HAL open science

The Point Decomposition Problem over Hyperelliptic Curves: toward efficient computations of Discrete Logarithms in even characteristic

Jean-Charles Faugère, Alexandre Wallet

▶ To cite this version:

Jean-Charles Faugère, Alexandre Wallet. The Point Decomposition Problem over Hyperelliptic Curves: toward efficient computations of Discrete Logarithms in even characteristic. Designs, Codes and Cryptography, 2018, 86, pp.2279-2314. 10.1007/s10623-017-0449-y . hal-01658573

HAL Id: hal-01658573

https://inria.hal.science/hal-01658573v1

Submitted on 15 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Design, Codes and Cryptography manuscript No.
(will be inserted by the editor)

The Point Decomposition Problem over hyperelliptic curves

Toward efficients computation of discrete logarithms in even characteristic

Jean-Charles Faugère · **Alexandre Wallet**

the date of receipt and acceptance should be inserted later

Keywords discrete logarithm, index calculus, algebraic curves, curve-based cryptography, algebraic attacks, Gröbner bases

Mathematics Subject Classification (2000) 14G50

Abstract Computing discrete logarithms is generically a difficult problem. For divisor class groups of curves defined over extension fields, a variant of the Index-Calculus called Decomposition attack is used, and it can be faster than generic approaches. In this situation, collecting the relations is done by solving multiple instances of the Point m-Decomposition Problem (PDP_m). An instance of this problem can be modelled as a zero-dimensional polynomial system. Solving is done with Gröbner bases algorithms, where the number of solutions of the system is a good indicator for the time complexity of the solving process. For systems arising from a PDP_m context, this number grows exponentially fast with the extension degree. To achieve an efficient harvesting, this number must be reduced as much as as possible. Extending the elliptic case, we introduce a notion of Summation Ideals to describe PDP_m instances over higher genus curves, and compare to Nagao's general approach to PDP_m solving. In even characteristic we obtain reductions of the number of solutions for both approaches, depending on the curve's equation. In the best cases, for a hyperelliptic curve of genus g, we can divide the number of solutions by $2^{(n-1)(g+1)}$. For instance, for a type II genus 2 curve defined over $\mathbb{F}_{2^{93}}$ whose divisor class group has cardinality a near-prime 184 bits integer, the number of solutions is reduced from 4096 to 64. This is enough to build the matrix of relations in around 7 days with 8000 cores using a dedicated implementation.

1 Introduction

The Point m-Decomposition Problem (PDP_m)

The Discrete Logarithm Problem (DLP) is a well-known and generically difficult problem, and several standard cryptographic protocols rely on its hardness (for example, Diffie-Hellman key exchange or digital signature algorithms). We focus on its instance over the divisor class group $\mathrm{Jac}(\mathscr{H})$ of an hyperelliptic[1] curve $\mathscr{H}$ of genus g defined over "small" field extensions. By small, we mean that the extension degree admits a small

Sorbonnes Universités, UPMC Univ Paris 06, CNRS, INRIA, Laboratoire d'Informatique de Paris 6 (LIP6), Equipe PolSys, 4 place Jussieu 75005 Paris, France
E-mail: jean-charles.faugere@inria.fr · wallet.alexandre@gmail.com

[1] Throughout the article we usually consider elliptic curves as hyperelliptic curves of genus 1. It makes no differences in our contributions.

factor — typically, it is $\mathbb{F}_{q^{nk}}$ with $2 \leq k \leq 6$. Computing discrete logarithms in such groups can be done using a variant of the Index-Calculus algorithm called *Decomposition attacks*. Such algorithms run in mainly two phases. Our main interest is the first one, called *relations collection* or *harvesting*. In the harvesting phase, a linear system is built by finding linear relations between (the discrete logarithms of) elements of a special subset, the so-called *factor base*. In Decomposition attacks, relations can be found by solving multiple instances of the following problem.

Definition 1 (Point m-Decomposition Problem (PDP$_m$)) *Given an element R and a subset $\mathscr{B}$ of* $\mathrm{Jac}(\mathscr{H})$*, find, if they exist, $D_1, \ldots, D_m$ in $\mathscr{B}$ such that:*

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

The m-tuple $(D_1, \ldots, D_m)$ is called a m-decomposition of R, or a decomposition if the context is clear.

Nagao proposed in [38] to solve instances of this problem by describing decompositions using functions in adequate Riemann-Roch spaces. If $\mathscr{H}$ has genus g and is defined over some $\mathbb{F}_{q^n}$, he selected the factor base as $\mathscr{B} = \{P \in \mathscr{H},\ x(P) \in \mathbb{F}_q\}$ and used the linear structure of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$ to describe a decomposition by a multivariate polynomial system. This process is usually called *Weil descent* in the litterature, and we will do the same. When the curve is elliptic, *i.e.* $g = 1$, an alternate approach involving Weil descent on Summation polynomials [40] can also be used, as shown by Diem [10] and Gaudry [23]. The systems arising from these methods are generally zero-dimensional and solved by Gröbner bases methods.

For zero-dimensional ideals, the standard solving strategy using Gröbner bases is to first compute a basis for a total degree order, then to change for a lexicographical basis using a change-order algorithm. Total degree order bases are computed with the algorithms F4 [14] or F5 [15], and for zero-dimensional ideals, the change-order step is done with FGLM's algorithm [13] or its Sparse variant [18]. On the one hand, F4 or F5's complexities are expressed using the degree of regularity of the ideal [1]. In the context of a Decomposition attack, it can be approximated by the number of solutions of the system. On the other hand, the classic FGLM is usually the computational bottleneck in PDP$_m$ solving, even when the Sparse variant is used (although very important speed-ups are observed). Its complexity depends polynomially on the dimension of the quotient algebra as a linear space. This dimension equals the number of solutions of the system over an algebraic closure, and coincides with the degree of the ideal. Hence we will use both terminologies throughout the presentation, and we use this quantity to estimate the complexity of solving a PDP$_m$ instance. Efficient implementations of Gröbner bases algorithms exist in Magma [2] and in Maple with the FGb package [12]. These were also our main tool for experimentations.

In genus 1, Summation and Nagao's approaches give systems with $2^{n(n-1)}$ solutions, the Summation approach being experimentally faster. When $g > 1$, Nagao's approach leads to systems with $d_{\mathrm{Nag}} = 2^{n(n-1)g}$ solutions. In both cases, the number of solutions grows too quickly with the extension degree (and the genus) to consider practical computations, and even experiments. For example, for $k = \mathbb{F}_{65521}$, $g = 2$ and $[K : k] = n = 3$, a Magma 2.19 [2] implementation of Nagao's method needs roughly 1300 sec to solve a degree 4096 system and thus a PDP$_6$ instance. The probability to find a 6-decomposition is $1/6!$, hence $6! \times 1300\text{sec.} = 936000$ sec. are needed in average to find a single relation. Therefore to achieve an efficient relation harvesting in a Decomposition attack, the degree of the ideals must be reduced.

Contributions

Throughout this article efficiency means time efficiency. Our general goal is to design an efficient approach for solving PDP$_m$ instances, in order to implement a Decomposition attack over a meaningful genus 2 curve. In other words, the cardinality of its divisor class group is a near-prime integer whose size is close to lowest acceptable security level. To do this, we propose new ways to reduce the degree of the systems arising from the harvesting in even characteristic. Our contributions can be separated in two categories: those which deal with Nagao's approach, and those which focus on the Summation approach. We develop both in parallel and show

how to reduce the degree for both methods in even characteristic. Another benefit is that, informally, the reduction process tends to give a more "homogeneous" shape to the system's equations. The more homogeneous a system is, the better its behaviour during a Gröbner basis computation tends to be [46]. In the last chapter we finally compare them, adding our new degree reduction algorithms, to determine which is more efficient when $g > 1$. Nagao's approach reveals to be more efficient in practice, and thus it is used in the last Section to estimate the total running time for the harvesting on a binary genus 2 curve with a class group of around 2^{184} elements. We now detail the organization of the article.

Reducing the degree of the ideals for Nagao's modelling in even characteristic Our first focus is Nagao's approach for solving PDP_{ng} instances when $g > 1$. We introduce the *Decomposition polynomial*, which describes the generic intersection between a function with prescribed valuation at infinity and the target hyperelliptic curve. Its coefficients depend symbolically on the coordinates of the function in an adequate basis of a Riemann-Roch space, and are used to generate the polynomials systems describing PDP_{ng} instances.

In characteristic 2 we observe that one of these coefficients is always univariate. This enables a "presolving" by determining up to $n-1$ solutions of the system. Additionally we observe that some other coefficients of the Decomposition Polynomial are squares. Any square equation can be replaced by its square root, and each replacement reduces the degree of the final system. We give an explicit formula for the number of square equations, that depends on the *length* of h_1, defined as the difference between the degree of the leading and the trailing term. From this we deduce bounds on the reduced degree of the system after the Weil Descent. More precisely, if we denote by d_{opt} the reduced degree, we obtain

$$2^{(n-1)((n-1)g-1)} \leq d_{\mathrm{opt}} \leq 2^{(n-1)(ng-1)},$$

compared to the previous $d_{\mathrm{Nag}} = 2^{n(n-1)g}$. The Section ends with an exhaustive analysis of the degree reduction for binary genus 2 curves, where a complete classification is known.

Defining Summation ideals for hyperelliptic curves Our second focus is the use of Summation Polynomials for modelling PDP_m instances when $g \geq 2$. The idea behind this study is that using Summation polynomials for Decomposition attacks over elliptic curves revealed to be more efficient than Nagao's approach. The major reason was the possibility to exploit symmetries to reduce the degree of the system to solve [16][17][22], and another yet less impactful reason was the smaller number of variables. While the existence of such objects is intuitive, to the best of our knowledge, the only computational approach proposed in the litterature was [44]. However, several problems inherent to the proposed modelling prevented a reasonable usage for PDP_m solving. The main concerns were the cost of solving a system, and the harder and seemingly not possible generalization to any hyperelliptc curves of genus $g > 1$. Our new approach is more close to the geometric framework proposed in [10]. It is also computational in nature, and solves the previous concerns at least theoretically. In particular our notion of Summation polynomials can be defined for any algebraic curves, although we focus on hyperelliptic ones.

For any hyperelliptic curve $\mathscr{H}$ given in imaginary model, we follow the geometric presentation of [10] and the ideas of [30] to introduce *Summation varieties* as

$$\mathscr{V}_{m,R} = \{(P_1, \ldots, P_m) : \sum_{i=1}^{m} (P_i - P_\infty) = R, P_i \in \mathscr{H}\},$$

where P_∞ is the point at infinity and $R \in \mathrm{Jac}(\mathscr{H})$ is a fixed reduced divisor. Let π be the projection on the x-line (informally). We define the mth *Summation ideals* of $\mathscr{H}$ as the ideal associated to $\pi(\mathscr{V}_{m,R})$, and m^{th} *Summation polynomials* as any generating sets for this ideal. We give a polynomial parametrization of $\pi(\mathscr{V}_{m,R})$, thus an algorithmic way to describe and compute Summation polynomials. This description enables us to show that $\operatorname{codim} \pi(\mathscr{V}_{m,R}) = g$, so that such sets must have at least g elements. The standard elliptic Summation Polynomial from Semaev is recovered[2] as the case $g = 1$, hence this new notion extends Semaev's [40]. To the best of our

[2] It was already observed in [30]

knowledge, it was not mentioned anywhere in the litterature before, neither in the recent survey [21]. Also, our experiments and the geometric framework leads us to formulate a conjecture that $\deg \pi(\mathscr{V}_{m,R}) = 2^{m-g}$. This conjecture allows us to estimate the number of solution if a PDP_m instance is generated using Summation polynomials. We note that it was already proven by Diem [10] for elliptic curves. We give a new algorithm to solve PDP_m instances for hyperelliptic curves and discuss its efficiency compared to Nagao's, assuming our degree conjecture is true.

