, and $\mathscr{H}_1 : y^2 + (x^2)y = x^5 + x^4 + t^{15}x + t^7$ and $\mathscr{H}_2 : y^2 + xy = x^5 + t^3x^3 + x^2 + t^{15}$ two genus 2 curves of type respectively I_b and II. We fix $R_1(x^2 + t^{29}x + t^{13}, t^{21}x + t^7) \in \text{Jac}(\mathscr{H}_1)$ and $R_2(x^2 + t^{28}x + t^{19}, t^{11}x + t^4) \in \text{Jac}(\mathscr{H}_2)$ and compute 5 and 6-Specialized Summation Sets with symmetric variables for each to find:

$$\mathbb{S}_{5,1} = \begin{cases} e_2^2 + t^{29}e_2 + t^{27}\boxed{e_1^2} + t^{18}, \\ e_3 + t^{15}e_2 + t^{17} \end{cases} \qquad \mathbb{S}_{5,2} = \begin{cases} \boxed{e_2^2} + t^{25}e_1^2 + e_1 + t^3, \\ e_3 + t^{19}e_1 + t^{13} \end{cases}$$

$$\mathbb{S}_{6,1} = \begin{cases} t^8 e_4 \boxed{e_2^2} + t^{30} e_4 \boxed{e_1^2} + t^{11} e_4 + e_3^3 + t^5 e_2^2 + t^2 e_3 \boxed{e_2^2} + \\ \quad t^{22} e_3 \boxed{e_1^2} + t^4 e_3 + t \boxed{e_2^2} + t^7 \boxed{e_1^2} + t^{18}, \\ e_4^2 + t^{17} e_4 + t^{30} e_3^2 + t e_3 + t^3 \boxed{e_1^2} + t^{30}, \\ e_4 e_3 + t^{20} e_4 + t^{25} e_3^2 + t^4 e_3 + t^{15} \boxed{e_2^2} + t^6 \boxed{e_1^2} + t^{22} \end{cases}$$

$$\mathbb{S}_{6,2} = \begin{cases} e_4^2 + t^7 \boxed{e_2^2} + t^{26} e_1^2 + t^7 e_1 + 1, \\ t^{12} e_4 e_1 + t^9 e_4 + \boxed{e_3^2} + t^{25} \boxed{e_2^2} + t^{10} e_1^2 + \\ \quad t^{25} e_1 + t^{11} \end{cases}$$

Focusing first on $\mathscr{H}_1$, we observe that e_1 appears only with even exponent 0 or 2 in $\mathbb{S}_{5,1}$ and $\mathbb{S}_{6,1}$, and that e_2 appear only with exponent 0 or 2 as well in $\mathbb{S}_{6,1}$. Now for $\mathscr{H}_2$, we check that e_2 appears only with even exponent in $\mathbb{S}_{5,2}, \mathbb{S}_{6,2}$ and that the same can be said for e_3 in $\mathbb{S}_{6,2}$. In Proposition 6.7 and the toy-example of Section 6.1 we also observe that in both cases, the squared variables e_i are exactly the variables equal to a non-squared coefficient in the parametrization of $\mathscr{V}_{m,R}$. This implies a natural system of weights on the non-squared variables. To obtain a general description, we temporarily switch to the setting of the Frobenius action over ideal of relations in positive characteristic.

Action of the Frobenius automorphism and degree of parametrization Let $\mathbb{F}$ be a field of characteristic $p \geq 2$, and $\sigma(x) = x^p$ the Frobenius Automorphism. If $f = \sum c_\alpha \mathfrak{m}_\alpha \in \mathbb{F}[X_1, \ldots, X_m]$, we denote by $f^\sigma = \sum c_\alpha^p \mathfrak{m}_\alpha$ the polynomial obtained by Frobenius action over its coefficients. We observe that $f^\sigma(X_1^p, \ldots, X_m^p) = f(X_1, \ldots, X_m)^p$. For an ideal $I = \langle g_1, \ldots, g_r \rangle$, define $I^\sigma = \langle g_1^\sigma, \ldots, g_r^\sigma \rangle$ and $I^p = \langle f^p : f \in I \rangle$. Assume $m \geq 2$, let $1 \leq l \leq k \leq m$ be integers and let $\mathbf{a} = (a_1, \ldots, a_l)$, $\mathbf{X} = (X_1, \ldots, X_m)$. For polynomials $P_1, \ldots, P_m \in \mathbb{F}[\mathbf{a}]$, we consider the ideals

$$\begin{aligned} I &= \langle\ X_i - P_i(\mathbf{a})^p\ :\ 1 \leq i \leq k\ \ ;\ \ X_i - P_i(\mathbf{a}),\ k+1 \leq i \leq m\ \rangle, \\ J &= \langle\ X_i - P_i(\mathbf{a})\ :\ 1 \leq i \leq m\ \rangle. \end{aligned}$$

We also consider their ideals of relations, that is to say their l-th elimination ideals

$$I_e = I \cap \mathbb{F}[\mathbf{X}],\ J_e = J \cap \mathbb{F}[\mathbf{X}].$$

Such ideals are prime and therefore radical (Corollary 1.72). It is straightforward to check that $(z_1, \ldots, z_m, a_1, \ldots, a_l) \in V(I)$ if and only if $(\sqrt[p]{z_1}, \ldots, \sqrt[p]{z_k}, z_{k+1}, \ldots, z_m, a_1, \ldots, a_l) \in V(J)$. This suggests a natural weight p on $X_{k+1}, \ldots, X_m$. We turn to eliminations ideals and derive a similar property.

Lemma 6.11. *Let $I_e = I \cap \mathbb{F}[\mathbf{X}]$ and $J_e = J \cap \mathbb{F}[\mathbf{X}]$ be the ideals of relations associated to I, J.*

1. $g \in J_e \Leftrightarrow g^\sigma(X_1,\ldots,X_k,X_{k+1}^p,\ldots,X_m^p) \in I_e$.
2. $g \in I_e \Leftrightarrow g(X_1^p,\ldots,X_k^p,X_{k+1},\ldots,X_m) \in J_e$.

Proof. From the definition of I_e and J_e we get

$$g \in J_e \Leftrightarrow g(P_1,\ldots,P_m) = 0 \text{ and } g \in I_e \Leftrightarrow g(P_1^p,\ldots,P_k^p,P_{k+1},\ldots,P_m) = 0.$$

Then we observe that

1. $g \in J_e \Leftrightarrow g(P_1,\ldots,P_m)^p = 0 \Leftrightarrow g^\sigma(P_1^p,\ldots,P_m^p) = 0 \Leftrightarrow g^\sigma(X_1,\ldots,X_k,X_{k+1}^p,\ldots,X_m^p) \in I_e$.
2. $g \in I_e \Leftrightarrow g(X_1^p,\ldots,X_k^p,X_{k+1},\ldots,X_m) \in J_e$.

□

In the next Proposition, we introduce an important ideal I'. Indeed, it is used to show that finding points in $V(I_e)$ or $V(J_e)$ are equivalent tasks, and it is also involved in the estimation of $\deg J_e$.

Proposition 6.12. *Let* $I_e = \langle f_1,\ldots,f_s\rangle$ *and* $J_e = \langle g_1,\ldots,g_r\rangle$, $I' = \left\langle\ g_i^\sigma(X_1,\ldots,X_k,X_{k+1}^p,\ldots,X_m^p), 1 \le i \le r\ \right\rangle$ *and* $J' = \left\langle\ f_i(X_1^p,\ldots,X_k^p,X_{k+1},\ldots,X_m)\ , 1 \le i \le s\ \right\rangle$. *Then* $I_e^p \subset I' \subset I_e$ *and* $J_e^p \subset J' \subset J_e$.

Proof. From Lemma 6.11 we know that $I' \subset I_e$. Let $f \in I_e$. Lemma 6.11 also gives that $f(X_1^p,\ldots,X_k^p,X_{k+1},\ldots,X_m) \in J_e$. Hence there exists $q_i \in \mathbb{F}[\mathbf{X}]$ such that

$$f(X_1^p,\ldots,X_k^p,X_{k+1},\ldots,X_m) = \sum_{i=1}^{r} q_i(X_1,\ldots,X_m)g_i(X_1,\ldots,X_m).$$

Evaluating at $X_1,\ldots,X_k,X_{k+1}^p,\ldots X_m^p$ and taking p-th power give

$$\begin{aligned} f(X_1^p,\ldots,X_m^p)^p &= \sum_{i=1}^{r} q_i(X_1,\ldots,X_k,X_{k+1}^p,\ldots,X_m^p)^p g_i(X_1,\ldots,X_k,X_{k+1}^p,\ldots,X_m^p)^p \\ &= \sum_{i=1}^{r} q_i^\sigma(X_1^p,\ldots,X_k^p,X_{k+1}^{p^2},\ldots,X_m^{p^2}) g_i^\sigma(X_1^p,\ldots,X_k^p,X_{k+1}^{p^2},\ldots,X_m^{p^2}) \end{aligned}$$

which means that

$$f(X_1,\ldots,X_m)^p = \sum_{i=1}^{r} q_i^\sigma(X_1,\ldots,X_k,X_{k+1}^p,\ldots,X_m^p)^p g_i^\sigma(X_1,\ldots,X_k,X_{k+1}^p,\ldots,X_m^p)$$

so that $f^p \in I'$. The other inclusions follow similar arguments. □

Corollary 6.13. *With the previous notations,* I_e *is the radical of* I' *and* J_e *is the radical of* J'.

Proof. Proposition 6.12 implies that $I_e \subset \sqrt{I'}$. As $\sqrt{I'}$ is the smallest radical ideal containing I', and since I_e is radical, then in fact $I_e = \sqrt{I'}$. The other statement is proved the same way. □

Keeping the same notations for ideals, we let $g_1,\ldots,g_r$ be generators for J_e and we define the ideal $I' = \langle g_i^\sigma(X_1,\ldots,X_k,X_{k+1}^p,\ldots,X_m^p)\ :\ 1 \le i \le r\rangle$. Assuming the base field is algebraically closed, we know from Corollary 6.13 that $\sqrt{I'} = I_e$, so that $\mathbf{I}(\mathbf{V}(I')) = I_e$ and $\mathbf{V}(I_e) = \mathbf{V}(I')$. This gives

$$\begin{aligned} (z_1,\ldots,z_m) \in \mathbf{V}(I_e) &\Leftrightarrow (z_1,\ldots,z_m) \in \mathbf{V}(I') \\ &\Leftrightarrow g_i^\sigma(\sqrt[p]{z_1}^{\,p},\ldots,\sqrt[p]{z_k}^{\,p},z_{k+1}^p,\ldots,z_m^p) = 0,\ \forall\ 1 \le i \le r \\ &\Leftrightarrow g_i(\sqrt[p]{z_1},\ldots,\sqrt[p]{z_k},z_{k+1},\ldots,z_m)^p = 0,\ \forall\ 1 \le i \le r \\ &\Leftrightarrow (\sqrt[p]{z_1},\ldots,\sqrt[p]{z_k},z_{k+1},\ldots,z_m) \in \mathbf{V}(g_1,\ldots,g_r) = \mathbf{V}(J_e). \end{aligned}$$

This implies that it is equivalent to work with $\mathbf{V}(I_e)$ or $\mathbf{V}(J_e)$. Since the two associated ideals are radical, in practice we can use either I_e or J_e for computations.