Reducing the degree for the new Summation modelling in even characteristic: exploiting Frobenius The properties we observe for the coefficients of the Decomposition Polynomial translate differently when a Summation approach is selected. In even characteristic, the presence of square coefficients expresses as the action of the Frobenius automorphism. These systems are particular case of polynomial parametrizations in perfect fields of characteristic $p \geq 2$. We elaborate on the general situation rather than directly on the PDP_m setting, as we believe our results here to be of more general interest. A polynomial parametrization is an ideal generated by polynomials as $X_i - P_i(a_1, \ldots, a_l)$. If some P_i's can be written as a pth power of other polynomials, then we can consider the ideal obtained by removing these powers. In characteristic p, the solutions we found after elimination of $a_1, \ldots, a_l$ should be essentially the same for both ideals, up to action of the Frobenius automorphism. We show that this is indeed the case. While it makes no difference geometrically to work with an ideal or the other, removing the pth powers reduces the degrees of the defining equations, which is a crucial parameter in any Gröbner basis computation. This reduction expresses as a faster running time in the computation of a basis. It also leads to a reduction of the degree of the systems in a PDP_m context in even characteristic. If $\mathbb{I}$ is the system to be solved in this setting, we show that its number of solution is

$$\deg \mathbb{I} = C_1^n \cdot \frac{d_{\mathrm{Nag}}}{2^{(n-1)g+L_1}},$$

where the *degree ratio* C_1 (Definition 28) is a constant that depends only the polynomial h_1 in the curve's equation, and L_1 is the length of h_1.

Comparisons of methods and discrete logarithm computations for genus 2 *curves over* $\mathbb{F}_{2^{93}}$ The next step is to compare both methods using Magma [2] implementations, see Tables 5 and 6. In odd characteristic our experiments show that Nagao's approach is more efficient. In even characteristic, we focus on genus 2 binary curves. The reason is that there were at a time suggested as additional standards [4], and they have been shown some arithmetical interest recently [7][25][26][36][39]. For Type *II* curves, that is to say, curves such that $\mathrm{Jac}(\mathscr{H})$ has 2-rank one, the Summation approach and Nagao's lead to ideals of the same degrees after reduction, and the reduced degree is the smallest we could obtain with this work. Our comparative experiments reveal that Nagao's is faster overall.

The last Section describes our implementation of a Decomposition attack on a Type *II* binary curve $\mathscr{H}$ defined over $\mathbb{F}_{2^{3\cdot 31}}$, whose Class Group $\mathrm{Jac}(\mathscr{H})$ has a nearly prime cardinality of

$$\#J = 2 \times 3 \times 16346619102569543707881667303220993643142373107431938653,$$

where the biggest factor is 184 bits long. Hence a generic algorithm would need around 2^{92} operations to compute a discrete logarithm in this group. From our previous comparisons, Nagao's approach is selected and our degree reduction algorithm is added. Then we implemented an optimized and dedicated version of the relation harvesting using efficient Gröbner bases algorithms. More details on the implementation can be found in Section 5.3.

For this curve, $n = 3$ and $g = 2$, and systems of degree $2^{(n-1)((n-1)g-1)} = 2^{2\cdot(4-1)} = 64$ must be solved. Our dedicated implementation reduces the time to solve one PDP_6 instance to about $3.2 \cdot 10^{-3}$sec. on $\mathscr{H}$. Overall, approximately $6! \times 3.2 \cdot 10^{-3}$sec.$=$ 2.3sec. are needed in average to find a solution. Using 8000 cores, it then takes a bit more than 7 days to build an overdetermined matrix. We also estimate the number of operations linear algebra to around 2^{63} (after efficient filtering steps [3][5]). Security-wise, this suggest that Type *II* curves are weaker than expected against Decomposition attacks.

Magma code: The Magma code we used to obtain the timings in the experiments is available at `hypersum.gforge.inria.fr`.

2 Nagao's approach for PDP_m solving

We first remind some theoretical background about divisors and the Jacobian variety of hyperelliptic curves, seen as its degree 0 divisor class group. We also recall Nagao's approach to PDP_m solving, highlighting the *Decomposition polynomial* (Definition 6). Its coefficients are used to generate the multivariate systems related to an instance, and they are the focus of the rest of the Section. In the rest of the article, we always assume that we are working over a perfect field.

2.1 Riemann-Roch's coordinates to model PDP_{ng} instances

Let $\mathscr{H}$ be an imaginary hyperelliptic curve of genus g, defined over a field $\mathbb{F}$ by a Weierstrass equation $y^2 + h_1(x)y = h_0(x)$, $\deg h_1 \leqslant g, \deg h_0 = 2g+1$, and let P_∞ be its single point at infinity. The degree 0 divisor class group of the curve is denoted by $\mathrm{Jac}(\mathscr{H})$. In any class, there exists a unique divisor $R = P_1 + \cdots + P_k - kP_\infty$, with $k \leq g$, such that no two P_i, P_j are the images of each other by the hyperelliptic involution. Such divisor is called *reduced*, so that any class in $\mathrm{Jac}(\mathscr{H})$ can be thought as a reduced divisor. A computational way to represent reduced divisors is the Mumford Representation.

Definition 2 (Mumford representation) *Let $R = P_1 + \cdots + P_k - kP_\infty \in \mathrm{Jac}(\mathscr{H})$ be a reduced divisor with $P = (x_i, y_i) \in \mathscr{H}$. Let $u(X) = \prod_{1 \leq i \leq k}(X - x_i)$. There exists a unique $v(X) \in \mathbb{F}[X]$ such that:*

- $\deg v < \deg u = k$, *and* $v(x_i) = y_i$ *for* $1 \leq i \leq k$.
- $u \mid (v^2 + vh_1 - h_0)$.

The pair $(u(X), v(X))$ is called the Mumford representation of R, and we write $R = (u, v)$ to denote that (u, v) is the Mumford representation of R. We call the integer k the weight of D.

When we write $R \in \mathrm{Jac}(\mathscr{H})$ we mean that we consider a reduced divisor, unless we state otherwise.

Remark 3 *A random divisor in $\mathrm{Jac}(\mathscr{H})$ has weight g with very high probability.*

For any divisor D, we denote by $\mathscr{L}(D)$ the Riemann-Roch space associated to D. It is a $\mathbb{F}$-linear space of finite dimension. When a reduced divisor R of weight g is fixed, we are particularly interested in Riemann-Roch spaces as $\mathscr{L}(mP_\infty - R)$, $m \in \mathbb{N}$.

Remark 4 *If $m < g+1$ then no basis of $\mathscr{L}(mP_\infty - R)$ can contain a function involving y, since it has a pole of order $2g+1$ at P_∞. But if f is a function of x and vanishes at P, then it also vanishes at $-P$ and thus $P + (-P) - 2P_\infty$ is in the support of $\mathrm{div}\, f$. Any such divisor reduces to $\mathscr{O}$ in the Jacobian, and therefore we need at least $m \geq g+1$. We will always assume that this is the case in this Section.*

If (u, v) is the Mumford representation of R with $\deg u = g$, then a natural basis of $\mathscr{L}(mP_\infty - R)$ is given by

$$\{u, xu, \ldots, x^{d_1}u, y - v, x(y - v), \ldots, x^{d_2}(y - v)\}, \tag{1}$$

with $d_1 = \lfloor (m-g)/2 \rfloor$ and $d_2 = \lfloor (m-g-1)/2 \rfloor$. If we let $d = m - g = d_1 + d_2 + 1$, then any $f \in \mathscr{L}(mP_\infty - R)$ can be written as

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

where $a_1,\dots,a_{d+1}\in\mathbb{F}$. It can be verified that $f(x,y)f(x,-y-h_1(x))$ is a degree $m+g$ polynomial in x, and is generally given by:

$$\begin{aligned} f(x,y)f(x,-y-h_1(x)) &= (vq-up)^2+q(vq-up)h_1-q^2h_0 \\ &= (up)^2-upq(2v+h_1)+q^2(v^2+vh_1-h_0). \end{aligned}$$

Considering the degrees of all polynomials involved, we see that its leading coefficient is $\mathrm{LC}((up)^2)=a_{2d_1+1}=a_{d+1}$ if $m-g$ is even and $\mathrm{LC}(-q^2h_0)=-a_{2d_2+2}=-a_{d+1}$ if $m-g$ is odd. In the rest of this article, functions are always normalized at infinity, that is to say we set $a_{d+1}=1$. This implies that $(-1)^{m-g}f(x,y)f(x,-y-h_1(x))$ is monic.

In term of of the quadratic field extension $\mathbb{F}(\mathscr{H})\mid\mathbb{F}[x]$, the polynomial $f(x,y)f(x,-y-h_1(x))$ is known as the norm of the function f, and can also be computed as a resultant with respect to y since the function is polynomial. We keep a close terminology in the next Definition.

Definition 5 *Let $R=(u,v)\in\mathrm{Jac}(\mathscr{H})$ of weight g, and let $d=\dim\mathscr{L}(mP_\infty-R)-1$.*

- *A polynomial $f(X,Y)\in(\mathbb{F}[a_1,\dots,a_d])[X,Y]$ as in Equation (2) is called a generic function in $\mathscr{L}(mP_\infty-R)$.*
- *The generic norm of a generic function is $N(f)=(-1)^{m-g}Res_Y(f(X,Y),Y^2+h_1(X)Y-h_0(X))$.*

The generic norm is a monic polynomial in $(\mathbb{F}[a_1,\dots,a_d])[X]$. We now fix a reduced divisor $R=(u,v)\in\mathrm{Jac}(\mathscr{H})$ of weight g. Recall that an equivalence of divisors $R=P_1+\cdots+P_m-mP_\infty$ is called a *m-decomposition of R* in this article, or decomposition if the context is clear. Considering a decomposition means there is $f\in\mathscr{L}(mP_\infty-R)$ such that $\operatorname{div}f+R=\sum_{i=1}^m(P_i-P_\infty)$. Hence for a generic function $f\in\mathscr{L}(mP_\infty-R)$,

$$\frac{N(f)}{u(X)}=F(X)=X^m+\sum_{i=0}^{m-1}N_{m-i}(a_1,\dots,a_d)X^i, \tag{3}$$

is also a polynomial in $(\mathbb{F}[a_1,\dots,a_d])[X]$, with $\deg N_i=2$ for all i. As F vanishes exactly at the abscissae of the P_i's, it describes a decomposition of R. This polynomial is core to our contributions so we give it its own definition.

Definition 6 *Let $R=(u,v)\in\mathrm{Jac}(\mathscr{H})$ of weight g, and let f be a generic function in $\mathscr{L}(mP_\infty-R)$. The polynomial $F(X)=\frac{N(f)}{u(X)}\in(\mathbb{F}[a_1,\dots,a_d])[X]$ is called the Decomposition polynomial.*

If the context is not clear, we may say the R-Decomposition polynomial to highlight that it describes a decomposition of R.

Nagao's approach to find decompositions In a Decomposition attack, the field is $\mathbb{F}_{q^n}$ and the genus g is fixed. The standard factor basis is $\mathscr{B}=\{P-P_\infty : x(P)\in\mathbb{F}_q\}$ and we solve PDP_{ng} instances: given $R\in\mathrm{Jac}(\mathscr{H})$, we try to find a decomposition as $R=P_1+\cdots+P_{ng}-ngP_\infty$. This is equivalent to the existence of $f\in\mathscr{L}(ngP_\infty-R)$ such that

$$\operatorname{div}f+R=P_1+\cdots+P_{ng}-ngP_\infty, \tag{4}$$

where all P_i are in $\mathscr{B}$, and with $m=ng$ and $d=(n-1)g$. The goal is to determine such a function f, that is to say, its coefficients $a_1,\dots,a_d$. as in Equation (2). To do this we compute the Decomposition polynomial as in Equation (3). Let x_i be the abscissa of P_i. To have all $x_i\in\mathbb{F}_q$, it is necessary that $F\in\mathbb{F}_q[x]$, or equivalently, that we find $a_1^*,\dots,a_d^*\in\mathbb{F}_{q^n}$ such that $N_i(a_1^*,\dots,a_d^*)\in\mathbb{F}_q$ for all $1\le i\le ng$.

This can be achieved with a so-called *Weil descent*. Let $1,t,\dots,t^{n-1}$ be a power basis of $\mathbb{F}_{q^n}$, and write $a_i=\sum_{j=0}^{n-1}a_{i,j}t^j$ with $a_{i,j}\in\mathbb{F}_q$. Using the notation $\mathbf{a}=(a_{1,0},\dots,a_{1,n-1},\dots,a_{d,0},\dots,a_{d,n-1})$, we have

$$N_i(a_1,\dots,a_d)=\sum_{j=0}^{n-1}N_{i,j}(\mathbf{a})t^j, \tag{5}$$

with $N_{i,j} \in \mathbb{F}_q[\mathbf{a}]$, so that all N_i belong to $\mathbb{F}_q$ exactly at the solutions of the system

$$\mathscr{N} = \{N_{i,j}(\mathbf{a}) = 0,\ 1 \leqslant i \leqslant ng,\ 1 \leqslant j \leqslant n-1\}. \tag{6}$$

Assume $\mathbf{s}$ is such a solution. Then in addition, we have to check that the specialized polynomial $F^*(X) = X^{ng} + \sum_{i=0}^{ng-1} N_{ng-i}(\mathbf{s})X^i$ 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. If $\mathfrak{S}_n$ denotes the symmetric group of n elements, the probability of finding such decompositions is heuristically given by

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

Solving systems in Nagao's approach Having $n(n-1)g$ quadratic equations in $n(n-1)g$ variables, systems like $\mathscr{N}$ are generally zero-dimensional, and we solve them using Gröbner bases methods. In the introduction, we mentioned that the complexity of the strategy using Gröbner bases can be estimated by the number of solutions.

With Nagao's approach to Decomposition attacks, systems as $\mathscr{N}$ usually have $d_{\mathrm{Nag}} = 2^{n(n-1)g}$ solutions. This grows exponentially fast with the genus and the extension degree, and moreover, the probability of finding a solution drops exponentially fast as well. Even experimentally, computations generally take too long when $n(n-1)g > 12$ to find relations. This is a first reason why we strive to reduce the degree d_{Nag} as much as possible.

2.2 Properties of the Decomposition polynomials' coefficients

We give a general expression for the Decomposition polynomial. In even characteristic, it is used to show that the coefficient of highest degree is an univariate polynomial, and that some other are squares. If L_1 is the *length* of h_1 (Definition 10), we count the number of square coefficients as $g+1-L_1$. Additional squares can be found when the leading coefficient of h_1 belong to a subfield — in practice, h_1 is monic. The number of square is directly related to the degree reduction, as any squared equation can be replaced by a linear equation in even characteristic.

Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ a hyperelliptic curve of genus g defined over a field $\mathbb{F}$ of arbitrary characteristic for now, but soon we will impose that it is even. Throughout this Section, we fix $R \in \mathrm{Jac}(\mathscr{H})$ of weight g, unless we state otherwise.

An expression for the Decomposition polynomial Using the natural basis (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 $p(X) = \sum_{i=0}^{d_1} a_{2i+1}X^i$ and $q(X) = \sum_{i=0}^{d_2} a_{2i+2}X^i$ to write a generic function f, normalized at infinity, as:

$$f(X,Y) = u(X)p(X) + (Y - v(X))q(X).$$

Its norm is then a monic polynomial in $(\mathbb{F}[a_1,\ldots,a_d])[X]$ of degree $m+g$ in X, given by:

$$\begin{aligned}(-1)^{m-g}N(f) &= (vq-up)^2 + q(vq-up)h_1 - q^2h_0\\ &= (up)^2 - 2upvq - upqh_1 + q^2(v^2 + vh_1 - h_0)\\ &= u(up^2 - pq(2v+h_1) + q^2w),\end{aligned}$$

where $u, h_1, w \in \mathbb{F}[X]$ and w is a polynomial such that $uw = v^2 + h_1v - h_0$, coming from the properties of the Mumford representation. We note that $\mathrm{LC}_X(v^2 + h_1v - h_0) = \mathrm{LC}_X(-h_0) = -1$, that is to say $-w$ is monic. Hence the Decomposition polynomial has the following general expression in $(\mathbb{F}[a_1,\ldots,a_d])[X]$:

$$(-1)^{m-g}F(X) = up^2 - pq(2v+h_1) + q^2w = X^m + \sum_{i=0}^{m-1} N_{m-i}(a_1,\ldots,a_d)X^i. \tag{7}$$

The coefficient N_1 is univariate We now assume that $\mathrm{Char}(\mathbb{F}) = 2$ for the rest of the Section, unless stated otherwise. The next Proposition generalizes an observation in [30] to every genus.

Proposition 7 *Let $h_1(X) = \sum_{i=0}^{g} H_i X^i$ with $H_i = 0$ for $i > \deg h_1$, and write $h_0(X) = X^{2g+1} + h_{2g}X^{2g} + \cdots$. Let $R = (u,v) \in \mathscr{H}$ of weight g, and write $u(X) = X^g + u_1 X^{g-1} + \cdots \in \mathbb{F}[X]$. Let F be the R-Decomposition polynomial. Then its coefficient $N_1(a_1,\ldots,a_d)$ is always an univariate polynomial. More precisely, we have:*

$$N_1(a_1,\ldots,a_d) = N_1(a_d) = \begin{cases} a_d^2 + H_g a_d + u_1, \text{ if } d \text{ is even} \\ a_d^2 + H_g a_d + u_1 + h_{2g}, \text{ if } d \text{ is odd} \end{cases}.$$

Proof First we notice that $\deg_X pqh_1 \leq m-1$ and if $H_g \neq 0$, then $\mathrm{LC}_X(pqh_1) = H_g a_d$. Before normalization, we can write

$$\begin{aligned} up^2 =& (X^g + u_1 X^{g-1} + \cdots)(a_{2d_1+1}^2 X^{2d_1} + a_{2d_1}^2 X^{2d_1-2} + \cdots) \\ =& a_{2d_1+1}^2 X^{2d_1+g} + u_1 a_{2d_1+1}^2 X^{2d_1+g-1} + \cdots \end{aligned}$$

and

$$\begin{aligned} q^2 w =& (a_{2d_2+2}^2 X^{2d_2} + a_{2d_2}^2 X^{2d_2-2} + \cdots)(X^{g+1} + w_g X^g + \cdots) \\ =& a_{2d_2+2}^2 X^{2d_2+g+1} + w_g a_{2d_2+2}^2 X^{2d_2+g} + \cdots \end{aligned}$$

If d is even, then p is monic as a polynomial in X, and so is up^2 in expression (7), and we have $\deg_X up^2 = m$, $\deg_x q^2 w = m-1$. The leading coefficient in X of $up^2 - X^m$ is u_1. We conclude as $\mathrm{LC}_X(q^2 w) = a_d^2$.

If d is odd, then q is monic as a polynomial in X, and we have $\deg_X up^2 = m-1$, $\deg_X q^2 w = m$. In this case $\mathrm{LC}_X(q^2 w - X^m) = w_g$. Since we have $uw = v^2 + 2h_1 v + h_0$, and that v^2 and vh_1 have degree less than $2g$, then $w_g = u_1 + h_{2g}$. We conclude as $\mathrm{LC}_X(up^2) = a_d^2$.

□

Now let $\mathbb{F} = \mathbb{F}_{2^{kn}}$ with $\mathbb{F}_{2^k}$-power basis $1, t, \ldots, t^{n-1}$ and write $a_d = \sum_{i=0}^{n-1} a_{d,i} t^i$. For simplicity, we assume temporarily that d is even. If $H_g \in \mathbb{F}_{2^k}$, which is generally the case as h_1 is monic in practice, then Proposition 7 gives:

$$\begin{aligned} N_1(a_d) &= a_d^2 + H_g a_d + u_1 \\ &= \left(\sum_{i=0}^{n-1} a_{d,i} t^i\right)^2 + H_g \sum_{i=0}^{n-1} a_{d,i} t^i + \sum_{i=0}^{n-1} u_{1,i} t^i \\ &= \sum_{i=0}^{n-1} a_{d,i}^2 t^{2i} + H_g a_{d,0} + H_g \sum_{i=1}^{n-1} a_{d,i} t^i + \sum_{i=0}^{n-1} u_{1,i} t^i \\ &= N_{1,0}(a_{d,0},\ldots,a_{d,n-1}) + \sum_{i=1}^{n-1} N_{1,i}(a_{d,1},\ldots,a_{d,n-1}) t^i. \end{aligned}$$

This shows that the last $n-1$ coefficients in t of $N_1(a_d)$ form a system with $n-1$ equations of degree 2

$$\mathscr{S}_1 = \{N_{1,i}(a_{d,1},\ldots,a_{d,n-1}) = 0 : 1 \leq i \leq n-1\}.$$

It is generally 0-dimensional, with 2^{n-1} solutions. As $n \leq 4$ in practice, solving it is quasi-instantaneous and leads 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 8 *Let $\mathbb{F} = \mathbb{F}_{2^{kn}}$ and use the same notations as Proposition 7. If $H_g = 0$ or $\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}) \neq 0$, then there exists $x \in \mathbb{F}_{2^{kn}}$ such that $N_1(x) \in \mathbb{F}_{2^k}$.*

Proof First, we assume that d is even. 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 $N_1(\sqrt{x} + \sqrt{u_1}) \in \mathbb{F}_{2^k}$ 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 $\overline{N_1}(a) = a^2 + a + H_g^{-2}(u_1 + z)$. From [35, prop 3.79 p.127], 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 $z = \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_2}(u_1)$ if h_1 is monic. If it is not monic, "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 of the 2-polynomial $\mathrm{Tr}_{\mathbb{F}_{2^k}|\mathbb{F}_2}$, which is split [37] over $\mathbb{F}_{2^k}$. Let $\alpha \in \mathbb{F}_{2^k}$ be such a root, so that we want z such that $\alpha = \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}(u_1 + z))$. Properties of trace gives

$$\begin{aligned} \alpha + \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2} u_1) &= \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2} z) \\ &= z\,\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2}). \end{aligned}$$

With the hypothesis it is possible to write

$$z = \frac{\alpha + \mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2} u_1)}{\mathrm{Tr}_{\mathbb{F}_{2^{kn}}|\mathbb{F}_{2^k}}(H_g^{-2})} \in \mathbb{F}_{2^k}.$$

The proof for d odd is obtained by replacing any u_1 in the above by $u_1 + h_{2g}$. □

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 9 *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 8, we always find a value $a_d^* \in \mathbb{F}_{2^{kn}}$ such that $N_1(a_d^*) \in \mathbb{F}_{2^k}$ in this situation. The conclusion follows since $N_1(a_d^*) \in \mathbb{F}_{2^k}$ if and only if there exists a solution $(a_{d,1}^*, \ldots, a_{d,n-1}^*)$ of $\mathscr{S}_1$. □

Square Coefficients Using the previous notations, we have in characteristic 2

$$F(X) = p(X)^2 u(X) + p(X)q(X)h_1(X) + q(X)^2 w(X) = X^m + \sum_{i=0}^{m-1} N_{m-i}(a_1, \ldots, a_d) X^i, \tag{8}$$

with $p(X) = \sum_{i=0}^{d_1} a_{2i+1} X^i$ and $q(X) = \sum_{i=0}^{d_2} a_{2i+2} X^i \in (\mathbb{F}[a_1, \ldots, a_d])[X]$. Let

$$\mathscr{M} = \{a_i a_j : 1 \leq i \neq j \leq d\} \cup \{a_1, \ldots, a_d\} \text{ and } \overline{\mathscr{M}} = \{a_i^2 : 1 \leq i \leq d\}.$$

Then any monomial of $\mathscr{M}$ appearing in a $N_i \in \mathbb{F}[a_1, \ldots, a_d]$ in expression (8) 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. Hence, the number of such square coefficients depends only on h_1, or more precisely on its *length*.