Proposition 6.14. *Let $w_1 = \ldots = w_k = 1$ and $w_{k+1} = \ldots = w_m = p$, and let also $w'_1 = \ldots = w'_{m-k} = p$, $w'_{m-k+1} = \ldots = w'_m = 1$. Consider the systems of weight $w = (w_1, \ldots, w_m)$ and $w' = (w'_1, \ldots, w'_m)$. Keeping previous notations, we have:* $\deg_w J_e = \dfrac{\deg I'}{p^{m-k}}$ *and* $\deg_{w'} I_e = \dfrac{\deg J'}{p^k}$.

Proof. Let first $A = (\mathbb{F}[X_1, \ldots, X_m], (1, \ldots, 1))$ be the polynomial algebra with standard graduation. Let $w = (w_1, \ldots, w_m)$ with $w_1 = \ldots = w_k = 1$ and $w_{k+1} = \ldots = w_m = p$, and consider the w-graded algebra $A_w = (\mathbb{F}[Y_1, \ldots, Y_m], (w_1, \ldots, w_m))$. We see the ideal J_e in this algebra. Using the injective homomorphism of graded algebras

$$\begin{array}{rccc} \varphi : & A_w & \longrightarrow & A \\ & Y_i & \longmapsto & X_i^{w_i} \end{array}$$

we have $\varphi(J_e^\sigma) = I'$, hence by Proposition 1.37, $\deg_w J_e^\sigma = \frac{\deg I'}{p^{m-k}}$.Wlog. we can assume that the generators $g_1, \ldots, g_r$ of J_e form a Gröbner Basis for some total degree order. Since $\mathrm{LM}(g_i) = \mathrm{LM}(g_i^\sigma)$ for all i, then $\{g_i^\sigma : 1 \leq i \leq r\}$ is a Gröbner Basis for J_e^σ, hence $\deg_w J_e = \deg_w J_e^\sigma$. The first equality follows and the other is obtained by adapting the whole argument. □

6.2.2 Application to Specialized Summation Varieties in even characteristic

Consider a hyperelliptic curve $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ of genus g, defined over a field $\mathbb{F}$ of characteristic 2. Let $R \in \mathrm{Jac}(\mathscr{H})$ of weight g and F be the Decomposition Polynomial associated with the PDP_m instance related to R:

$$F(x) = x^m + \sum_{i=0}^{m-1} N_i(\mathbf{a}) x^i.$$

If d_h (resp. i_h) is the degree of the leading (resp. trailing) coefficient of h_1 and $L_h = d_h - i_h$, Proposition 6.7 tells us that F has $k = g - L_h$ squared coefficients - not counting the fact that it is monic. Assume that $L_h < g$, and for simplicity, renumber the coefficients of F and the e_i such that the squares are $N_1(\mathbf{a}) = \tilde{N}_1(\mathbf{a})^2, \ldots, N_k(\mathbf{a}) = \tilde{N}_k(\mathbf{a})^2$, and then the ideals corresponding to the previous paragraph are

$$\begin{aligned} I &= \langle\ e_i + N_i(\mathbf{a}) \ :\ 1 \leq i \leq m\ \rangle,\ I_e = I \cap \mathbb{F}[\mathbf{e}], \\ J &= \langle\ e_i + \tilde{N}_i(\mathbf{a}) \ :\ 1 \leq i \leq k,\ e_i + N_i(\mathbf{a}),\ k+1 \leq i \leq m\ \rangle,\ J_e = J \cap \mathbb{F}[\mathbf{e}]. \end{aligned}$$

Estimating this reduction depends on the shape of the polynomial h_1 as $\deg I'$ is linked to $\deg I_e$ depending on the length L_h of h_1.

Definition 6.15. *The degree ratio between I' and I_e is noted* $C_1 = \dfrac{\deg I'}{\deg I_e}$.

In the major part of our genus 2 experiments, $C_1 = 1$ or 2, see for example the next analysis and Section 5.3. In fact, for thousands of experiments in genus 2 to 4, C_1 only depends on the curve's type, *i.e.* on h_1, and is a power of 2. Further in the presentation we propose a conjecture to the value of this exponent. Proposition 6.12 tells that $C_1 \leq \dfrac{\deg I_e^2}{\deg I_e}$.

Proposition 6.16. *Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be a binary hyperelliptic curve of genus g defined over a field $\mathbb{F}$ of characteristic 2, and $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. Let L_h be the length of h_1. Define I and J as the ideals associated to the parametrization of $\mathscr{V}_{m,R}$, and $I_e = I \cap \mathbb{F}_{2^{dn}}[\mathbf{e}], J_e = J \cap \mathbb{F}_{2^{dn}}[\mathbf{e}]$. Then:*

$$\deg_w \mathbf{V}(J_e) = C_1 \cdot \frac{\deg \mathbf{V}(I_e)}{2^{m-g+L_h}}.$$

Proof. Since I_e and J_e are radical, we have $\deg_w \mathbf{V}(J_e) = \deg_w J_e$ and $\deg \mathbf{V}(I_e) = \deg I_e$. Thus it is enough to prove that the formulae are valid for the ideals. Recall that the number of squares among the coefficients of the Decomposition Polynomial associated to R is $k = g - L_h$. Renumber the equations defining $\mathscr{V}_{m,R}$ such that the squares are the k first equations. For $w = (w_1, \ldots, w_m)$ with $w_1 = \ldots = w_k = 1$ and $w_{k+1} = \ldots = w_m = 2$, we consider J_e in the graded algebra $(\mathbb{F}_{2^{dn}}[\mathbf{e}], (w_1, \ldots, w_m))$. Proposition 6.14 states that

$$\deg J_e = \frac{\deg I'}{2^{m-g+L_h}} = \frac{C_1 \cdot \deg I_e}{2^{m-g+L_h}}.$$

□

Experimentally C_1 is a power of 2 with exponent lesser than $m - g + L_h$, so the degree of $V(J_e)$ is indeed divided by a certain factor. We now turn to PDP_{ng} instances, *i.e.* $m = ng$. In the previous Section we showed that finding points in $\mathbf{V}(J_e)$ and $\mathbf{V}(I_e)$ are equivalent tasks, and this extends straightforwardly to the Weil Restrictions. Hence solving a PDP_m instance can be done by finding points in $\mathscr{W}_n(J_e)$, since every such point corresponds to a point in $\mathscr{W}_n(I_e)$, which in turn describes a decomposition of R. This has two benefits. First it should be faster to eliminate the variables $\mathbf{a}$ for J_e than for I_e because some quadratic equations have been replaced by linear ones. Another crucial advantage is that using $\mathscr{W}_n(J_e)$ leads to ideal of smaller degree for solving a PDP_{ng} instance.

Heuristic 6.17. *Assume that the base field is $\mathbb{F}_{2^{dn}}$ and that we have an element $t \in \mathbb{F}_{2^{kn}}$ such that $\{1, t, \ldots, t^{n-1}\}$ is a $\mathbb{F}_{2^d}$-basis of $\mathbb{F}_{2^{dn}}$, and write $e_i = \sum_{i=0}^{n-1} e_{ij} t^i$. The Weil Descent in the Summation approach, that is to say, cutting the Weil Restriction of I_e or J_e by the hyperplanes $e_{ij} = 0$ for $1 \leq i \leq ng$ and $1 \leq j \leq n-1$ gives a 0-dimensional ideal.*

This heuristic was always verified in our experiments. From now on, the results are stated assuming the heuristic is true.

Proposition 6.18. *Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be a genus g hyperelliptic curve defined over $\mathbb{F}_{2^{dn}}$. Let L_h be the length of h_1. A PDP_{ng} instance in $\mathrm{Jac}(\mathscr{H})$ can be solved by computing a lexicographical Gröbner Basis for an ideal $\mathbb{I}$ of degree:*

$$\deg_w \mathbb{I} = C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{(n-1)g+L_h}}.$$

Proof. With $m = ng$ we get from Proposition 6.16:

$$\deg_w \mathscr{W}_n(J_e) = C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{n((n-1)g+L_h)}}.$$

Let $1, t, \ldots, t^{n-1}$ be a $\mathbb{F}_{2^d}$-basis of $\mathbb{F}_{2^{dn}}$. The Weil Restriction $\mathscr{W}_n(J_e)$ is built by considering the change of variables $e_i = \sum_{j=0}^{n-1} e_{i,j} t^j$, for which the graduation extends. Thus let $w = (w_{1,1}, w_{1,2}, \ldots, w_{ng,n-1})$ with $w_{1,1} = w_{1,2} = \ldots = w_{k,n-1} = 1$ and $w_{k+1,1} = w_{k+1,2} = \ldots = w_{ng,n-1} = 2$, define the w-graded algebra $(\mathbb{F}_{2^d}[e_{1,1}, \ldots, e_{ng,n-1}], (w_{1,1}, \ldots, w_{ng,n}))$, and consider $\mathbf{I}(\mathscr{W}_n(J_e))$ as an ideal in this algebra. Solving the related PDP_{ng} instance is done geometrically by considering the intersection of graded hyperplanes

$$H = \bigcap_{\substack{1 \leq i \leq ng \\ 1 \leq j \leq n-1}} \mathbf{V}(e_{i,j}),$$

with $\deg_w \mathbf{V}(e_{i,j}) = 2$ for $g - L_h + 1 \leq i \leq ng, 1 \leq j \leq n-1$ and thus

$$\deg_w H = 2^{((n-1)g+L_h)(n-1)}.$$

Let now $\mathbb{I} = \mathbf{I}(\mathscr{W}_n(J_e) \cap H)$. Under Heuristic 6.17, it is 0-dimensional, so that the intersection has degree $\deg_w \mathbb{I} = \deg_w \mathscr{W}_n(J_e) \cdot \deg_w H$. The claim follows:

$$\begin{aligned}\deg_w \mathbb{I} &= C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{n((n-1)g+L_h)}} \cdot 2^{((n-1)g+L_h)(n-1)} \\ &= C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{(n-1)g+L_h}}.\end{aligned}$$

□

Remark 6.19. *If Conjecture 5.13 is true, then* $\deg V(I_e) = 2^{m-g}$ *and* $\deg \mathscr{W}_n(I_e) = 2^{n(n-1)g}$. *This happened in all our experiments. In this case we have from Proposition 6.16 that* $\deg_w \mathbf{V}(J_e) = C_1 \cdot 2^{-L_h}$, *and Proposition 6.18 then tells that* $\deg_w \mathbb{I} = C_1^n \cdot 2^{(n-1)^2 g - L_h}$.