Definition 10 (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 $d_P - i_P$.*

Proposition 11 *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 $\mathbb{F}$. Let $R \in \mathrm{Jac}(\mathscr{H})$ of weight g, and L_1 be the length of h_1. There are $g + 1 - L_1$ squares among the coefficients in X of the R-Decomposition polynomial.*

Proof Let $pq = \sum_{i=0}^{d-1} M_i X^i$ with $M_i \in \mathbb{F}[a_1,\ldots,a_d], \deg M_i = 2$ and $\deg_{a_j} M_i = 1$ for $0 \le j \le d$, and $h_1 = \sum_{i=i_h}^{d_h} H_i X^i$. Then the Cauchy product rule gives

$$pqh_1 = \sum_{i=i_h}^{d-1+d_h} \left(\sum_{j=0}^{i} M_j H_{i-j} \right) X^i = \sum_{i=i_h}^{d-1+d_h} C_i X^i, \tag{9}$$

with the convention that $M_d = \ldots = M_{d-1+d_h} = H_{d_h+1} = \ldots = H_{d-1+d_h} = 0$, and $C_i \in \mathbb{F}[a_1,\ldots,a_d]$. We have $\mathrm{Supp}\, C_i \subset \mathscr{M}$ for all $i_h \le i \le d-1+d_h$. Recall that $m-g-1 \le 2d_1 \le m-g$, $m-g-2 \le 2d_2 \le m-g-1$ and that $\deg w = g+1$. We let

$$up^2 = u \cdot \sum_{i=0}^{d_1} a_{2i+1}^2 X^{2i} = \sum_{i=0}^{2d_1+g} D_i X^i,$$
$$q^2 w = w \cdot \sum_{i=0}^{d_2} a_{2i+2}^2 X^{2i} = \sum_{i=0}^{2d_2+g+1} E_i X^i,$$

with $\mathrm{Supp}\, D_i \subset \overline{\mathscr{M}}$ and $\mathrm{Supp}\, E_i \subset \overline{\mathscr{M}}$ for all i, and $\deg D_i = \deg E_i = 2$. We can write the Decomposition polynomial F as

$$F(X) = \sum_{i=0}^{i_h-1} (D_i + E_i) X^i + \sum_{i=i_h}^{d-1+d_h} (C_i + D_i + E_i) X^i + \sum_{i=d+d_h}^{m-1} (D_i + E_i) X^i + X^m. \tag{10}$$

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 their definition, we see that $C_i = 0$ can only happen if $H_i = 0$ for all i, which is excluded by the fact that $\mathscr{H}$ is a binary hyperelliptic curve. Now the number of squares among the coefficients of F amounts is read on Expression (10) as $m - (d + d_h) + 1 + i_h = g + 1 - L_1$. □

Remark 12 *Since F is monic in general, the number of relevant squares among the coefficients of F is $g - L_1$.*

Additional squares depending on $LC(h_1)$ Note that N_1 is a square if and only if $\deg h_1 = d_h < g$. If it is the case, 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 $\mathscr{M}$. When $H_{d_h} \in \mathbb{F}_{2^k}$ we write

$$H_{d_h} a_d = H_{d_h} \left(a_{d,0} + \sum_{i=0}^{n-1} a_{d,i} t^i \right),$$

and observe that in $N_{1+g-d_h}(a_1,\ldots,a_d) = \sum_{j=0}^{n-1} N_{1+g-d_h,j}(\mathbf{a}) t^j$ 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 $1 \le j \le n-1$.

Remark 13 *All results hold for binary elliptic curves.*

2.3 Reducing the degree of PDP_{ng} systems

Let $\mathbb{F} = \mathbb{F}_{2^{kn}}$, and consider a $\mathbb{F}_{2^k}$-power basis $1, t, \ldots, t^{n-1}$ of $\mathbb{F}_{2^{kn}}$. 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 $d_h \le g$. Consider a PDP_{ng} instance for $R \in \mathrm{Jac}(\mathscr{H})$, with factor basis $\mathscr{B} = \{P - P_\infty : P \in \mathscr{H}, x(P) \in \mathbb{F}_{2^k}\}$, and let F be the Decomposition polynomial as in Equation (8). Following Proposition 7 we obtain by Weil descent a first system over $\mathbb{F}_{2^k}$

$$\mathscr{S}_1 = \{N_{1,i}(a_{d,1},\ldots,a_{d,n-1}) = 0 : 1 \le i \le n-1\},$$

and we let $\mathbf{a}^* = (a^*_{d,1}, \dots, a^*_{d,n-1})$ be a solution of $\mathscr{S}_1$, see Proposition 8. We evaluate the remaining equations at $\overline{\mathbf{a}} = (a_{1,0}, \dots, a_{1,n-1}, \dots, a_{d-1,n-1}, a_{d,0})$ to form the system

$$\mathscr{S}_2 = \{N_{i,j}(\overline{\mathbf{a}}, \mathbf{a}^*) : 2 \le i \le ng, 1 \le j \le n-1\}.$$

with $(ng-1)(n-1)$ variables and equations. Generally, this quadratic system is 0-dimensional and therefore generates an ideal of degree $2^{(ng-1)(n-1)}$.

When we start the Weil descent over $\mathbb{F}_{2^{kn}}$, the characteristic enables to replace square equtions by linear ones: 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 then write

$$\begin{aligned} N_i(a_1 \dots, a_d) = \sum_{j=0}^{n-1} N_{i,j}(\mathbf{a}) t^j &= \overline{N}_i(a_1 \dots, a_d)^2 \\ &= \left(\sum_{j=0}^{n-1} \overline{N}_{i,j}(\mathbf{a}) t^j \right)^2 \\ &= \sum_{j=0}^{n-1} \widetilde{N}_{i,j}(\mathbf{a})^2 t^j \end{aligned}$$

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 for all i, j

$$N_{i,j}(\mathbf{a}) = 0 \Leftrightarrow \widetilde{N}_{i,j}(\mathbf{a}) = 0 \Leftrightarrow \widetilde{N}_{i,j}(\overline{\mathbf{a}}, \mathbf{a}^*) = 0,$$

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 14 *Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be a hyperelliptic curve of genus g defined over $\mathbb{F}_{2^{kn}}$, with h_1 monic. Let L_1 be the length of h_1 and $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. The unsquared system $\sqrt{\mathscr{S}_2}$ related to R contains $(n-1)(g-L_1)$ linear equations.*

Proof Recall that $L_1 = d_h - i_h$, 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 11 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 11 but we do not use it to build $\sqrt{\mathscr{S}_2}$ since it was used for $\mathscr{S}_1$, so that $g - L_1 - 1$ square coefficients are used. Using the description before Remark 13, the Weil Descent gives us $n-1$ additional square equations in $\mathscr{S}_2$. Overall, this leads to $(n-1)(g-L_1)$ linear equations in $\sqrt{\mathscr{S}_2}$.

In any case, there are $(n-1)(g-L_1)$ 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 assumption is reasonable:

Genericity assumption 15 *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 assumption, the degree of $\mathscr{S}_2$ is generally divided by 2 with every linear equation replacing a quadratic one, and any linear equation can be used to eliminate a variable. A new system $\mathscr{S}_3$ is built that way, containing the remaining quadratic equations. If L_1 is the length of h_1, there are $(n-1)((n-1)g + L_1 - 1)$ variables and as much quadratic equations left in $\mathscr{S}_3$. Hence it is generally of dimension 0 and has degree:

$$\deg \mathscr{S}_3 = 2^{(n-1)((n-1)g+L_1-1)}.$$

As $0 \leq i_h \leq d_h \leq g$, we see that the best case happens when $L_1 = 0$ 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 lower bound

$$d_{\mathrm{opt}} = 2^{(n-1)((n-1)g-1)} \leq \deg \mathscr{S}_3.$$

We conclude by remarking that since h_1 encodes the 2-rank of $\mathrm{Jac}(\mathscr{H})$, then there should be a link between this reduction and the action of 2-torsion elements over the set of decompositions of a given R, analogous to the one exploited in [17].

2.4 Analysis of the degree reduction for genus 2 binary curves

We conclude this Section with an exhaustive analysis of genus 2 binary curves, summed up in Table 1. Such curves are classified in three types depending on the rank of the two-torsion subgroup in $\mathrm{Jac}(\mathscr{H})$ — see Section 2.4 for details. Among such curves, Type II curves were particularly highlighted in [4] because of their lower cost arithmetic. Our study reveals that they are as weak as Supersingular curves (type *III*) considering Decomposition attacks.

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 $h_0(x) = x^5 + \sum_{i=0}^{4} f_i x^i$. Let $t \in \mathbb{F}_{2^n}$ an element of absolute trace 1, and $\varepsilon \in \mathbb{F}_2$, and see [4][42] for details on their definition. There are three types of binary genus 2 curves, depending 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. 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*, and therefore weak to the Frey-Rück attack [19] 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. We focus on the above forms for genus 2 binary curves, and we call them *canonical forms* in the rest of the presentation.

Comparisons of degree reductions depending on canonical forms Table 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 reduced degree resp. 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 $\text{LC}(h_1)$ 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 8 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 1 Degree reduction in genus 2 for small extension fields

Type	$\deg h_1$	L_1	n	Univariate	Square	$\text{LC}(h_1)$	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 [4], and we use this to design a practical Discrete Logarithm computation for realistic parameters, see Section 5.3.

It is also worth mentioning that if $g = 2, n = 4$ and while the computation time is not practical (more than 250 hours with Magma 2.19), it is now possible to solve a given PDP_8 instance on a Type II curve by solving ideals of degree 2^{15} instead of 2^{24}, a number of solutions previously too high to even consider a try.

The case $h_1(x) = x^2$: In our situation, the length of h_1 is the principal indicator of the reduction factor we can obtain. In particular, curves of type I_b with $h_1(x) = x^2$ are such that $L_1 = 0$ and therefore we can expect the best reduction factor. If $h_1(x) = x^2$, then $f_1 \neq 0$ or else it can be verified that the curve has a simple singularity at $(0, \sqrt{f_0})$, and so has genus 1. It can be checked that genus 2 curves with $h_1(x) = x^2$ are isomorphic to type II curves using the change of variables $x = 1/x'$ and $y = y'/x'^3 + \sqrt{f_0}$. However, as seen in Table 1, small differences appears depending on the chosen model. This is why we choose to distinguish the two cases.

3 Summation sets and PDP_m solving

This section introduces an alternate modelling of PDP_{ng} instances in all genus, derived from Gaudry and Diem's usage of elliptic *Summation polynomials* introduced in [40]. We generalize this notion to hyperelliptic curves

and mention that the presentation could be extended to any curves. As our description enables algorithmic computations of these new objects, we give thereafter simple examples and timings for experiments. The Section ends with some discussions on the impact of the canonical double cover of degree 2 given by the projection over the abscissa for any hyperelliptic curve regarding the computations of Summation Polynomials and their usage in solving PDP_m instances.

3.1 Geometric description of PDP_m setting

To simplify the presentation we assume that the base field $\mathbb{F}$ is algebraically closed, but the whole presentation extends to any fields. During this Section we fix an hyperelliptic curve $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ in imaginary model defined over $\mathbb{F}$ and of genus g, and a reduced divisor $R = (u,v) \in \mathrm{Jac}(\mathscr{H})$. Following Remark 4, we consider integers $m \geq g+1$. It is clear that the order of the points in a decomposition as $R = P_1 + \ldots + P_m - mP_\infty$ does not matter. This means the mth-symmetric group $\mathfrak{S}_m$ acts on the set of all such decomposition. This prompts the next Definition.