6.2.3 Analysis for genus 2 curves

We checked over thousands of genus 2 curves (of all types) that C_1 was a power of 2 depending on the polynomial h_1 in the curve's equation. More precisely,

$$C_1 = \begin{cases} 1, \text{ if } \mathscr{H} \text{ is Type } I_b \text{ with } h_1(x) = x^2, \text{ Type } II, \text{ or Type } III \\ 2, \text{ if } \mathscr{H} \text{ is Type } I_b \text{ with } h_1(x) \neq x^2 \\ 4, \text{ if } \mathscr{H} \text{ is Type } I_a. \end{cases}$$

Roughly, the more squares there are among the coefficient of the Decomposition Polynomial, the closer $\deg I'$ is to $\deg I_e$ and C_1 is to 1. Recall that no squares appear among the Decomposition Polynomial's coefficients if the type is I_a, hence no reduction can be obtained this way. If we consider the other types of curves, and instantiate the formula of Proposition 6.18 for PDP_{2n} where $m = 2(n-1)$ and the non-reduced degree is $d_{\mathrm{Nag}} = 2^{2n(n-1)}$, we obtain the following degrees:

Type	C_1	L_h	$\deg \mathbb{I}$	Reduction factor
$I_b,\ h_1(x) \neq x^2$	2	1	$2^{(2n-1)(n-1)}$	2^{n-1}
$I_b,\ h_1(x) = x^2$	1	0	$2^{2(n-1)^2}$	$2^{2(n-1)}$
II or *III*	1	0	$2^{2(n-1)^2}$	$2^{2(n-1)}$

Table 6.2: List of degree reductions using Frobenius action on genus 2 curves

Higher genus To our knowledge, there is no known classification on higher genus binary hyperelliptic curves based on the shape of the h_1 polynomial. However, the degree of the squarefree part of h_1 equals the 2-rank of the Jacobian Variety of the curve, so that this gives a classification criteria in our situation where the shape of h_1 determines the shape of the equations. The following additional experiments in genus 3 (over thousands of curves) further confirmed our observation for the behaviour of C_1:

- For curves with $h_1(x) \in \{1, x, x^2, x^3\}$, we always observe $C_1 = 1$.
- For curves with h_1 a monic degree 2 polynomial with two distinct roots, we observe $C_1 = 2$; up to a linear change of variables, such polynomial have a shape $x(x+\alpha)$ for some α in the ground field, and verifies $L_h = 1$. If h_1 is monic of degree 2 and irreducible, we observe $C_1 = 4$, and $L_h = 2$.
- When h_1 is monic of degree 3 and split or has exactly one root in the base field, $C_1 = 4$; up to a linear change of variables, such polynomials have respectively a shape $x(x+\alpha)(x+\beta)$ or $(x^2+\alpha x+\beta)x$ for some α, β in the ground field, thus $L_h = 2$. When h_1 is monic and irreducible of degree 3, then $C_1 = 8$ with $L_h = 3$. Recall that there are no square among the coefficients of the R-Decomposition Polynomial if h_1 is irreducible.

A similar behaviour was identified for some cases in genus 4. Hence we propose the next Conjecture to sum up this Section:

Conjecture 6.20. *Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be an hyperelliptic curve of genus g defined over $\mathbb{F}_{2^{dn}}$. Assume h_1 is not irreducible of degree g, with length L_h. Then the degree ratio C_1 defined in Proposition 6.16 is a power of 2 that only depends on the polynomial h_1. More precisely, we have:*

$$C_1 = 2^{L_h}.$$

Using a Summation modelling, a PDP_{ng} instance on $\mathscr{H}$ can then be solved by computing a lexicographical Gröbner Basis of an ideal $\mathbb{I}$ of degree

$$\deg_w \mathbb{I} = 2^{(n-1)((n-1)g+L_h)}.$$

Remark 6.21. *If this Conjecture is true, then we find the following bounds for the first reduction step:*

$$2^{(n-1)^2 g} \leq \deg_w \mathbb{I} \leq 2^{(n-1)(ng-1)}.$$

Example in genus 2: Let $k = \mathbb{F}_{32} = \mathbb{F}_2[t]/\langle t^5+t^2+1\rangle$ and $K = \mathbb{F}_{32^2} = \mathbb{F}_{32}[\omega]/\langle \omega^2+t^4\omega+t\rangle$. Consider the type I_b curve given by $\mathscr{H} : y^2 + x^2 y = x^5 + (t^{21}\omega + t^{19})x + t^8\omega + t^5$. Fix $R = (x^2 + (t^{23}\omega + t^{17})x + t^{20}\omega + t^7, (t^4\omega + t^{14})x + t^{21}\omega + t^{25}) \in \mathrm{Jac}(\mathscr{H})$, and try to solve a PDP_4 instance related to R. Using the reduction process we just described, since $L_h = 0$ we expect $C_1 = 1$ and $I' = I_e$, and also a degree $2^{(n-1)^2 g} = 2^2 = 4$ system to solve at the end. The parametrization of $\mathscr{V}_{4,R}$ is

$$I = \begin{cases} (t^{20}\omega + t^7)a_1^2 + (t^5\omega + t^2)a_2^2 + e_4, \\ (t^{23}\omega + t^{17})a_1^2 + (t^9\omega + t^9)a_2^2 + e_3, \\ a_1^2 + a_1a_2 + (t^{23}\omega + t^{17})a_2^2 + t^{20}\omega + t^7 + e_2, \\ a_2^2 + a_2 + t^{23}\omega + t^{17} + e_1 \end{cases}$$

We compute a Gröbner basis for the ideal $I_e = I \cap K[e_1, \ldots, e_4]$ and find

$$\begin{aligned} \mathrm{GB}_{I_e} = \{ & e_3^3 + (t^{17}\omega + t^{15})e_4e_2^2 + (t^{14}\omega + 1)e_3e_2^2 + (t^{30}\omega + t^{15})e_4e_1^2 + t^{19}\omega e_3e_1^2 + (t^{27}\omega + t^3)e_3^2 + (t^{23}\omega + t^{26})e_2^2 \\ & + (\omega + t^7)e_1^2 + (t^{21}\omega + t^{23})e_4 + (t\omega + t^{14})e_3 + t^{30}\omega + t^{11}, \\ & e_4^2 + \omega e_3^2 + (t^{23}\omega + t^{21})e_1^2 + (t^{25}\omega + t^{19})e_4 + (t^{22}\omega + t^9)e_3 + t^{15}\omega + t^9, \\ & e_4e_3 + (t^{29}\omega + t^{25})e_3^2 + (t^{21}\omega + t^{19})e_2^2 + (t^3\omega + t^{19})e_1^2 + (t^2\omega + t^{16})e_4 + (t^{28}\omega + t^4)e_3 + t^3\omega + t^{26}\}. \end{aligned}$$

The degree of this ideal is 4, which is confirmed by the Hilbert Series

$$\mathrm{HS}_{I_e}(T) = \frac{(1-T^2)^2}{(1-T)^4} = \frac{1+2T+T^2}{(1-T)^2}.$$

Notice now that

$$\begin{aligned} N_4(a_1, a_2) &= (t^{20}\omega + t^7)a_1^2 + (t^5\omega + t^2)a_2^2 = ((t^8\omega + t^{21})a_1 + t^{16}\omega a_2)^2, \\ N_3(a_1, a_2) &= (t^{23}\omega + t^{17})a_1^2 + (t^9\omega + t^9)a_2^2 = ((t^{25}w + t^{23})a_1 + (t^{18}w + t^2)a_2)^2. \end{aligned}$$

Therefore, we define the ideal J by the equations

$$J = \begin{cases} (t^8\omega + t^{21})a_1 + t^{16}\omega a_2 + e_4, \\ (t^{25}w + t^{23})a_1 + (t^{18}w + t^2)a_2 + e_3, \\ a_1^2 + a_1a_2 + (t^{23}\omega + t^{17})a_2^2 + t^{20}\omega + t^7 + e_2, \\ a_2^2 + a_2 + t^{23}\omega + t^{17} + e_1 \end{cases}$$

We compute a Gröbner Basis for $J_e = J \cap K[e_1,e_2,e_3,e_4]$ for weights $w=(2,2,1,1)$ in w-DRL order and obtain:

$$\begin{aligned}\mathrm{GB}_{J_e} = \{&e_3^3+(t^{27}\omega+t^{14})e_3^2+(t^{22}\omega+t^{16})e_4e_2+(t^5\omega+t^{25})e_3e_2+(t^{13}\omega+t^{19})e_4e_1+(t^{23}\omega+t^8)e_3e_1\\ &+(t^{24}\omega+t^{10})e_4+(t^{14}\omega+t^{19})e_3+(t^{25}\omega+t^8)e_2+(t^{29}\omega+t^{16})e_1+t^{13}\omega+t^{10},\\ &e_4^2+(t^{29}\omega+t^{14})e_3^2+(t^{26}\omega+t^{24})e_4+(t^9\omega+t^{22})e_3+(t^{25}\omega+t^{19})e_1+t^{21}\omega+t^{19},\\ &e_4e_3+(t^{28}\omega+t^6)e_3^2+(t^{30}\omega+t^{30})e_4+(t^{12}\omega+t^{30})e_3+(t^{24}\omega+t^{18})e_2+(t^{15}\omega+t^{21})e_1+t^{15}\omega+t^{14}\},\end{aligned}$$

Using Magma, we find the Hilbert series as $\mathrm{HS}_{J_e}(T)=\dfrac{1}{(1-T)^2}$, and confirm that $\deg_w J_e = 1 = \dfrac{\deg I_e}{4}$: the degree of I_e has been divided by the product of the weights. Let $\mathrm{GB}_{I_e} = \{f_1,f_2,f_3\}$, let $\mathrm{GB}_{J_e}=\{g_1,g_2,g_3\}$ and define $I' = \{g_i^\sigma(e_1,e_2^2,e_3^2,e_4) \ : \ 1\le i\le 3\}$. Observe that $g_i^\sigma(e_1^2,e_2^2,e_3,e_4)=f_i$, hence $I'=I_e$ and $C_1=1$ as expected.

We now finish the solving of the PDP_4 instance, collecting all elements in GB_{J_e} wrt. ω. We obtain 6 equations

$$\mathbb{I} = \begin{cases} t^{22}e_4e_2+t^{13}e_4e_1+t^{24}e_4+t^{27}e_3^2+t^5e_3e_2+t^{23}e_3e_1+t^{14}e_3+t^{25}e_2+t^{29}e_1+t^{13},\\ t^{16}e_4e_2+t^{19}e_4e_1+t^{10}e_4+e_3^3+t^{14}e_3^2+t^{25}e_3e_2+t^8e_3e_1+t^{19}e_3+t^8e_2+t^{16}e_1+t^{10},\\ t^{26}e_4+t^{29}e_3^2+t^9e_3+t^{25}e_1+t^{21},\\ e_4^2+t^{24}e_4+t^{14}e_3^2+t^{22}e_3+t^{19}e_1+t^{19},\\ t^{30}e_4+t^{28}e_3^2+t^{12}e_3+t^{24}e_2+t^{15}e_1+t^{15},\\ e_4e_3+t^{30}e_4+t^6e_3^2+t^{30}e_3+t^{18}e_2+t^{21}e_1+t^{14}\end{cases}$$

For a weighted lexicographical order $e_4>e_3>e_2>e_1$, we find the following Gröbner Basis

$$\begin{aligned}\mathrm{GB}_{lex,\mathbb{I}} = \{&e_4+t^{18}e_1^3+t^{12}e_1^2+t^3e_1+t^{20},\\ &e_3+t^{26}e_1^3+t^{17}e_1^2+t^2e_1+t^{27},\\ &e_2+t^7e_1^3+t^{21}e_1^2+t^{22}e_1+t^{29},\\ &e_1^4+t^2e_1^3+t^{30}e_1^2+t^{26}e_1+t^{28}\end{aligned}$$

so $\deg_w \mathbb{I}=4$ as expected, and we obtain the solutions $\{(t^{30},t^{11},t^{25},t^{19}),(t^{23},t^{20},t^7,t^{29})\}$ for (e_1,e_2,e_3,e_4). This means we have two elements in $V(I_e)$ as $\{(t^{30},t^{11},(t^{25})^2=t^{19},(t^{19})^2=t^7),(t^{23},t^{20},(t^7)^2=t^{14},(t^{29})^2=t^{27})\}$ that can be used to check for a decomposition of R. This leads to polynomials

$$\begin{aligned}F_1(x) =& x^4+t^{30}x^3+t^{11}x^2+t^{19}x+t^7,\\ F_2(x) =& x^4+t^{23}x^3+t^{20}x^2+t^{14}X+t^{27}.\end{aligned}$$

The second has only one roots over k, but $F_1(x)=(x+t^7)(x+t^{17})(x+t^{20})(x+t^{25})$, and we recover a decomposition in 4 points for R. In the next section we achieve further reduction. This example will be used again to illustrate that a degree 2 ideal can even be considered.

6.2.4 Additional reduction using the univariate coefficient

We let $\mathscr{H}$ be a hyperelliptic curve defined over $\mathbb{F}_{2^{kn}}$. A fixed $R(u,v)\in \mathrm{Jac}(\mathscr{H})$ of weight g is given, and we write $u=x^g+u_1x^{g-1}+\dots$. The associated Decomposition Polynomial is $F(x)=x^m+\sum_{i=0}^{m-1}N_i(\mathbf{a})x^i$. For simplicity we assume that m is even, the arguments being similar in the odd case. With $d=m-g$, we have $N_1(\mathbf{a})=N_1(a_d)=a_d^2+H_ga_d+u_1$ from Section 6.1. Therefore the first equation in the parametrization of $\mathscr{V}_{m,R}$ is

$$e_1 = a_d^2+H_ga_d+u_1.$$

This particular equation means that it is equivalent to find values for e_1 or to find values for a_d. Since e_1 appears nowhere else in the defining equations for $\mathscr{V}_{m,R}$, we can eliminate it instead of a_d to obtain "tweaked" Specialized Summation Sets. Keeping notations of the previous Section, if we define

$$I_a = I \cap \mathbb{F}_{2^{kn}}[a_d, e_2, \ldots, e_{ng}],\ J_a = J \cap \mathbb{F}_{2^{kn}}[a_d, e_2, \ldots, e_{ng}],$$

then it is equivalent to find points in $\mathbf{V}(I_e)$, $\mathbf{V}(I_a)$, $\mathbf{V}(J_e)$ or $\mathbf{V}(J_a)$. In practice, working with $\mathbf{V}(I_a)$ or $\mathbf{V}(J_a)$ means we can omit the equation $e_1 = N_1(\mathbf{a})$ in $\mathscr{V}_{m,R}$'s definition, which has the benefit that it is usually faster to compute elimination ideal for smaller orders and with less variables. Additionally, this idea also leads to an additional reduction in the degree of a PDP_{ng} system to solve. When building the Weil Restriction $\mathscr{W}_n(J_a)$ or $\mathscr{W}_n(I_a)$, we set $a_d = \sum_{i=0}^{n-1} a_{d,i} t^i$ for some $\mathbb{F}_{2^k}$-basis $1, t, \ldots, t^{n-1}$ of $\mathbb{F}_{2^{kn}}$. Following Section 6.1 we can find with very high probability values $a^*_{d,1}, \ldots, a^*_{d,n-1}$ by solving a system of degree 2^{n-1}. Geometrically, we then intersect the Weil Restriction with the hyperplanes $\mathbf{V}(e_{i,j})$ for $k+1 \leq i \leq m$ and $1 \leq j \leq n-1$, and with $V(a_{d,j} + a^*_{d,j})$ instead of $V(e_{1,j})$ for $1 \leq j \leq n-1$. The situation now splits in two cases whether $H_g = 0$ or not.

When $\mathbf{H_g \neq 0}$: Here $N_1(\mathbf{a})$ is not a square so there is a natural weight 2 on e_1, hence there should be a weight 2 on the $e_{1,j}$ for $1 \leq j \leq n-1$. However those variables have been replaced by the $a_{d,i}$ with weight 1. Consequently, the intersection H of all the needed hyperplanes has degree

$$\deg_w H = 2^{(n-1)(m-g+L_h-1)}.$$

It is straightforward to adapt the proofs of Proposition 6.14 and Proposition 6.16 to show that, in the general case of the parametrization of $\mathscr{V}_{m,R}$:

$$\deg_w \mathbf{V}(J_a) = \frac{C_1 \cdot \deg I_a}{2^{m-g+L_h}}.$$

Now consider a PDP_{ng} context and define $\mathbb{I}_2$ as the ideal associated to $\mathscr{W}_n(J_a) \cap H$, generally of dimension 0. Since $\deg I_e = \deg I_a$, then $\deg \mathscr{W}_n(J_a) = \deg \mathscr{W}_n(J_e)$ and Proposition 6.18 gives:

$$\begin{aligned}
\deg_w \mathbb{I}_2 &= \deg_w \mathscr{W}_n(J_a) \cdot \deg_w H \\
&= C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{n((n-1)g+L_h)}} \cdot 2^{(n-1)((n-1)g+L_h-1)} \\
&= C_1^n \frac{\deg \mathscr{W}_n(I_e)}{2^{(n-1)g+L_h+(n-1)}} \\
&= \frac{\deg_w \mathbb{I}}{2^{n-1}},
\end{aligned}$$

thus an additional factor of 2^{n-1} has been obtained.

Remark 6.22. *If both Conjecture 5.13 and 6.20 are true, then $C_1^n = 2^{nL_h}$, leading to $\deg_w \mathbb{I} = 2^{(n-1)((n-1)g+L_h)}$, and finally $\deg_w \mathbb{I}_2 = 2^{(n-1)((n-1)g+L_h-1)}$.*

When $\mathbf{H_g = 0}$: $N_1(\mathbf{a})$ is a square $a_d^2 + \lambda^2$ for some $\lambda \in \mathbb{F}_{2^{kn}}$. Let $z \in \mathbb{F}_{2^{kn}}$. For any $g \in \mathbb{F}[\mathbf{e}]$ we denote by g_z the polynomial $g(e_1 + z, e_2, \ldots, e_{ng})$ and for any $g \in \mathbb{F}[a_d, e_2, \ldots, e_{ng}]$ we denote by g_z the polynomial $g(a_d + z, e_2, \ldots, e_{ng})$.

Lemma 6.23. *Let $I_a = I \cap \mathbb{F}[a, e_2, \ldots, e_m]$ and $J_a = J \cap \mathbb{F}[a_d, e_2, \ldots, e_m]$.*

1. $g \in J_a \Rightarrow ((g_\lambda)^\sigma)_{\lambda^2}(a_d^2, e_2, \ldots, e_k, e_{k+1}^2, \ldots, e_m^2) \in I_a$.
2. $g \in I_a \Rightarrow (((g^\sigma)_{\lambda^2})^{\sigma^{-1}})_\lambda(a_d, e_2^2, \ldots, e_k^2, e_{k+1}, \ldots, e_m) \in J_a$.

Proof. Let $I_e = I \cap \mathbb{F}[\mathbf{e}]$ and $J_e = J \cap \mathbb{F}[\mathbf{e}]$.

1. Because of the relation $e_1 = a_d + \lambda$ in J, we have $g \in J_a \Rightarrow g(e_1+\lambda, e_2, \ldots, e_m) = g_\lambda \in J_e$. Lemma 6.11 states that $(g_\lambda)^\sigma(e_1, \ldots, e_k, e_{k+1}^2, \ldots, e_m^2) \in I_e$, and the result is obtained using the relation $e_1 = a_d^2 + \lambda^2$ in I.
2. If $g \in I_a$, then $g^2 = g^\sigma(a_d^2, e_2^2, \ldots, e_m^2) \in I_a$, and thus $g^\sigma(e_1+\lambda^2, e_2^2, \ldots, e_m^2) = (g^\sigma)_{\lambda^2}(e_1, e_2^2, \ldots, e_m^2) \in I_e$. Using Lemma 6.11 we obtain that $(g^\sigma)_{\lambda^2}(e_1^2, e_2^4, \ldots, e_k^4, e_{k+1}^2, \ldots, e_m^2) \in J_e$. This leads to $((g^\sigma)_{\lambda^2})^{\sigma^{-1}}(e_1, e_2^2, \ldots, e_k^2, e_{k+1}, \ldots, e_m) \in J_e$ and the result follows using the relation $e_1 = a_d + \lambda$ in J.

□

For $g_1 \ldots, g_r$ a Gröbner Basis of J_a, let $I'' = \left\langle \; (((g_i^\sigma)_{\lambda^2})^{\sigma^{-1}})(a_d^2, e_2, \ldots, e_k, e_{k+1}^2, \ldots, e_m^2) \;:\; 1 \le i \le r \right\rangle$. Lemma 6.23 says that I'' is a subideal of I_a. More precisely, let $w_1 = 2 = w_{k+1} = \ldots = w_m$ and $w_2 = w_3 = \ldots = w_k = 1$, and consider the injective homomorphism of graded algebra

$$\begin{array}{rccc} \varphi: & (\mathbb{F}_{2^{kn}}[Y_1, \ldots, Y_m], (w_1, \ldots, w_m)) & \longrightarrow & (\mathbb{F}_{2^{kn}}[X_1, \ldots, X_m], (1, \ldots, 1)) \\ & Y_i & \longmapsto & X_i^{w_i} \end{array}$$

to see that $I'' = \varphi(J_a)$. With a similar proof as Proposition 6.14 it can then be shown that there is constant C_2 such that

$$\begin{aligned} \deg_w J_a = \frac{\deg J'}{2^{m-g+L_h+1}} &= \frac{C_2 \deg I_a}{2^{m-g+L_h+1}} \\ &= \frac{C_2 \deg I_e}{2^{m-g+L_h+1}} \end{aligned}$$

and thus $\deg \mathscr{W}_n(J_a) = C_2^n \cdot 2^{n(-L_h-1)}$. If we write $a_d = \sum_{i=0}^{n-1} a_{d,j} t^j$, the hyperplanes $V(a_{d,j} + a_{d,j}^*)$ have degree 2 because the weight on a_d transfers to the variables $a_{d,j}$. Hence the variety H has degree $\deg_w H = 2^{(n-1)(m-g+L_h+1)}$. Turning to PDP_{ng} context, the ideal $\mathbb{I}_2$ associated to the 0-dimensional intersection $\mathscr{W}_n(J_a) \cap H$ has degree

$$\begin{aligned} \deg_w \mathbb{I}_2 &= C_2^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{n((n-1)g+L_h+1)}} \cdot 2^{(n-1)((n-1)g+L_h+1)} \\ &= C_2^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{(n-1)g+L_h-1}} \\ &= \frac{C_2^n}{C_1^n} \cdot \frac{\deg_w \mathbb{I}}{2} \end{aligned}$$

Remark 6.24. *Assuming $C_2 = C_1$, which happened in all our experiments, then only a factor 2 is obtained. If moreover Conjecture 5.13 and 6.20 are true, then $\deg_w \mathbb{I}_2 = 2^{(n-1)^2 g + L_h - 1}$.*

Analysis in genus 2 If $H_g \neq 0$, we are dealing with Type I_a and I_b curves. The previous paragraph shows that a reduction of the degree by 2^{n-1} can be added to the reduction of Proposition 6.18. If $H_g = 0$, then the curve is Type *II* or *III*. In this setting, $L_h = 0$ and we observed that $C_2 = C_1 = 1$. In a PDP_{2n} context, $m = ng$. Let d_{Sum} be the degree of the final system to solve a given PDP_{2n} instance using a Summation modelling. We have

$$d_{\mathrm{Sum}} = \begin{cases} 2^{(n-1)(2n-1)} \text{ if } \mathscr{H} \text{ is Type } I_a, \\ 2^{(n-1)(2n-2)} \text{ if } \mathscr{H} \text{ is Type } I_b \text{ with } h_1(x) \neq x^2, \\ 2^{(n-1)(2n-3)} \text{ if } \mathscr{H} \text{ is Type } I_b \text{ with } h_1(x) = x^2, \\ 2^{2(n-1)^2-1} \text{ if } \mathscr{H} \text{ is Type } II \text{ or Type } III. \end{cases}$$

Next we list the degree reduction using a Summation modelling for $n = 2, 3, 4$ depending on the curve's type, just as in Section 6.1.3. In order to compare all the reductions we achieved,

we also list the degrees d_{Nag} of the classical Nagao approach and the refined d_{Ref} from Table 6.1 that we obtained in Section 6.1.