Definition 16 *The algebraic variety $\mathscr{V}_{m,R} = \{ (P_1,\ldots,P_m) : \sum_{i=1}^m P_i - mP_\infty = R \}/\mathfrak{S}_m$ is called the m-Summation Variety associated with R, or the m-Summation Variety if the context is clear.*

The following description will allow us to compute "symmetrized" polynomials that generates this variety, that is to say, polynomials whose variables describes symmetric expression of the standard variables. Let $\pi : \mathscr{H}^m \longrightarrow (\mathbb{P}^1)^m$ be the map induced by the double cover $x : \mathscr{H} \longrightarrow \mathbb{P}^1$. Our first goal is to describe $\mathscr{V}_{m,R}$ in general and its projection "on the x-line" $\pi(\mathscr{V}_{m,R})$. We give a description for R of weight g as it is the usual case, but it can be extended to any R straightforwardly. From Section 2, a generic function normalized at infinity in $\mathscr{L}(mP_\infty - R)$ is written as $f(X,Y) = p(X)u(X) + (Y - v(X))q(X)$, with $p(X) = \sum_{i=0}^{d_1} a_{2i+1}X^i$ and $q(X) = \sum_{i=0}^{d_2} a_{2i+2}X^i$. We have $d_1 + d_2 = m - g - 1 = d - 1$ and we let $\mathbf{a} = (a_1,\ldots,a_d)$. From Section 2.2, the Decomposition polynomial is the monic polynomial in $(\mathbb{F}[\mathbf{a}])[X]$ given by

$$F(X) = \frac{N(f)}{u(X)} = (-1)^{m-g}(up^2 - pq(2v+h_1) + q^2w) = X^m + \sum_{i=0}^{m-1} N_{m-i}(\mathbf{a})X^i, \tag{11}$$

with $\deg N_i = 2$ for $1 \leq i \leq m$. Assume now that f describes a decomposition of R as $P_1 + \cdots + P_m - mP_\infty = R$ and let $\mathbf{x} = (x(P_1),\ldots,x(P_m))$. We know that F vanishes exactly at all the $x(P_i)$'s so we can write

$$F(X) = \prod_{i=1}^{m}(X - x(P_i)) = X^m + \sum_{i=0}^{m-1}(-1)^{m-i}E_{m-i}(\mathbf{x})X^i, \tag{12}$$

where E_i denotes the ith elementary symmetric polynomial in m variables. Let $\mathbf{e} = (e_1,\ldots,e_m)$ be variables standing for these symmetric expressions. Equating coefficients of (11) and (12) we obtain a polynomial ideal $\mathscr{I}_{m,R}$ generated by

$$\begin{cases} e_1 = N_1(\mathbf{a}), \\ \quad\vdots \\ e_m = (-1)^m N_m(\mathbf{a}), \end{cases} \tag{13}$$

of m equations in $2m - g$ variables. We claim that $\mathscr{V}_{m,R}$ is (isomorphic to) the variety associated to $\mathscr{I}_{m,R}$.

Proposition 17 *Let $\mathscr{H}$ be a hyperelliptic curve in imaginary model of genus g, and let $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. For any $m \geq g+1$, define $\mathscr{I}_{m,R}$ as the ideal in $\mathbb{F}[\mathbf{a},\mathbf{e}]$ generated by system (13). The Summation Variety $\mathscr{V}_{m,R}$ is isomorphic to $V(\mathscr{I}_{m,R})$. It is an irreducible variety and its associated ideal is $\mathscr{I}_{m,R}$.*

Proof If $((x_1,y_1),\dots,(x_m,y_m)) \in \mathscr{V}_{m,R}$, then there is a $f \in \mathscr{L}(mP_\infty - R)$, unique if normalized at infinity, such that $\operatorname{div} f + R = \sum_{i=1}^m P_i - mP_\infty$, hence $\mathscr{V}_{m,R} \subset V(\mathscr{I}_{m,R})$. For the reverse inclusion, let $(a_1,\dots,a_d,e_1,\dots,e_m)$ be in $V(\mathscr{I}_{m,R})$. As $\mathbb{F}$ is algebraically closed, we find $x_1,\dots,x_m$ such that $\prod_{i=1}^m (X-x_i) = X^m + \sum_{i=0}^{m-1}(-1)^{m-i}e_{m-i}X^i$. 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 $\sum_{i=1}^m P_i - mP_\infty = R$. First we specialize a generic function f with $a_1,\dots,a_d$. This gives an element $f \in \mathscr{L}(mP_\infty - R)$. Next, provided that $q(x_i) \neq 0$, we can set $y_i = \frac{v(x_i)q(x_i) - p(x_i)u(x_i)}{q(x_i)}$ for $1 \le i \le m$ and check that $P_i \in \mathscr{H}$ and that $f(x_i,y_i) = 0$.

If $q(x_i) = 0$ for at least one i, then the expression of the generic function in $\mathscr{L}(mP_\infty - R)$ implies that $u(x_i)p(x_i) = 0$. If $u(x_i) = 0$, then $y_i = v(x_i)$ by properties of Mumford representation. Else, then $p(x_i) = 0$, which means $f(x_i,y) = 0$ for all $y \in \mathbb{F}$, so that $f(x,y) = (x-x_i)\tilde{f}(x,y)$ with $\tilde{f} \in \mathscr{L}((m-2)P_\infty)$. Since $\mathbb{F}$ is algebraically closed[3], 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 and $-P_i$.

The ideal $\mathscr{I}_{m,R}$ is an example of a polynomial parametrization. It is known [8, Prop. 5, p. 199] that such ideals are always primes, and therefore radical. □

We now define *Summation sets*:

Definition 18 (Summation polynomials for hyperelliptic curves) *Let $\mathscr{H}$ be a hyperelliptic curve of genus g given by a Weierstrass equation $y^2 + h_1(x) = h_0(x)$, $m \ge g+1$ and $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. The mth summation ideal associated to $\mathscr{H}$ and R is defined as the elimination ideal $\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}]$ where $\mathscr{I}_{m,R}$ is the ideal in $\mathbb{F}[\mathbf{a},\mathbf{e}]$ generated by equations* (13). *Any (finite) set $\mathbb{S}_{m,R} \subset \mathbb{F}[\mathbf{e}]$ generating $\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}]$ is called a set of mth Summation polynomials, or a mth summation set, associated with R.*

The ideal $\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}]$ essentially describes $\pi(\mathscr{V}_{m,R})$. To "compute" this projection, we usually compute a Gröbner basis of $\mathscr{I}_{m,R}$ for an adequate elimination order (we refer to [8] for a description of the corresponding notions). For any set $\mathbb{S}$ of polynomials we denote by $\mathbb{S}(x_1,\dots,x_m)$ the set of all elements in $\mathbb{S}$ evaluated at $(x_1,\dots,x_m)$. The next proposition generalizes a result known for elliptic curves:

Proposition 19 *For any $m \ge g+1$, the variety $V(\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}])$ is irreducible, and its associated ideal is the Summation ideal. A set $\mathbb{S}_{m,R}$ of mth Summation polynomials associated to R exists, and it verifies:*

$$\mathbb{S}_{m,R}(e_1,\dots,e_m) = 0 \Leftrightarrow \exists\, P_i = (x_i,y_i) \in \mathscr{H}, 1 \leqslant i \leqslant m, \text{ such that } e_i = E_i(x_1,\dots,x_m) \text{ and } P_1 + \dots + P_m - mP_\infty = R.$$

Proof It is shown in the proof of [8, Prop. 3, p. 347] that relation ideals associated with polynomial parametrizations are prime. The (mostly) technical part of the proof is to show that if $h(N_1(\mathbf{a}),\dots,N_m(\mathbf{a})) = 0$, then h is in $\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}]$: this tells that the elimination ideal is the relation ideal, in other words that $\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}] = \{g \in \mathbb{F}[\mathbf{e}] \,:\, g(N_1(\mathbf{a}),\dots,N_m(\mathbf{a})) = 0\}$. We reproduce the (more interesting) argument of [8, Prop.1, p.344] for primality, for the sake of completeness. Let gh be in $\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}]$. Then $g(N_1(\mathbf{a}),\dots,N_m(\mathbf{a}))h(N_1(\mathbf{a}),\dots,N_m(\mathbf{a})) = 0$ in the integral domain $\mathbb{F}[\mathbf{a}]$. This means g or h is in $\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}]$. Thus $V(\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}])$ is irreducible, and the Summation ideal is the ideal associated to $V(\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}])$. This proves the first statement. The existence of Summation sets comes from Hilbert Basis theorem. Now if $\mathbf{e}$ is in $V(\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}])$, according to the extension theorem [8, p. 118] we can find $\mathbf{a} = (a_1^*,\dots,a_d^*)$ such that $(\mathbf{a},\mathbf{e}) \in V(\mathscr{I}_{m,R})$. The conclusion comes from Proposition 17. □

Remark 20 *Geometrically, a Summation set $\mathbb{S}_{m,R}$ satisfies $V(\mathbb{S}_{m,R}) = V(\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}])$.*

We briefly discuss the cardinality of Summation sets, assuming for simplicity that we are in a generic situation. Being described by m equations in a $2m-g$ dimensional space, $\mathscr{V}_{m,R}$ has dimension $m-g$, so $\pi(\mathscr{V}_{m,R})$ has codimension g in an ambient space of dimension m. This means that a minimal generating family for a Summation ideal should have at least g elements. These varieties seem far from being complete intersections, as our experiments in the next Section suggest.

[3] In the general case, we look for y_i in the algebraic closure of $\mathbb{F}$.

When $g = 1$, the Summation ideal is principal, which "shows" that the mth summation polynomial is unique (up to a constant) in the elliptic case. Proposition 19 gives an another proof of the irreducibility of elliptic Summation polynomials.

While this presentation focuses on the hyperelliptic case, it can be adapted to non-hyperelliptic curves as well by using bases of $\mathscr{L}(m\mathscr{O} - R)$, where $\mathscr{O}$ is a distinguished point of the curve. We ran some experiments for superelliptic curves and $\mathscr{C}_{a,b}$ curves of small genus but we did not investigate further as such curves are not considered in practice.

3.2 Examples of summation sets and experiments

Using the Magma code from the URL we provided, one can compute Summation polynomials for elliptic curves and confirms that, following the process described in the previous Section, the elliptic Summation polynomial from [40] is obtained. Here we give Summation sets in genus 2, for the smallest possible sum (of length 3). We then report experimental timings for computation of Summation sets.

3.2.1 First summation polynomials 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 3.1, the smallest decomposition is obtained for $m = g + 1 = 3$. For a fixed $R = (u, v)$ of weight g in $\mathrm{Jac}(\mathscr{H})$, a convenient $\mathbb{F}_q$-basis of $\mathscr{L}(3P_\infty - R)$ is $\{u, y - v\}$, and we have $d_1 = d_2 = 0$, $d = m - g = 1$. With the previous notations, this means $p(X) = a_1$ and $q(X) = 1$. Let $u = X^2 + u_1X + u_0$ and $v = v_1X + v_0$ to find

$$w = -X^3 + u_1X^2 + (u_0 - h_3 - u_1^2)X + (u_1^3 + h_3u_1 - 2u_1u_0 + v_1^2 - h_2).$$

Hence the Decomposition polynomial can be written

$$\begin{aligned} F(X) &= (-1)^d(up^2 - 2pqv + q^2w) \\ &= X^3 - (a_1^2 + u_1)X^2 + (2a_1v_1 + u_1^2 + h_3 - u_0 - a_1^2u_1)X + 2a_1v_0 + 2u_1u_0 + h_2 - a_1^2u_0 - u_1^3 - h_3u_1 - v_1^2, \end{aligned}$$

and can also be expressed as

$$F(X) = X^3 - e_1X^2 + e_2X - e_3.$$

Equating coefficients gives the following system:

$$\begin{cases} e_1 = a_1^2 + u_1, \\ e_2 = 2a_1v_1 + u_1^2 + h_3 - u_0 - a_1^2u_1, \\ e_3 = u_1^3 + h_3u_1 + v_1^2 - 2a_1v_0 - 2u_1u_0 - h_2 + a_1^2u_0. \end{cases}$$

Treating the parameters u_i, v_i, h_i as non-zero numbers, that is to say, compute a Gröbner basis over an adequate function field, we obtain the following "symbolic" Summation polynomials after elimination of a_1, assuming $e_3 > e_2 > e_1$ and that the order for this block of variables is grevlex:

$$\begin{aligned} S_{5,1} &= e_2^2 + 2u_1e_2e_1 + u_1^2e_1^2 + (-2h_3 - 4u_1^2 + 2u_0)e_2 + (-2h_3u_1 - 4u_1^3 + 2u_1u_0 - 4v_1^2)e_1 + h_3^2 + 4h_3u_1^2 - 2h_3u_0 \\ &\quad + 4u_1^4 - 4u_1^2u_0 - 4u_1v_1^2 + u_0^2, \\ S_{5,2} &= v_1e_3 + v_0e_2 + (u_1v_0 - u_0v_1)e_1 + 3u_1u_0v_1 + u_0v_0 + h_2v_1 - h_3u_1v_1 - h_3v_0 - u_1^3v_1 - 2u_1^2v_0 - v_1^3. \end{aligned}$$

We observe that if both $v_0 = v_1 = 0$, then $S_{5,2}$ above is always zero. Then the result of the Gröbner basis computation will not be the one displayed here. However, this case rarely happen, as $v_1 = v_0 = 0$ implies either R is a 2-torsion element in $\mathrm{Jac}(\mathscr{H})$. In odd characterstic, there is at most 16 such elements. For the sake of clarity we do not display the Summation set obtained in this special case. Instead, we give an instantiated

example on very small parameters. Let $\mathbb{F} = \mathbb{F}_{31}$, and $\mathscr{H} : y^2 = x^5 + 6x^3 + 27x^2 + 11x + 29$. We find $P_1 = (20,17), P_2 = (17,7), P_3 = (4,11)$ as rational points in $\mathscr{H}$, and the divisor $P_1 + P_2 + P_3 - 3P_\infty$ reduces to $R = (X^2 + 13X + 9, 15X + 12)$. This gives the Summation set

$$\begin{cases} e_2^2 + 26e_2e_1 + 14e_1^2 + 26e_2 + 27e_1 + 27, \\ e_3 + 7e_2 + 20e_1 + 15. \end{cases}$$

One checks that both polynomials vanish at the symmetric expressions of the $x(P_i)$'s.

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_{1j} is the jth coefficient of h_1, we obtain the following parametrization

$$\begin{cases} e_1 = a_1^2 + h_{12}a_1 + h_4 + h_2 + u_1, \\ e_2 = u_1a_1^2 + h_{11}a_1 + h_4u_1 + h_3 + h_2u_1 + h_{12}v_1 + u_1^2 + u_0, \\ e_3 = u_0a_1^2 + h_{10}a_1 + h_4u_1^2 + h_4u_0 + h_3u_1 + h_2u_1^2 + h_2u_0 + h_{12}u_1v_1 + h_{12}v_0 + h_{11}v_1 + u_1^3 + v_1^2. \end{cases}$$

We first compute a Gröbner basis of the elimination ideal as in the previous paragraph and obtain:

$$\begin{aligned} S_{5,1} = {} & e_2^2 + u_1^2e_1^2 + (h_{12}^2u_1 + h_{12}h_{11})e_2 + (h_{12}h_{11}u_1 + h_{11}^2)e_1 + h_4h_{12}^2u_1^2 + h_4h_{11}^2 + h_3^2 + h_3h_{12}^2u_1 + h_3h_{12}h_{11} \\ & + h_2h_{12}^2u_1^2 + h_2h_{11}^2 + h_{12}^3u_1v_1 + h_{12}^2h_{11}v_1 + h_{12}^2u_1^3 + h_{12}^2u_1u_0 + h_{12}^2v_1^2 + h_{12}h_{11}u_0 + h_{11}^2u_1 + u_0^2, \\ S_{5,2} = {} & (h_{12}u_1 + h_{11})e_3 + (h_{12}u_0 + h_{10})e_2 + (h_{11}u_0 + h_{10}u_1)e_1 + h_4h_{12}u_1^3 + h_4h_{11}u_1^2 + h_3h_{12}u_1^2 + h_3h_{12}u_0 \\ & + h_3h_{11}u_1 + h_3h_{10} + h_2h_{12}u_1^3 + h_2h_{11}u_1^2 + h_{12}^2u_1^2v_1 + h_{12}^2u_1v_0 + h_{12}^2u_0v_1 + h_{12}h_{11}v_0 + h_{12}h_{10}v_1 + h_{12}u_1^4 \\ & + h_{12}u_1^2u_0 + h_{12}u_1v_1^2 + h_{12}u_0^2 + h_{11}^2v_1 + h_{11}u_1^3 + h_{11}u_1u_0 + h_{11}v_1^2 + h_{10}u_0. \end{aligned}$$

One can instantiate these formula on small parameters to check the vanishing of the Summation set. For a type II genus 2 curve over $\mathbb{F}_{2^d}$ with d odd, an equation is $y^2 + xy = x^5 + f_3x^3 + \varepsilon x^2 + f_0$, $\varepsilon \in \mathbb{F}_2$, see 2.4. Then a Summation set is way sparser:

$$\begin{aligned} S_{5,1}(e_1,e_2,e_3) &= e_2^2 + u_1^2e_1^2 + e_1 + h_3^2 + u_1 + u_0^2, \\ S_{5,2}(e_1,e_2,e_3) &= e_3 + u_0e_1 + h_3u_1 + u_1^3 + u_1u_0 + v_1^2 + v_1 + 1. \end{aligned}$$

The expressions of those summation polynomials are also very similar to the genus 1 case.

3.2.2 Computation Timings

Timings in odd characteristic Table 2 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 2.19 [2], on a Intel® Xeon®@2.93GHz processor. The time is expressed in seconds, and averaged over several curves. Next columns give the average number (rounded) of monomials and average total degree of elements in the summation set. The degree is computed considering that $\deg e_i = i$. When a Summation set $\mathbb{S}_{m,R}$ can be computed, we also compute $\deg V(\mathbb{S}_{m,R})$ 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 GB.

genus g	m	#vars	Time	$\#\mathbb{S}_{m,R}$	Avg. len.	Avg. deg.	$\deg V(\mathbb{S}_{m,R})$
2	3	4	0.000s	2	5	4	2
	4	6	0.000s	7	28	10	4
	5	8	0.18s	13	248	21	8
	6	10	3505s	130	5901	50	16
3	4	5	0.000s	3	5	4	2
	5	7	0.000s	6	16	8	4
	6	9	0.22s	45	159	19	8
	7	11	54.3s	194	2028	36	16
	8	13	-	-	-	-	32
4	5	6	0.00s	4	5	4	2
	6	8	0.00s	7	15	8	4
	7	10	0.03s	24	80	15	8
	8	12	-	-	-	-	16

Table 2 Computations of Specialized Summations Sets in odd characteristic

Timings in even characteristic In Table 3 we report computation times for the first summation sets for binary hyperelliptic curve 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, we observe that the use of canonical forms speeds up the computation and lead to sparser sets, because less non-zero coefficients in the curve's equation means less monomials in the support of the parametrization of $\mathscr{V}_{m,R}$. 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.

genus g	m	#vars	Time	$\#\mathbb{S}_{m,R}$	Avg. len.	Avg. deg.	$\deg V(\mathbb{S}_{m,R})$
2	3	4	0.000s	2	5	4	2
	4	6	0.000s	3	14	8	4
	5	8	0.03s	5	89	17	8
	6	10	12.7s	15	1032	36	16
	7	12	-	-	-	-	-
3	4	5	0.000s	3	4	4	2
	5	7	0.000s	4	12	7	4
	6	9	0.1s	6	46	13	8
	7	11	0.89s	14	276	23	16
	8	13	-	-	-	-	32
4	5	6	0.00s	4	4	4	2
	6	8	0.00s	5	11	7	4
	7	10	0.01s	7	40	12	8
	8	12	0.3s	12	127	19	16
	9	14	-	-	-	-	-

Table 3 Computation of Specialized Summation Sets in even characteristic

As for elliptic Summation polynomials, computations are easier to complete in even characteristic, and the summations sets' elements are sparser and fewer.

3.3 Degree of Summation Ideals

In general, the degree of an algebraic variety can be defined as the number of elements in a "generic enough" subvariety of dimension 0. If the variety is an hypersurface, then it is also the (total) degree of a defining polynomial. Because we are interested in solving 0-dimensional systems linked to Summation varieties, we need at least an estimation of $\deg \mathscr{V}_{m,R}$. This is the purpose of this Section.

Let R be a point on an elliptic curve, and let $S_{m,R} = S_{m+1}(X_1,\dots,X_m,x(R))$ be the $m+1^{th}$ elliptic Summation polynomial evaluated at $x(R)$. It is known [10] that $\deg S_{m,R} = 2^{m-1}$. This fact added to the last column of Tables 2 and 3 leads us to the following conjecture.

Conjecture 21 *Let $\mathscr{H}$ be an hyperelliptic curve of genus $g \geq 2$, $R \in \mathrm{Jac}(\mathscr{H})$ of weight g and $m \geq g+1$. The degrees of the m-Summation variety $\mathscr{V}_{m,R}$ and its projection $V(\mathbb{S}_{m,R})$ are:*

$$\deg \mathscr{V}_{m,R} = \deg V(\mathbb{S}_{m,R}) = 2^{m-g}.$$

Conjecture 21 is strengthened by the following informal discussion, where it is assumed that the base field is algebraically closed. For an imaginary hyperelliptic curve $\mathscr{H}$ in a Weierstrass model and $P \in \mathscr{H}$, we denote by $-P$ the image of P by the canonical hyperelliptic involution $[-]$. If $x : \mathscr{H} \longrightarrow \mathbb{P}^1$ is the double cover given by the abscissa, for all $m \in \mathbb{N}^*$, we denote by $\pi : \mathscr{H}^m/\mathfrak{S}_m \longrightarrow (\mathbb{P}^1)^m/\mathfrak{S}_m$ the induced cover of degree 2^m . Let $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. With notations of Section 3.1, Proposition 19 tells us that $\pi(\mathscr{V}_{m,R}) = V(\mathscr{I}_{m,R} \cap \mathbb{F}[\mathbf{e}]) = V(\mathbb{S}_{m,R})$ for any summation set[4] $\mathbb{S}_{m,R}$. Overall we have a commutative diagram

$$\begin{array}{ccc} \mathscr{V}_{m,R} & \hookrightarrow & \mathscr{H}^m/\mathfrak{S}_m \\ \downarrow & & \downarrow \pi \\ V(\mathbb{S}_{m,R}) & \hookrightarrow & (\mathbb{P}^1)^m/\mathfrak{S}_m \end{array}$$

If we consider a vanishing sum as $P_1 + \cdots + P_m - mP_\infty = R$ on a genus g curve, then once $m-g$ points have been fixed the last g points are generally determined. In other words, $\dim V(\mathbb{S}_{m,R}) = m-g$ and if $(e_1,\dots,e_{m-g}) \in \mathbb{F}^{m-g}$ are given, then it determines $e_{m-g+1},\dots,e_m$ such that $(e_1,\dots,e_m) \in V(\mathbb{S}_{m,R})$. With a slight abuse of notations, the fiber $\pi^{-1}(\{e_1,\dots,e_{m-g}\})$ has 2^{m-g} elements, that all lead to a decomposition of R. While this is just a sketch of proof and intuitive thinking, it strongly suggests that $\deg \mathscr{V}_{m,R} = 2^{m-g}$.