Table 6.3: Degree reduction in genus 2 for small extension fields using a Summation Modelling

Type	$\deg h_1$	H_g	C_1	n	d_{Sum}	d_{Nag}	d_{Ref}
I_a	2	1	4	2	8	16	8
				3	1024	4096	1024
				4	2^{21}	2^{24}	2^{21}
I_b	2	1	2	2	4	16	4
				3	256	4096	256
				4	2^{18}	2^{24}	2^{18}
I_b with $h_1(x)=x^2$	2	1	1	2	$2=d_{\text{opt}}$	16	$2=d_{\text{opt}}$
				3	$64=d_{\text{opt}}$	4096	$64=d_{\text{opt}}$
				4	$2^{15}=d_{\text{opt}}$	2^{24}	$2^{15}=d_{\text{opt}}$
II	1	0	1	2	$2=d_{\text{opt}}$	16	$2=d_{\text{opt}}$
				3	128	4096	$64=d_{\text{opt}}$
				4	2^{17}	2^{24}	$2^{15}=d_{\text{opt}}$
III	0	0	1	2	$2=d_{\text{opt}}$	16	$2=d_{\text{opt}}$
				3	128	4096	$64=d_{\text{opt}}$
				4	2^{17}	2^{24}	$2^{15}=d_{\text{opt}}$

Example in genus 2, continued: We use the same notations and curve as the previous example. Recall that the ideal J is defined by

$$J=\begin{cases}(t^8\omega+t^{21})a_1+t^{16}\omega a_2+e_4,\\ (t^{25}w+t^{23})a_1+(t^{18}w+t^2)a_2+e_3,\\ a_1^2+a_1a_2+(t^{23}\omega+t^{17})a_2^2+t^{20}\omega+t^7+e_2,\\ a_2^2+a_2+t^{23}\omega+t^{17}+e_1.\end{cases}$$

This time we compute a Gröbner Basis for the ideal $J_a=J\cap K[a_2,e_2,e_3,e_4]$ with weights $w=(1,2,1,1)$ in w-DRL order, and find

$$\begin{aligned}\text{GB}_{J_a}=\{&e_3^2+(t^{25}w+t^{23})e_3a_2+(t^{14}w+t^{20})a_2^2+(t^{23}w+t^{17})e_2+t^{18}w+t,\\ &e_4+(t^{28}w+t^3)e_3+(t^{26}w+t^{24})a_2\}.\end{aligned}$$

From this we obtain that $\deg_w J_a=1$, as confirmed by the Hilbert Series $\text{HS}_{J_a}(T)=\dfrac{1}{(1-T)(1-T^2)}$.
First, we write $a_2=a_{2,0}+a_{2,1}\omega$ and find a value for $a_{2,1}$ using the equation $N_1(a_2)=e_1$. This leads to $t^4a_{2,1}^2+a_{2,1}+t^{23}=(a_{2,1}+t^9)(a_{2,1}+t^{10})=0$. Let $a_{2,1}^*=t^{10}$. We build the ideal $\mathbb{I}_2$ by Weil Restriction, *i.e.* evaluating elements of GB_{J_a} at $(a_{2,0}+a_{2,1}^*\omega,e_2,e_3,e_4)$ and collecting wrt. w, to obtain:

$$\mathbb{I}_2=\begin{cases}t^{25}e_3a_{2,0}+t^{29}e_3+t^{23}e_2+t^{14}a_{2,0}^2+t^3,\\ e_3^2+t^{23}e_3a_{2,0}+t^5e_3+t^{17}e_2+t^{20}a_{2,0}^2+t^{24},\\ t^{28}e_3+t^{26}a_{2,0}+t^{30},\\ e_4+t^3e_3+t^{24}a_{2,0}+t^6\end{cases}$$

and a lexicographical basis with $e_4>e_3>e_2>a_2$ is:

$$\begin{aligned}\text{GB}_{lex,\mathbb{I}_2}=\{&e_4+t^{13}a_2+t^{23},\\ &e_3+t^{29}a_2+t^2,\\ &e_2+t^{17}a_2+t^{18},\\ &a_{2,0}^2+t^{10}a_2+t^{22}\},\end{aligned}$$

an ideal of (weighted) degree 2 as expected. We find solutions $\{(t^{16},t^{11},t^{25},t^{19}),(t^{6},t^{20},t^{7},t^{29})\}$ for $(a_{2,0},e_2,e_3,e_4)$, recover two values for e_1:

$$e_1^* = (t^{16})^2 + t^{16} + (t^4\omega + t){a_{2,1}^*}^2 + \omega a_{2,1}^* + t^{23}\omega + t^{17} = t^{30},$$
$$e_1^{**} = (t^{6})^2 + t^{6} + (t^4\omega + t){a_{2,1}^*}^2 + \omega a_{2,1}^* + t^{23}\omega + t^{17} = t^{23},$$

from which we build back the solutions found in the previous example.

6.3 Comparisons of Modellings

From Table 6.3 we observe that Nagao and Summation modelling are equivalent for Type I_a and I_b, and that the refined Nagao approach is slightly better for Type II and III. Moreover, it is very easy to build the system with reduced degree in Nagao's modelling: basically only square roots of some of the defining equations have to be computed. A last advantage is that Nagao's refinement holds for any hyperelliptic curve of any genus, with precise bounds on the degree reduction process. For Summation Modelling, only the genus 2 case is totally described — even if the result seems to extend to higher genus hyperelliptic curves. Furthermore, it is harder to build the system using a Summation modelling, as a Gröbner basis for a well-chosen elimination order has to be computed. The order on the variables themselves may also have a practical impact on the running time. Overall, Nagao's approach seems more stable and the easiness of the reduction process makes it more interesting at first sight. Nevertheless, in this last Section we compare the practical running time for all modellings. We conclude with an Index-Calculus simulation on a realistic genus 2 curve of Type II and Type I_b with $h_1(x) = x^2$, satisfying a generic security bound of approximately 2^{92} and 2^{93} operations respectively.

6.3.1 Nagao vs Summation in Odd characteristic

A first natural question is to compare the methods in odd characteristic, even if there are no known improvements at the moment. Experiments were done on $\mathbb{F}_{q^n}$ with $\log q = 16$, $n = 2, 3$, and imaginary genus 2 curves given by general Weierstrass equations $y^2 = h(x)$. This means we look for decompositions of a given R of size $2n$. For each approach we listed the time needed to build the system, to compute a Degree Order basis, then to obtain a lexicographical basis with FGLM. For Summation modelling, building the system means computing a Specialized Summation Set for a given R of weight 2, that is to say, eliminating variables from a parametrization of the corresponding $\mathscr{V}_{m,R}$. We used Magma 2.19 [BCP97] for those experiments, so that a DRL Gröbner basis and an elimination basis are computed with $F4$, on the same computer as the previous experiments of this article.

Table 6.4: Comparisons of Nagao and Summation modelling in odd characteristic

n	Degree	Method								Ratio
		Nagao				Summation				
		System	DRL	FGLM	Total	System	DRL	FGLM	Total	
2	16	-	0.001s.	0.001s.	**0.002s.**	0.005s.	0.004s.	0.001	0.010	5
3	4096	-	159s.	1254s.	**1413s.**	137.6s*	2280s.	7358s.	9775s.	6.9

For $n = 2$, both approaches are extremely fast and of comparable speed. Therefore timings of this row are averaged over thousands of tests, for several curves. For $n = 3$, we stress that a well planned computing strategy had to be designed to obtain Summation systems in reasonable time. Indeed, eliminating without care the variables to compute $\mathbb{S}_{6,R}$ takes more than 116000 sec. We avoided this very long computation by eliminating only 3 variables in two steps, computing a basis for weighted degree order — this is higlighted by a star in the table. The system is then solved with the standard strategy. Even if we do not count this time for Summation modelling and assume a symbolic Specialized Summation set is given as raw input, we see that Nagao's modelling is faster by a ratio of nearly 7. This may be explained by the degree of the defining equations obtained in Summation modelling. Nagao's approach always gives as much quadratic equations as variables, whereas Summation's approach needs less variables but gives equations of larger degree. The above table shows that, in odd characteristic, Nagao's modelling is the most efficient.

6.3.2 Nagao vs Summation for binary genus 2 curves

Here we focus on fields $\mathbb{F}_{2^{nd}}$ with $d = 15$, $n = 3$, and curves of type I_b with $h_1(x) = x^2$ as well as curves of type II with $h_1(x) = x$. This choice is made because these are curves where $d_{\text{opt}} = 64$ can be reached for both modelling, as observed in Tables 6.1 and 6.3. For $n = 2$, the systems have degree 2 after the degree reduction. In particular a symbolic lexicographical Gröbner Basis could be precomputed, then solved for each new R. Therefore we did not consider this very simple case. To show the impact of the degree reduction we also give timings for "Old" approaches, that is to say, Nagao or Summation modelling without any degree reduction. Headings "Method" refer to Nagao or Summation approach. For each of those rows, the upper subrow gives the timing for "Old" approach and the lower subrow gives timing for the new Reduced approach. "Style Ratio" is obtained by comparing Old and Reduced approaches, and "Method Ratio" by comparing Reduced Nagao and Reduced Summation. In the first column, d_{old} stands for the degree of the system obtained with the old approach, while d_{red} stands for the new reduced degree.

Table 6.5: Comparisons of Nagao and Summation modelling in even characteristic

Curve	Method	System	DRL	FGLM	Total	Style Ratio	Method Ratio
Type I_b, $h_1(x) = x^2$, $d_{old} = 4096$, $d_{red} = 64$	Nagao	-	166.76s.	34152s. !!	34318s. !!	$1.7 \cdot 10^6$	**17**
		-	0.02s.	0.000s.	0.02s.		
	Summation	1.04s.	0.9s.	8.7s.	10.64s.	31	
		0.27s.	0.06s.	0.01s.	0.34s.		
Type II, $h_1(x) = x$, $d_{old} = 4096$, $d_{red} = 64$	Nagao	-	185.56s.	33917s. !!	34102s !!	$1.1 \cdot 10^6$	**14**
		-	0.02s.	0.009s.	0.029s.		
	Summation	0.84s.	0.65s.	7.7s.	9.19s.	23	
		0.27s.	0.14s.	0.01s.	0.42s.		

The timings highlighted by exclamations marks are abnormaly long. Since, once computed, the lexicographical bases are not in Shape position, this suggests a problem in Magma 2.19 implementation[2] of FGLM, as it should be faster to compute a lexicographical basis not in Shape position than a basis in Shape position. To obtain a fairer comparison, we estimated the running time of FGLM on random systems over $\mathbb{F}_{2^{15}}$ with $n(n-1)g = 12$ quadratic equations in $n(n-1)g = 12$ variables. The running time of FGLM for such systems (usually in Shape Position) is around 1500sec. If we consider this time as a reference for the Old Nagao approach, the speed-up ratio obtained by the Reduced approach is around 75000. Computational strategies were used to compute Specialized Summation Sets. The elimination Basis was computed for a weighted order, in two steps: of the 4 variables to be eliminated, three are eliminated in a first step, then the last is eliminated. This strategy leads to important speed-ups in our experiments for the elimination. Table 6.5 shows that Refined Nagao's modelling is also practically faster than the Refined Summation Modelling. For the next and final Section of this article, we therefore used a Refined Nagao's approach to solve PDP instances.

6.3.3 Running time of DLP solving for a realistic binary genus 2 curves

Let ω such that $\omega^{31} + \omega^3 + 1 = 0$ and $\mathbb{F}_{2^{31}} \simeq \mathbb{F}_2[\omega]$, and let t be such that $t^3 + \alpha t + \beta = 0$ with $\alpha = 7\text{BCEB1AC}$ and $\beta = 50\text{F6CCC4}$ in hexadecimal form, that is to say, the hexadecimal representation of α and β when ω is evaluated at 2, and put $\mathbb{F}_{2^{93}} = \mathbb{F}_{2^{31\cdot 3}} \simeq \mathbb{F}_{2^{31}}[t]$. We solve PDP_6 instances using our refined Nagao modelling.

[2] We did not try a more recent version.

Type II curve: Let $\mathscr{H} : y^2 + xy = x^5 + f_3x^3 + x^2 + f_0$, with

$$\begin{aligned} f_3 &= \text{A814B6C09256168AC93ABA1}, \\ f_0 &= \text{16400CBCC65A5EE5F67165AC}, \\ \#\mathscr{H}(\mathbb{F}_{2^{93}}) &= 9903520314283080096056319534 \geq 2^{93}, \end{aligned}$$

with the same encoding convention. Using the Magma2.19 implementation of Vercauteren's version [Ver02] of Kedlaya's algorithm for counting points, it takes approximately 24 seconds to check that its Jacobian Variety has order

$$\#\text{Jac}(\mathscr{H}) = 2 \times 3 \times 163466191025695437078816673032209936431423731074319386 53,$$

which is nearly prime. Its larger prime factor is a 184 bits number, hence a generic attack method would need around 2^{92} operations.

We start by counting (with Magma) the elements in the factor base $\mathscr{B} = \{P : P \in \mathscr{H}, x(P) \in \mathbb{F}_{2^{31}}\}$ and find a set with cardinal a number of 31 bits ; its enumeration can be parallelized easily. For example, with 8000 cores, each can enumerate on a subset of size $2^{31}/8000 \approx 2^{19}$ of a partition of $\mathbb{F}_{2^{31}}$. A single Intel®Xeon®@2.93GHz cpu needs roughly 40 sec. to complete its part of the enumeration. Now the systems coming from the univariate polynomial can be symbolically solved by hand. If we write $R = (x^2 + u_1x + u_0, v_1x + v_0) = (u_1, u_0, v_1, v_0)$, then $N_1(a_4) = a_4^2 + u_1 = (a_4 + \sqrt{u_1})^2$. Because the Frobenius automorphism fixes every subfield, $N_1(a_4) \in \mathbb{F}_{2^{31}} \Leftrightarrow a_4 + \sqrt{u_1} \in \mathbb{F}_{2^{31}}$. Hence if we let $a_4 = a_{4,0} + a_{4,1}t + a_{4,2}t^2$ and $\sqrt{u_1} = u'_{1,0} + u'_{1,1}t + u'_{1,2}t^2$ then we have

$$N_1(a_4) \in \mathbb{F}_{2^{31}} \Leftrightarrow a_{4,i} = u'_{1,i}, \ 1 \leq i \leq n.$$

Hence those values are directly obtained once an input R is given. It is even possible to precompute a symbolic unsquared system $\mathscr{S}_2$ with $a_{4,1}, a_{4,2}, u_1$ and u_0 as parameters. After this, the harvesting of relations is started. Each new $R \in \text{Jac}(\mathscr{H})$ to decompose is computed using a pseudo-random walk as proposed by Gaudry [Gau00]. If it is not of weight 2, then it is discarded and a new one is computed. The symbolic unsquared system $\sqrt{\mathscr{S}_2}$ is then evaluated at coordinates of R and corresponding values for $a_{4,1}, a_{4,2}$, following Section 6.1. The resulting system has resp. 4 (resp. 6) linear (resp. quadratic) equations in 10 variables, and is solved following the standard strategy for 0-dimensional systems:

- a DRL Gröbner Basis for $\sqrt{\mathscr{S}_2}$ is computed in $3.87 \cdot 10^{-4}$sec, using code generating techniques and F5 [Fau02] algorithm. We can check that $\sqrt{\mathscr{S}_2}$ has 64 solutions.
- With Sparse-FGLM algorithm [FM11], we obtain indeed a univariate polynomial of degree 64 in $5.93 \cdot 10^{-4}$sec.
- The last step of the solving process is to find its roots using NTL [Sho05]. This is done in $2.22 \cdot 10^{-3}$sec.

Overall, solving one PDP_6 instance over $\mathscr{H}$ takes $3.2 \cdot 10^{-3}$sec., and finding the roots of the degree 64 univariate polynomial surprisingly becomes the bottleneck of the computation. Memory-wise the whole process is really efficient as approximately 1.1 Mo is needed. The probability to get a decomposition for each R is $1/6!$, so we need in average $720 \times 3.2 \cdot 10^{-3}$sec.$= 2.3$ sec. to find a relation. The factor base has approximately 2^{31} elements and is invariant by the canonical involution on $\mathscr{H}$, we would normally need around $2^{31}/2 = 2^{30}$ relations to start linear algebra. However, computing at least twice this minimal number of relations enables us to use efficient filtering techniques [Bou13, Tea15] to reduce the size of the matrix. Computing more relations can lead to even more efficient filtering.

Using 8000 cores, the harvesting phase can be completed in a bit more than 7 days. The filtering is then performed and can reduce the size from 2^{31} to 250 millions rows (around 2^{28}) with 87 non-zero elements per row in average. The size of the matrices were obtained from a personal communication with Jean-Charles Faugère. It is noteworthy to state that the filtering algorithms considered are normally dedicated to factoring integers and discrete logarithm over finite fields; better reduction might be expected with a filtering step designed for a hyperelliptic curve DLP. A sparse linear algebra algorithm, usually a block Wiedemann, is expected to run in 2^{56} operations. As a comparison, in the record factorizations of a RSA-768 modulus [KAF+10] the matrix used was near 193 millions rows with 144 non-zero elements in average. In the factorization of a 1061 bits number [Chi12], the matrix had around 282 millions rows, and the filtering-merge phase reduced it to 93 millions, averaging 125 non-zero elements by row. We also mention that the final algebra step in RSA has to be performed over $\mathbb{F}_2$.

Conclusion: This practical simulation confirms that characteristic 2 curves are weaker than their odd characteristic counterparts in general. This strenghtens that curves based cryptographic standards should now focus on odd characteristic. In particular, we highlighted that, on a binary genus 2 curves defined over extensions whose degree admits a factor of 2 or 3, an efficient harvesting phase can be designed. Indeed, we showed that, using 8000 cores, around 1 week is needed to build an overdetermined matrix for a curve satisfying a generic bound of 2^{92}. The degree reduction is linked to the length of the polynomial h_1 defining the curve. The shorter h_1 is, the more efficient the arithmetic can be, but the more vulnerable the curve is to decomposition attacks. Therefore extensions whose degree admit a small factor should in general be avoided for curves with short h_1.

Bibliography

[ADH99] L. M. Adleman, J. DeMarrais, and M. A. Huang. A subexponential algorithm for discrete logarithms over hyperelliptic curves of large genus over GF(q). *Theor. Comput. Sci.*, 226(1-2):7–18, 1999. 7, 8, 9, 61, 69, 72

[Ari03] S. Arita. An addition algorithm in Jacobian of $\mathscr{C}_{\mathrm{ab}}$ curves. *Discrete Applied Mathematics*, 130(1):13–31, 2003. 73

[Bar04] M. Bardet. *Étude des systèmes algébriques surdéterminés. Applications aux codes correcteurs et à la cryptographie.* PhD thesis, Université Pierre et Marie Curie - Paris VI, December 2004. 34, 35, 36

[Bas03] A. Basiri. *Bases de Groebner et LLL. Arithmétique rapide des courbes* C_{ab}. PhD thesis, 2003. 73

[BBB+12] E. Barker, W. Barker, W. Burr, W. Polk, and M. Smid. Recommandation for key management, part 1: General (revision 3). *Computer Security*, 2012. 6

[BBJ+08] D. J. Bernstein, P. Birkner, M. Joye, T. Lange, and C. Peters. Twisted Edwards curves. In *Progress in Cryptology - AFRICACRYPT 2008, First International Conference on Cryptology in Africa, Casablanca, Morocco, June 11-14, 2008. Proceedings*, pages 389–405, 2008. 59

[BCHL16] J. W. Bos, C. Costello, H. Hisil, and K. E. Lauter. Fast cryptography in genus 2. *J. Cryptology*, 29(1):28–60, 2016. 8, 9, 15, 49, 57

[BCKL15] D. J. Bernstein, C. Chuengsatiansup, D. Kohel, and T. Lange. Twisted Hessian curves. In *Progress in Cryptology - LATINCRYPT 2015 - 4th International Conference on Cryptology and Information Security in Latin America, Guadalajara, Mexico, August 23-26, 2015, Proceedings*, pages 269–294, 2015. 60

[BCP97] W. Bosma, J. J. Cannon, and C. Playoust. The Magma algebra system I: the user language. *J. Symb. Comput.*, 24(3/4):235–265, 1997. 8, 10, 34, 91, 100, 133

[BD04] B. Byramjee and S. Duquesne. Classification of genus 2 curves over $\mathbb{F}_{2^n}$ and optimization of their arithmetic. *IACR Cryptology ePrint Archive*, 2004:107, 2004. 8, 12, 55, 109, 111, 118, 120

[BFJ+16] B. Bonnard, J-C. Faugere, A. Jacquemard, M. Safey El Din, and T. Verron. Determinantal sets, singularities and application to optimal control in medical imagery. In *International symposium on symbolic and algebraic computations*, Waterloo, Canada, July 2016. 40

[BFS14] M. Bardet, J-C. Faugère, and B. Salvy. On the complexity of the F5 Gröbner basis algorithm. *Journal of Symbolic Computation*, pages 1–24, September 2014. 30, 36

[BGJT14] R. Barbulescu, P. Gaudry, A. Joux, and E. Thomé. A heuristic quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. In *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, pages 1–16, 2014. 8

[BLF08] D. J. Bernstein, T. Lange, and R. R. Farashahi. Binary Edwards curves. In *Cryptographic Hardware and Embedded Systems - CHES 2008, 10th International Workshop, Washington, D.C., USA, August 10-13, 2008. Proceedings*, pages 244–265, 2008. 59