Now, whenever $(e_1,\dots,e_m)$ is given in $V(\mathbb{S}_{m,R})$, then it determines the two decompositions $R = P_1 + \ldots + P_m - mP_\infty$ and $-R = (-P_1) + \cdots + (-P_m) - mP_\infty$. The latter is an element of the larger variety $\mathscr{V}' = \{(P_1,\dots,P_m) : \sum_{i=1}^m P_i - mP_\infty = \pm R\}$, so the previous sentence informally says that $\mathscr{V}'/[-] \simeq \mathscr{V}_{m,R}$ and that the projection $\overline{\pi} : \mathscr{V}' \longrightarrow V(\mathbb{S}_{m,R})$ has degree at least 2. It is possible to show that it has degree 2. Factoring this map through the quotient, we obtain that $\mathscr{V}_{m,R}$ is birationally equivalent to $V(\mathbb{S}_{m,R})$, the map giving the equivalence being the restriction of π to $\mathscr{V}_{m,R}$, so $\deg V(\mathbb{S}_{m,R}) = 2^{m-g}$.

3.4 Using Summation polynomials for PDP_m solving

In this Section we consider fields as $\mathbb{F}_{q^n}$, and we fix a hyperelliptic curve $\mathscr{H}$ of genus g, and $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. Solving the PDP_{ng} related to R with the factor base $\mathscr{B} = \{P - P_\infty \ : \ P \in \mathscr{H}, x(P) \in \mathbb{F}_q\}$ can be done following [10][23] with a Weil Descent, which means we want to find points in a 0-dimensional subvariety of the Weil restriction of $V(\mathbb{S}_{m,R})$.

Degree of Weil restrictions For a variety V defined over $\mathbb{F}_{q^n}$ we denote by $\mathscr{W}_n(V)$ its Weil Restriction over $\mathbb{F}_q$. It is an algebraic variety defined over $\mathbb{F}_q$ with $\dim_{\mathbb{F}_q} \mathscr{W}_n(V) = n \cdot \dim_{\mathbb{F}_{q^n}} V$ and $\deg \mathscr{W}_n(V) = (\deg V)^n$. If a generating set $\mathbb{S}$ for the (radical) ideal I associated with V is given, we also use the notation $\mathscr{W}_n(\mathbb{S})$ or $\mathscr{W}_n(I)$. If Conjecture 21 holds, then $\deg \mathscr{W}_n(\mathbb{S}_{m,R}) = 2^{n(m-g)}$ in general, and in Decomposition attacks where $m = ng$, we obtain

$$\deg \mathscr{W}_n(\mathbb{S}_{m,R}) = 2^{n(n-1)g} = d_{\mathrm{Nag}}.$$

This has to be expected since we used the Decomposition polynomial to both describe Nagao's approach and Summation sets.

[4] In particular, Summation ideals depend on the choice of the double cover. When $g = 1$, the authors of [17] use the fact 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.

A new solving algorithm for PDP_{ng} instances Let $m = ng$, $\mathbf{e} = (e_1, \dots, e_m)$, $\bar{\mathbf{e}} = (e_{1,1}, \dots, e_{1,n}, \dots, e_{m,1}, \dots, e_{m,n})$. The solving algorithm is straightforward:

1. Compute a Summation set $\mathbb{S}_{m,R} = \{S_1, \dots, S_r\} \subset \mathbb{F}_{q^n}[\mathbf{e}]$.
2. Using a power basis $1, t, \dots, t^{n-1}$ of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$, write $S_i(\mathbf{e}) = \sum_{j=0}^{n-1} S_{i,j}(\bar{\mathbf{e}}) t^j$, and build the system $\{S_{1,0}, \dots, S_{1,n-1}, \dots, S_{r,0}, \dots, S_{r,n-1}\} \subset \mathbb{F}_q[\bar{\mathbf{e}}]$.
3. Build the system $\mathscr{S} = \{S_{i,j}(e_1, 0, \dots, 0, e_2, 0, \dots, 0, \dots, e_m, 0, \dots, 0)\ \forall\, i, j\}$ by evaluation.
4. Solve $\mathscr{S}$ over $\mathbb{F}_q$ with the standard Gröbner bases strategy.
5. For all solutions $(e_1^*, \dots, e_m^*)$ of $\mathscr{S}$:
   - check if $F^*(X) = X^m + \sum_{i=0}^{m-1} (-1)^{m-i} e_{m-i}^* X^i$ is split over $\mathbb{F}_q$.
     - If it is, build the associated decomposition of R.

Steps 2, 3 and 5 are usually done in time negligible compared to the others. While step 1 can become a blocking step, as highlighted by the timings in Section 3.2.2, we assume it finishes in reasonable time compared to step 4 for the sake of the following analysis. As $\operatorname{codim} V(\mathbb{S}_{m,R}) = g$, we deduce that $\mathscr{S}$ contains more than $m = ng$ equations. By construction, it also depends on at most ng variables. The following general assumption is natural, and it was always true in our experiments.

Genericity assumption 22 *The Weil descent on Summation varieties produces* 0*-dimensional systems.*

We already stated that the number of solutions of $\mathscr{S}$ is a good indicator of the complexity of its solving. By definition of the degree of a variety, it follows that $\mathscr{S}$ has precisely $\deg \mathscr{W}_n(\mathbb{S}_{m,R}) = 2^{n(n-1)g}$ solutions. The efficiency of this algorithm should be really close to that of Nagao's, provided step 1 finishes in a reasonably short time.

3.5 On computational aspects of Summation sets

We discuss the impact of the curve's genus and the degree of the projection $\pi : \mathscr{H}^m \longrightarrow (\mathbb{P}^1)^m$ induced by the projection over the abscissae in the computation of Summation set.

Obstruction for recursive computations Semaev proposed [40] a recursive approach for computing summation polynomials for a genus 1 curve E is found by decomposing a sum into two smaller sums:

$$P_1 + \cdots + P_m = \mathscr{O} \Leftrightarrow \ \forall\, k \in \{2, \dots, m-3\},\ \exists\, Q \in E(\overline{\mathbb{F}}) : \begin{cases} P_1 + \cdots + P_k = Q \\ P_{k+1} + \cdots + P_m = -Q \end{cases}$$

Using X as an indeterminate for the abscissae of the intermediate summand Q and $x_i = x(P_i)$, we deduce that $S_{k+1}(x_1, \dots, x_k, X)$ and $S_{m-k+1}(x_{k+1}, \dots, x_m, X)$ have a common root. Hence their resultant with respect to X must vanish. If we see S_k and S_{m-k+1} in $\mathbb{F}[X_1, \dots, X_m, X]$, then geometrically this corresponds to the projection of $V(S_k(X_1, \dots, X_k, X)) \cap V(S_{m-k+1}(X_{k+1}, \dots, X_m, X))$ on the m first coordinates. In general, both varieties are hypersurfaces in a $m+1$-dimensional space. Thus their intersection has dimension $m-1$. The projection on a m-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.

However this observation cannot be generalized in higher genus to obtain a recursive method of computation. Because a fiber as $\pi^{-1}(\{x_1, \dots, x_g\})$ has cardinality 2^g, the projection of $V(\mathbb{S}_k) \cap V(\mathbb{S}_{m-k-2})$ describes more than the vanishing sums of m points (or their opposite). 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 3.1, 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 4 could be computed in genus 2 — the running time of the computation was longer than with the method of Section 3.1. This set of polynomials indeed vanished on sums of length m as well as other.

Usage in PDP$_m$ solving In the algorithm of the previous Section, the first step is to compute a Summation set related to a given $R \in \mathrm{Jac}(\mathscr{H})$. This step can dominate the whole routine, and its complexity is hard to derive, as not much is known on the cost of computing Gröbner bases for elimination orders. Several strategies can be used to speed-up this computation, such as eliminating variables in several steps instead oof one. However, such strategies are mainly based on observations made on the behaviour on the computation and the shape of a particular system.

Another approach would be to compute a more general type of Summation sets, in the spirit of what is done for Decomposition attacks over elliptic curves. More precisely, we can compute once and for all a generating set for the projection of the variety $\mathscr{V}_m = \{(P_1,\ldots,P_{m+g}) \,:\, \sum P_i - (m+g)P_\infty = \mathscr{O}\}$, where $\mathscr{O}$ is the neutral element of $\mathrm{Jac}(\mathscr{H})$, then evaluate it at the "coordinates" of an $R \in \mathrm{Jac}(\mathscr{H})$ that we try to decompose. Describing this variety can be done straightforwardly following the presentation of Section 3.1, but considering the generic norm instead of the Decomposition polynomial. While such sets can be used to find decompositions of R, from the point of view of the polynomial system solving, this approach will always be less efficient because $\pi(\mathscr{V}_m)$ has degree greater than $V(\mathbb{S}_{m,R})$ in general.

Indeed, let $\mathbb{S}_m$ be a generating set for the variety $\pi(\mathscr{V}_m)$. Assume R is represented by the reduced divisor $R_1+\ldots+R_g-gP_\infty$, and let H be the intersection of the hyperplanes describing the symmetric expression in the $x(R_i)$'s. This way, computing $\mathbb{S}_m$ then specializing it at the $x(R_i)$'s amounts to working in the variety $V(\mathbb{S}_m)\cap H$. In general, the fiber $\pi^{-1}(\pi(R)) = \{\pm R_1 \cdots \pm R_g\}$ contains 2^g elements, so as soon as $g \geq 2$, the previous variety describes more tuples of points than we actually need, since we are only interested in the decomposition of R (or $-R$). More precisely, we have generally

$$\deg(V(\mathbb{S}_m)\cap H) = 2^{g-1}\cdot \deg V(\mathbb{S}_{m,R}). \tag{14}$$

First, this shows that working with $V(\mathbb{S}_m)\cap H$ and $V(\mathbb{S}_{m,R})$ is equivalent in genus 1. In fact it can be shown that these varieties are equal: both are hypersurfaces of same degree, and the latter is (informally) included in the former. In particular, if S_{m+1} denotes the $m+1$ elliptic (symmetrized) Summation polynomial, it implies that $S_{m,R}(e_1,\ldots,e_m) = S_{m+1}(e_1,\ldots,e_m,x(R))$. Second, it explains why it will always be less efficient to use the former in PDP$_m$ solving context when $g \geq 2$.

4 Reducing degree of ideals in Summation approach in even characteristic

If a polynomial parametrization is generated by polynomials as $X_i - P_i(a_1,\ldots,a_l)^p$ in characteristic p, the action of the Frobenius automorphism expresses as a non-standard "hidden" graduation on the polynomial algebra. This can be described by the *weighted degree* of an ideal, that can be determined by computing the Hilbert series of the graded quotient algebra. The analysis of the link between the Hilbert series of the involved ideals allows us to precisely quantify the impact of the different graduation, as the reduction factor reveals to be close to the product of the weight involved in the graduation. Instantiating to a PDP$_m$ context, this leads to a degree reduction of the systems to be solved, akin to the one we describe in Section 2.2.

We emphasize here that it is possible to furhter reduce the degree in a PDP$_m$ setting modelled by a Summation approach, by exploiting the properties of the Decomposition polynomial in a different manner than in Section 2.2. However, a rigorous description of the reduction would mean introducing more notations and subcases, and does not give more insight on the situation either. Lastly, the best reduction we can obtain this way is equivalent to the one we obtain in Section 2.2. For this reason we do not go into more details in this reduction.