[BMMT94] E. Becker, T. Mora, M. G. Marinari, and C. Traverso. The shape of the shape lemma. *Proceedings of the International Symposium on Symbolic and Algebraic Computation (ISSAC' 94), New York, USA*, pages 129–133, 1994. 37, 38

[Bou13] C. Bouvier. The filtering step of discrete logarithm and integer factorization algorithms. Preprint, 22 pages, 2013. 70, 135

[BSSC05] I. Blake, G. Seroussi, N. Smart, and J. W. S. Cassels. *Advances in Elliptic Curve Cryptography (London Mathematical Society Lecture Note Series)*. Cambridge University Press, New York, NY, USA, 2005. 57

[Buc65] B. Buchberger. *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenrings nach einem nulldimensionalen Polynomideal.* PhD thesis, Universität Innsbruck, 1965. 8, 30, 32

[Buc06] B. Buchberger. Logic, mathematics and computer science: Interactions in honor of bruno buchberger (60th birthday) 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):475 – 511, 2006. 30

[Can97] D.G. Cantor. Computing in the Jacobian of an hyperelliptic curve. *Math. Comp.*, 48:565–582, 1997. 56

[CCS15] P. N. Chung, C. Costello, and B. Smith. Fast, uniform, and compact scalar multiplication for elliptic curves and genus 2 Jacobians with applications to signature schemes. *CoRR*, abs/1510.03174, 2015. preprint. 8, 9, 57

[CF05] H. Cohen and G. Frey, editors. *Handbook of elliptic and hyperelliptic curve cryptography.* CRC Press, 2005. 12, 46, 47, 54, 57, 58, 62, 65, 67, 68, 70

[Chi12] G. Childers. Factorization of a 1061-bit number by the special number field sieve. *IACR Cryptology ePrint Archive*, 2012:444, 2012. 136

[CJ03] Y. Choie and E. Jeong. Isomorphism classes of hyperelliptic curves of genus 2 over $\mathbb{F}_{2^n}$. *IACR Cryptology ePrint Archive*, 2003:213, 2003. 55, 109, 118

[CL11] C. Costello and K. E. Lauter. Group law computations on Jacobians of hyperelliptic curves. In *Selected Areas in Cryptography - 18th International Workshop, SAC 2011, Toronto, ON, Canada, August 11-12, 2011, Revised Selected Papers*, pages 92–117, 2011. 57

[CLO97] D.A. Cox, J.B. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra.* Undergraduate texts in mathematics. Springer, 1997. 16, 20, 26, 28, 32, 38, 40, 41, 42, 44

[Col97] A Colin. Solving a system of algebraic equations with symmetries. *Journal of Pure and Applied Algebra*, 117:195 – 215, 1997. 40

[CW90] D. Coppersmith and S. Winograd. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*, 9:251–280, 1990. 36

[CZ81] D. Cantor and H. Zassenhaus. A new algorithm for factoring polynomials over finite fields. *Mathematics of Computation*, 36:587–592, 1981. 74

[dB88] Bert den Boer. Diffie-Hellman is as strong as discrete log for certain primes. In *Advances in Cryptology - CRYPTO '88, 8th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1988, Proceedings*, pages 530–539, 1988. 64

[DH76] W. Diffie and M. E. Hellman. New directions in cryptography. *IEEE Trans. Information Theory*, 22(6):644–654, 1976. 5, 62, 63

[Die03] C. Diem. The GHS attack in odd characteristic. *J. Ramanujan Math. Soc. 18*, No.1:1–32, 2003. 8, 61, 78

[Die06] C. Diem. An index calculus algorithm for plane curves of small degree. In *Algorithmic Number Theory, 7th International Symposium, ANTS-VII, Berlin, Germany, July 23-28, 2006, Proceedings*, pages 543–557, 2006. 7, 9, 69, 75, 76, 86, 92

[Die11] C. Diem. On the discrete logarithm problem in elliptic curves. *Compositio Mathematica*, 147:75–104, 2011. 9, 11, 15, 62, 78, 80, 81, 93, 95, 100, 102

[DK13] C. Diem and S. Kochinke. Computing discrete logarithms with special linear systems. preprint, available at http://www.math.uni-leipzig.de/∼diem/preprints/dlp-linear-systems.pdf, 2013. 10, 86, 90, 91, 92

[DT08] Claus Diem and Emmanuel Thomé. Index calculus in class groups of non-hyperelliptic curves of genus three. *J. Cryptology*, 21(4):593–611, 2008. 71

[Duq10] S. Duquesne. Traces of the group law on the Kummer surface of a curve of genus 2 in characteristic 2. *Mathematics in Computer Science*, 3(2):173–183, 2010. 15

[Edw07] H. M. Edwards. A normal form for elliptic curves. *Bulletin of American Mathematical Society*, 44:393–422, 2007. 59

[EG02] A. Enge and P. Gaudry. A general framework for subexponential discrete logarithm algorithms. *Acta Arithmetica*, 102:83–103, 2002. 7, 9, 61, 72, 73

[EGT11] A. Enge, P. Gaudry, and E. Thomé. An $L(1/3)$ discrete logarithm algorithm for low degree curves. *J. Cryptology*, 24(1):24–41, 2011. 7, 8, 61, 69, 73

[Eng02] A. Enge. Computing discrete logarithms in high-genus hyperelliptic Jacobians in provably subexponential time. *Math. Comput.*, 71(238):729–742, 2002. 7, 9, 61, 72

[ES02] A. Enge and A. Stein. Smooth ideals in hyperelliptic function fields. *Math. Comput.*, 71(239):1219–1230, 2002. 61, 72, 73

[Fau99] J-C. Faugère. A new efficient algorithm for computing Gröbner bases (F4). *Journal of Pure and Applied Algebra*, 139(1–3):61–88, June 1999. 8, 10, 19, 30, 34

[Fau02] 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*, ISSAC '02, pages 75–83, New York, NY, USA, 2002. ACM. 8, 19, 30, 34, 135

[Fau10] J-C. Faugère. FGb: A Library for Computing Gröbner Bases, http://www-polsys.lip6.fr/~jcf/FGb/index.html. 6327:84–87, September 2010. 10, 14, 34

[FDS11] J-C. Faugère, M. Safey El Din, and P-J. Spaenlehauer. Gröbner bases of bihomogeneous ideals generated by polynomials of bidegree (1, 1): Algorithms and complexity. *J. Symb. Comput.*, 46(4):406–437, 2011. 40

[FGHR] J-C. Faugère, P. Gaudry, L. Huot, and G. Renault. Sub-cubic change of ordering for Gröbner basis: a probabilistic approach, booktitle = International Symposium on Symbolic and Algebraic Computation, ISSAC '14, Kobe, Japan, July 23-25, 2014, pages = 170–177, year = 2014, url = http://doi.acm.org/10.1145/2608628.2608669, doi = 10.1145/2608628.2608669. 38

[FGHR14] J-C. Faugère, P. Gaudry, L. Huot, and G. Renault. Using symmetries in the index calculus for elliptic curves discrete logarithm. *J. Cryptology*, 27(4):595–635, 2014. 9, 12, 40, 81, 94

[FGLM93] J-C. Faugère, P. M. Gianni, D. Lazard, and T. Mora. Efficient computation of zero-dimensional Gröbner bases by change of ordering. *J. Symb. Comput.*, 16(4):329–344, 1993. 19, 30, 37, 38

[FHJ+14] J-C. Faugère, L. Huot, A. Joux, G. Renault, and V. Vitse. Symmetrized summation polynomials: using small order torsion points to speed up elliptic curve index calculus. In *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, pages 40–57, 2014. 9, 12, 15, 40, 80, 81, 94, 102, 121

[Fly93] E. V. Flynn. The group law on the Jacobian of a curve of genus 2. *J. Reine Angew. Math.*, 439:45–69, 1993. 15

[FM11] J-C. Faugère and C. Mou. Fast Algorithm for Change of Ordering of Zero-dimensional Gröbner Bases with Sparse Multiplication Matrices. In *Proceedings of the 36th international symposium on Symbolic and algebraic computation*, ISSAC '11, pages 115–122, New York, NY, USA, 2011. ACM. 19, 30, 38, 39, 135

[FMR94] G. Frey, M. Müller, and H-G. Rück. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Math. Comp.*, 62(206):865–874, 1994. 8, 9, 58, 111, 119

[Fou13] E. Fouotsa. *Pairing computation and arithmetic of elliptic curves for cryptography.* PhD thesis, Université Rennes 1 ; Université européenne de Bretagne, December 2013. 60

[FS12] J-C. Faugère and J. Svartz. Solving polynomial systems globally invariant under an action of the symmetric group and application to the equilibria of N vortices in the plane. In *International Symposium on Symbolic and Algebraic Computation, ISSAC'12, Grenoble, France - July 22 - 25*, pages 170–178, 2012. 40

[FS13] J-C. Faugère and J. Svartz. Gröbner bases of ideals invariant under a commutative group: the non-modular case. In *International Symposium on Symbolic and Algebraic Computation, ISSAC'13, Boston, MA, USA, June 26-29, 2013*, pages 347–354, 2013. 40

[FSEDV16] J-C. Faugère, M. Safey El Din, and T. Verron. On the complexity of computing Gröbner bases for weighted homogeneous systems. *Journal of Symbolic Computation*, 2016. 8, 16, 28, 36, 40

[FSS14] J-C. Faugère, P-J. Spaenlehauer, and J. Svartz. Sparse Gröbner Bases: the Unmixed Case. In *ISSAC 2014*, Kobe, Japan, July 2014. 20 pages, Corollary 6.1 has been corrected. 40

[Ful08] W. Fulton. *Algebraic curves: an introduction to algebraic geometry.* Advanced book classics. Addison-Wesley Pub. Co., Advanced Book Program, 2008. 16, 22, 50, 52

[Gal14] F. Le Gall. Powers of tensors and fast matrix multiplication. In *International Symposium on Symbolic and Algebraic Computation, ISSAC '14, Kobe, Japan, July 23-25, 2014*, pages 296–303, 2014. 36

[Gau00] P. Gaudry. An algorithm for solving the discrete log problem on hyperelliptic curves. In *Advances in Cryptology - EUROCRYPT 2000, International Conference on the Theory and Application of Cryptographic Techniques, Bruges, Belgium, May 14-18, 2000, Proceeding*, pages 19–34, 2000. 7, 9, 69, 74, 86, 135

[Gau07] P. Gaudry. Fast genus 2 arithmetic based on theta functions. *J. Mathematical Cryptology*, 1(3):243–265, 2007. 9, 15, 49, 57, 61

[Gau09] P. Gaudry. Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem. *J. Symb. Comput.*, 44(12):1690–1702, 2009. 8, 9, 11, 15, 62, 78, 80, 82, 93

[GG14] S. D. Galbraith and S. W. Gebregiyorgis. Summation polynomial algorithms for elliptic curves in characteristic two. In *Progress in Cryptology - INDOCRYPT 2014 - 15th International Conference on Cryptology in India, New Delhi, India, December 14-17, 2014, Proceedings*, pages 409–427, 2014. 9, 81, 94

[GG16] S. Galbraith and P. Gaudry. Recent progress on the elliptic curve discrete logarithm problem. *Designs, Codes and Cryptography*, 78(1):51–72, 2016. 78

[GHS02] P. Gaudry, F. Hess, and N. P. Smart. Constructive and destructive facets of Weil descent on elliptic curves. *J. Cryptology*, 15(1):19–46, 2002. 8, 61, 77, 78

[GJ90] M. R. Garey and D. S. Johnson. *Computers and Intractability; A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1990. 8

[GL09] P. Gaudry and D. Lubicz. The arithmetic of characteristic 2 Kummer surfaces and of elliptic Kummer lines. *Finite Fields and Their Applications*, 15(2):246–260, 2009. 9, 15, 49, 57, 61

[GPS02] S. D. Galbraith, S. M. Paulus, and N. P. Smart. Arithmetic on superelliptic curves. *Math. Comput.*, 71(237):393–405, 2002. 53

[GTTD07] P. Gaudry, E. Thomé, N. Thériault, and C. Diem. A double large prime variation for small genus hyperelliptic index calculus. *Math. Comput.*, 76(257):475–492, 2007. 10, 71, 75, 88

[HC14] H. Hisil and C. Costello. Jacobian coordinates on genus 2 curves. In *Advances in Cryptology - ASIACRYPT 2014 - 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014. Proceedings, Part I*, pages 338–357, 2014. 57

[Huo13] L. Huot. *Polynomial systems solving and elliptic curve cryptography.* PhD thesis, Université Pierre et Marie Curie - Paris VI, December 2013. 8

[Jou09] A. Joux. *Algorithmic Cryptanalysis.* Chapman & Hall/CRC, 1st edition, 2009. 65, 68, 70

[JV12] A. Joux and V. Vitse. Cover and decomposition index calculus on elliptic curves made practical - application to a previously unreachable curve over $\mathbb{F}_{q^6}$. In *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*, pages 9–26, 2012. 9, 15, 78, 86, 87, 88

[JV13] A. Joux and V. Vitse. Elliptic curve discrete logarithm problem over small degree extension fields - application to the static Diffie-Hellman problem $E(\mathbb{F}_{q^5})$. *J. Cryptology*, 26(1):119–143, 2013. 80, 94, 95, 98, 113

[KAF+10] T. Kleinjung, K. Aoki, J. Franke, A. K. Lenstra, E. Thomé, J. W. Bos, P. Gaudry, A. Kruppa, P. L. Montgomery, D. A. Osvik, H. J. J. te Riele, A. Timofeev, and P. Zimmermann. Factorization of a 768-bit RSA modulus. In *Advances in Cryptology - CRYPTO 2010, 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings*, pages 333–350, 2010. 136

[Kem11] G. Kemper. *Hilbert Series and Dimension.* Springer Berlin Heidelberg, Berlin, Heidelberg, 2011. 26

[Kob87] N. Koblitz. Elliptic curve cryptosystems. *Math. Comp.*, 48(177):203–209, 1987. 6

[Laz83] D. Lazard. Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In *Computer Algebra, EUROCAL '83, European Computer Algebra Conference, London, England, March 28-30, 1983, Proceedings*, pages 146–156, 1983. 8, 30, 34

[LL93] A.K. Lenstra and H.W.J. Lenstra. *The Development of the Number Field Sieve.* Number n° 1554 in Lecture Notes in Mathematics. Springer, 1993. 69

[LL15] K. Laine and K. E. Lauter. Time-memory trade-offs for index calculus in genus 3. *J. Mathematical Cryptology*, 9(2):95–114, 2015. 9, 71, 76, 90

[LM10] M. Lochter and J. Merkle. ECC Brainpool standard curves and curves generation, rfc 5639. 2010. 49, 61

[LN97] R. Lidl and H. Niederreiter. *Finite Fields.* Number vol. 20,ptie. 1 in EBL-Schweitzer. Cambridge University Press, 1997. 113

[LO91] B. A. LaMacchia and A. M. Odlyzko. Computation of discrete logarithms in prime fields. *Des. Codes Cryptography*, 1(1):47–62, 1991. 91

[LR16] D. Lubicz and D. Robert. Arithmetic on abelian and Kummer varieties. *Finite Fields and Their Applications*, 39:130–158, 2016. 8, 9, 49, 61

[Mau94] Ueli M. Maurer. Towards the equivalence of breaking the Diffie-Hellman protocol and computing discrete algorithms. In *Advances in Cryptology - CRYPTO '94, 14th Annual International Cryptology Conference, Santa Barbara, California, USA, August 21-25, 1994, Proceedings*, pages 271–281, 1994. 64

[Mil86a] V. S. Miller. Use of elliptic curves in cryptography. *Advances in Cryptology - CRYPTO 1985, in Lecture Notes in Comput. Sci.*, 330:419–453, 1986. 6

[Mil86b] J. S. Milne. *Abelian varieties*. G. Cornell and J. H. Silverman, editors, Arithmetic geometry (Storrs, Conn., 1984), Springer-Verlag, 1986. 77

[MM07] G.L. Mullen and C. Mummert. *Finite Fields and Applications*. Student mathematical library. American Mathematical Society, 2007. 114

[Mon87] P. L. Montgomery. Speeding the Pollard and elliptic curve methods of factorization. *Mathematics of Computation*, 48(177):243–264, 1987. 60

[MOV93] A. Menezes, T. Okamoto, and S. A. Vanstone. Reducing elliptic curve logarithms to logarithms in a finite field. *IEEE Trans. Information Theory*, 39(5):1639–1646, 1993. 8

[Nag10] K. Nagao. Decomposition attack for the Jacobian of a hyperelliptic curve over an extension field. In *Algorithmic Number Theory, 9th International Symposium, ANTS-IX, Nancy, France, July 19-23, 2010. Proceedings*, pages 285–300, 2010. 8, 9, 11, 78

[Nec94] V. I. Nechaev. Complexity of a determinate algorithm for the discrete logarithm. *Mat. Zametki*, 55:91–101, 1994. 7, 61, 63

[NIS99] NIST. Recommended elliptic curves for federal government use, http://csrc.nist.gov/groups/st/toolkit/documents/dss/nistrecur.pdf. 1999. 49, 61

[PH78] S. C. Pohlig and M. E. Hellman. An improved algorithm for computing logarithms over GF(p) and its cryptographic significance (corresp.). *IEEE Trans. Information Theory*, 24(1):106–110, 1978. 63, 66

[Res10] Certicom Research. Sec 2: Recommended elliptic curve domain prameters, http://www.secg.org/sec2-v2.pdf. 2010. 49, 61

[RSA78] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Comm. ACM*, 21(2):120–126, 1978. 5, 62

[RSSB16] J. Renes, P. Schwabe, B. Smith, and L. Batina. μKummer: efficient hyperelliptic signatures and key exchange on microcontrollers. In *Cryptographic Hardware and Embedded Systems – CHES 2016*, volume 9813, page 20, Santa Barbara, United States, August 2016. IACR, Springer-Verlag. 8, 9, 15, 49

[Sem98] I. A. Semaev. Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curve in characteristic p. *Math. Comput.*, 67(221):353–356, 1998. 8, 9

[Sem04] I. A. Semaev. Summation polynomials and the discrete logarithm problem on elliptic curves. *IACR Cryptology ePrint Archive*, 2004:31, 2004. 11, 20, 78, 79

[Sho97] V. Shoup. Lower bounds for discrete logarithms and related problems. *Advances in Cryptology - EUROCRYPT '97, International Conference on the Theory and Application of Cryptographic Techniques, Konstanz, Germany, May 11-15, 1997, Proceeding*, pages 256–266, 1997. 7, 61, 63

[Sho05] V. Shoup. *NTL*: A library for doing number theory, http://www.shoup.net/ntl/. 2005. 135

[Sil13] J. H. Silverman. *The Arithmetic of Elliptic Curves*. Graduate Texts in Mathematics. Springer New York, 2013. 16, 22, 49, 50, 52, 59

[Sma99] N. P. Smart. The discrete logarithm problem on elliptic curves of trace one. *J. Cryptology*, 12(3):193–196, 1999. 8, 9

[Sma01] N. P. Smart. The Hessian form of an elliptic curve. pages 118–125, 2001. 60

[Smi08] B. Smith. Isogenies and the Discrete Logarithm Problem in Jacobians of Genus 3 Hyperelliptic Curves. In Nigel Smart, editor, *Eurocrypt 2008*, volume 4965, pages 163–180, Istanbul, Turkey, April 2008. International Association for Cryptologic Research. 77

[Spa12] P-J. Spaenlehauer. *Solving multi-homogeneous and determinantal systems: algorithms, complexity, applications.* PhD thesis, Université Pierre et Marie Curie (Univ. Paris 6), October 2012. 8, 16, 25, 29, 35, 40

[SS14] P. Sarkar and S. Singh. A new method for decomposition in the Jacobian of small genus hyperelliptic curves. Preprint (2014), available at eprint.iacr.org, 2014. 10, 86, 87, 88, 92

[Ste13] S. Steidel. Gröbner bases of symmetric ideals. *Journal of Symbolic Computation*, 54:72 – 86, 2013. 40

[Str69] V. Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 13:354–356, 1969. 36

[Stu08] B. Sturmfels. *Algorithms in Invariant Theory*. Texts & Monographs in Symbolic Computation. Springer, 2008. 40, 43

[Sva14] J. Svartz. *Solving zero-dimensional structured polynomial systems*. PhD thesis, Université Pierre et Marie Curie - Paris VI, October 2014. 8, 16

[Tea15] The CADO-NFS Development Team. CADO-NFS, an implementation of the number field sieve algorithm, 2015. Release 2.2.0. 70, 135

[Thé03] N. Thériault. Index calculus attack for hyperelliptic curves of small genus. In *Advances in Cryptology - ASIACRYPT 2003, 9th International Conference on the Theory and Application of Cryptology and Information Security, Taipei, Taiwan, November 30 - December 4, 2003, Proceedings*, pages 75–92, 2003. 71

[UDP15] F. Urvoy De Portzamparc. *Algebraic and Physical Security in Code-Based Cryptography*. PhD thesis, Université Pierre et Marie Curie - Paris VI, April 2015. 8

[Ver02] F. Vercauteren. Computing zeta functions of hyperelliptic curves over finite fields of characteristic 2. In *Advances in Cryptology - CRYPTO 2002, 22nd Annual International Cryptology Conference, Santa Barbara, California, USA, August 18-22, 2002, Proceedings*, pages 369–384, 2002. 135

[Ver16] T. Verron. *Régularisation du calul de bases de Gröbner pour des systèmes avec poids et déterminantiels, et application en imagerie médicale.* PhD thesis, Université Pierre et Marie Curie, Septembre 2016. 29, 38

[Vit11] V. Vitse. *Attaques algébriques du problème du logarithme discret sur courbes elliptiques*. PhD thesis, 2011. 46, 80

[VW15] V. Vitse and A. Wallet. Improved sieving on algebraic curves. In *Progress in Cryptology - LATINCRYPT 2015 - 4th International Conference on Cryptology and Information Security in Latin America, Guadalajara, Mexico, August 23-26, 2015, Proceedings*, pages 295–307, 2015. 10, 17, 86

[vzGG13] J. von zur Gathen and J. Gerhard. *Modern Computer Algebra*. Cambridge University Press, third edition, 2013. Cambridge Books Online. 39, 74, 81

[Wie86] D. H. Wiedemann. Solving sparse linear equations over finite fields. *IEEE Trans. Information Theory*, 32(1):54–62, 1986. 38, 70, 74