4.1 Action of the Frobenius automorphism over polynomial parametrizations

Let $\mathbb{F}$ be a perfect 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 $f^\sigma(X_1^p,\dots,X_m^p) = f(X_1,\dots,X_m)^p$. Assume $m \geq 2$, let $1 \leq l \leq k \leq m$ be integers and let $\mathbf{a} = (a_1,\dots,a_l), \mathbf{X} = (X_1,\dots,X_m)$. For polynomials $P_1,\dots,P_m \in \mathbb{F}[\mathbf{a}]$, 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 define their *ideals of relations*, i.e. the l-th elimination ideals

$$I_e = I \cap \mathbb{F}[\mathbf{X}],\ J_e = J \cap \mathbb{F}[\mathbf{X}].$$

Recall that all such ideals are radical — arguments have been given in Section 3.1. It is straightforward to check that $(z_1,\dots,z_m,a_1,\dots,a_l) \in V(I)$ if and only if $(\sqrt[p]{z_1},\dots,\sqrt[p]{z_k},z_{k+1},\dots,z_m,a_1,\dots,a_l) \in V(J)$. This suggests a natural weight p on $X_{k+1},\dots,X_m$. We turn to eliminations ideals and derive a similar property.

Lemma 23 *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. $g \in J_e \Leftrightarrow g^\sigma(X_1,\dots,X_k,X_{k+1}^p,\dots,X_m^p) \in I_e$.
2. $g \in I_e \Leftrightarrow g(X_1^p,\dots,X_k^p,X_{k+1},\dots,X_m) \in J_e$.

Proof From the definition of I_e and J_e we get

$$g \in J_e \Leftrightarrow g(P_1,\dots,P_m) = 0 \text{ and } g \in I_e \Leftrightarrow g(P_1^p,\dots,P_k^p,P_{k+1},\dots,P_m) = 0.$$

Then we observe that

1. $g \in J_e \Leftrightarrow g(P_1,\dots,P_m)^p = 0 \Leftrightarrow g^\sigma(P_1^p,\dots,P_m^p) = 0 \Leftrightarrow g^\sigma(X_1,\dots,X_k,X_{k+1}^p,\dots,X_m^p) \in I_e$.
2. $g \in I_e \Leftrightarrow g(X_1^p,\dots,X_k^p,X_{k+1},\dots,X_m) \in J_e$. □

For any ideal I, let $I^p = \langle f^p \ :\ f \in I\rangle$. Write $I_e = \langle g_1,\dots,g_r\rangle$ and $J_e = \langle f_1,\dots,f_s\rangle$, and define

$$\begin{aligned} I' &= \left\langle\, g_i^\sigma(X_1,\dots,X_k,X_{k+1}^p,\dots,X_m^p), 1 \leq i \leq r \,\right\rangle, \\ J' &= \left\langle\, f_i(X_1^p,\dots,X_k^p,X_{k+1},\dots,X_m)\ , 1 \leq i \leq s \,\right\rangle, \end{aligned}$$

then Lemma 23 states that $I' \subset I_e$ and $J' \subset J_e$. The next Proposition makes this link precise.

Proposition 24 *With the previous notations, $I_e^p \subset I' \subset I_e$ and $J_e^p \subset J' \subset J_e$.*

Proof Let $f \in I_e$. Lemma 23 gives that $f(X_1^p,\dots,X_k^p,X_{k+1},\dots,X_m) \in J_e$. Hence there exists $q_i \in \mathbb{F}[\mathbf{X}]$ such that

$$f(X_1^p,\dots,X_k^p,X_{k+1},\dots,X_m) = \sum_{i=1}^r q_i(X_1,\dots,X_m) g_i(X_1,\dots,X_m).$$

Evaluating at $X_1,\dots,X_k,X_{k+1}^p,\dots X_m^p$ and taking p-th power give

$$\begin{aligned} f(X_1^p,\dots,X_m^p)^p &= \sum_{i=1}^r q_i(X_1,\dots,X_k,X_{k+1}^p,\dots,X_m^p)^p g_i(X_1,\dots,X_k,X_{k+1}^p,\dots,X_m^p)^p \\ &= \sum_{i=1}^r q_i^\sigma(X_1^p,\dots,X_k^p,X_{k+1}^{p^2},\dots,X_m^{p^2}) g_i^\sigma(X_1^p,\dots,X_k^p,X_{k+1}^{p^2},\dots,X_m^{p^2}) \end{aligned}$$

which means that

$$f(X_1,\dots,X_m)^p = \sum_{i=1}^r q_i^\sigma(X_1,\dots,X_k,X_{k+1}^p,\dots,X_m^p)^p g_i^\sigma(X_1,\dots,X_k,X_{k+1}^p,\dots,X_m^p)$$

so that $f^p \in I'$. The other inclusion follows similar arguments. □

Corollary 25 *With the previous notations, I_e is the radical of I' and J_e is the radical of J'.*

Proof Proposition 24 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. □

Assuming the base field is algebraically closed, we know from Corollary 25 that $\sqrt{I'} = I_e$, so that $I(V(I')) = I_e$ and $V(I_e) = V(I')$. Then a tuple $(z_1, \dots, z_m)$ is in $V(I_e)$ if and only if for all $1 \le i \le r$,

$$g_i^\sigma(\sqrt[p]{z_1}^{\,p}, \dots, \sqrt[p]{z_k}^{\,p}, z_{k+1}^p, \dots, z_m^p) = 0 = g_i(\sqrt[p]{z_1}, \dots, \sqrt[p]{z_k}, z_{k+1}, \dots, z_m)^p,$$

equivalently, $(\sqrt[p]{z_1}, \dots, \sqrt[p]{z_k}, z_{k+1}, \dots, z_m) \in V(J_e)$. In other words it is equivalent to work with $V(I_e)$ or $V(J_e)$. Since the two associated ideals are radical, in practice we can use either I_e or J_e for computations. To proceed to degree analysis, we now need to introduce the *weighted degree* of an ideal. This quantity can be computed using the Hilbert Series of the quotient algebra, which is the generating power series for the number of monomials of degree d, $d \ge 0$, in the algebra. It is defined for homogeneous ideals, but it can be extended to any ideals by considering its homogenization. Indeed, the homogenization of the elements in a Gröbner basis for a a degree order is a Gröbner basis for the homogenization ideal [8, Thm. 4, p.388]. When the ideal is radical, geometrically it amounts to working in the projective closure of the variety generated by the ideal. More details can be found in [31][46].

Definition 26 ([46]) *Let I be a polynomial ideal of dimension d in $K[X_1, \dots, X_n]$ equipped with weight $w = (w_1, \dots, w_n)$. Let $HS_I(T)$ be the Hilbert Series of $K[X_1, \dots, X_n]/I$. Let $Q(T) = (1-T)^d HS_I(T)$. The weighted degree of I is* $\deg_w I = Q(1)$. *The weighted degree of a variety is the weighted degree of its associated ideal.*

If the weights give the standard graduation (*i.e.* $w = (1, \dots, 1)$), then $Q(T)$ is a polynomial and the weighted degree is the classical degree of an ideal, denoted by $\deg I$. We now use the ideals I' resp. J' to estimate the weighted degree of I_e, resp. I_e.

Proposition 27 *For $1 \le i \le k$ let $w_i = 1$ and $w'_i = p$, and for $k < i \le m$, let $w_i = p$ and $w'_i = 1$. For the weight systems $w = (w_1, \dots, w_m)$ and $w' = (w'_1, \dots, w'_m)$, 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 $A = (\mathbb{F}[X_1, \dots, X_m], (1, \dots, 1))$ be the polynomial algebra with standard graduation, and consider the w-graded algebra $A_w = (\mathbb{F}[Y_1, \dots, Y_m], (w_1, \dots, w_m))$. We see the ideal $J_e = \langle g_1, \dots, g_r \rangle$ in this algebra, and we let also $J_e^\sigma = \langle g_1^\sigma, \dots, g_r^\sigma \rangle$. Using the injective homomorphism of graded algebras $\varphi : A_w \longrightarrow A$ defined by $\varphi(Y_i) = X_i^{w_i}$, Lemma 23 restates as $\varphi(J_e^\sigma) = I'$. From [46, prop. 3.10, p.96], we have $\deg_w J_e^\sigma = \frac{\deg \varphi(J_e^\sigma)}{p^{m-k}}$, so the last thing to do is to verify that $\deg_w J_e = \deg_w J_e^\sigma$.

Wlog. we can assume that the generators 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 $\{g_i^\sigma : 1 \le i \le r\}$ is a Gröbner Basis for J_e^σ, hence $\deg_w J_e = \deg_w J_e^\sigma$. The other equality is obtained by adapting the whole argument. □

4.2 Application to 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 (perfect) field $\mathbb{F}$ of characteristic 2. Let $R \in \mathrm{Jac}(\mathscr{H})$ of weight g and F be the associated m-Decomposition polynomial, for some $m \ge g+1$:

$$F(X) = X^m + \sum_{i=0}^{m-1} N_i(\mathbf{a}) X^i.$$

If L_1 is the length of h_1, Proposition 11 tells us that F has $k = g - L_1$ relevant squared coefficients. Assume that $L_1 < g$, and for simplicity, renumber the coefficients of F and the e_i such that the squares are $N_1(\mathbf{a}) = \tilde{N}_1(\mathbf{a})^2, \dots, N_k(\mathbf{a}) = \tilde{N}_k(\mathbf{a})^2$. We will always assume this is the case through the rest of this Section. We focus the ideal I associated to $\mathscr{V}_{m,R}$ and the ideal J defined by:

$$\begin{aligned} I &= \langle\, e_i + N_i(\mathbf{a}) \;:\; 1 \le i \le m \,\rangle,\ I_e = I \cap \mathbb{F}[\mathbf{e}], \\ J &= \left\langle\, e_i + \tilde{N}_i(\mathbf{a}) \;:\; 1 \le i \le k,\ e_i + N_i(\mathbf{a}),\ k+1 \le i \le m \,\right\rangle,\ J_e = J \cap \mathbb{F}[\mathbf{e}]. \end{aligned} \tag{15}$$

A first benefit of using J instead of I is that some quadratic equations have been replaced by linear equations. Hence it should be faster to compute a basis of J_e than to compute a basis of I_e. A second benefit is that the degree of the ideal obtained after Weil Descent over J_e is lower than the one obtained with I_e. To show this, we need to highlight the differences between the degrees of J_e and I_e. However, Proposition 27 gives only a link between $\deg_w J_e$ and $\deg I'$. This prompts the introduction of the next constant.

Definition 28 *The degree ratio between I' and I_e is noted $C_1 = \dfrac{\deg I'}{\deg I_e}$.*

Since I' is the image of J_e^σ by an injective homomorphism of algebras, then $\dim I' = \dim J_e = \dim I_e$. Since $I' \subset I_e$, we infer that $\deg I' \geq \deg I_e$. With Proposition 24, we obtain $1 \leq C_1 \leq \frac{\deg I_e^2}{\deg I_e}$. We can now estimate the reduction factor obtained by working with J_e.

Proposition 29 *With the previous notations, we have:*

$$\deg_w V(J_e) = C_1 \cdot \frac{\deg V(I_e)}{2^{m-g+L_1}}.$$

Proof Since I_e and J_e are radical, we have $\deg_w V(J_e) = \deg_w J_e$ and $\deg V(I_e) = \deg I_e$ for any weight system. From Proposition 11, the Decomposition Polynomial has $k = g - L_1$ squares among its coefficients in X. Let $w = (w_1, \ldots, w_m)$ with $w_1 = \ldots = w_k = 1$ and $w_{k+1} = \ldots = w_m = 2$ and consider J_e in the graded algebra $(\mathbb{F}[\mathbf{e}], (w_1, \ldots, w_m))$. Proposition 27 states that

$$\deg_w J_e = \frac{\deg I'}{2^{m-g+L_1}} = \frac{C_1 \cdot \deg I_e}{2^{m-g+L_1}}.$$

□

Experimentally from genus 2 to 4, in this setting, C_1 is a power of 2 with exponent much less than $m - g + L_1$, so the weighted degree of $V(J_e)$ is indeed divided by a number close to the product of the weights. Further in the presentation we propose a conjecture to the value of its exponent.

In the context of a Decomposition attack, the field is some $\mathbb{F}_{2^{dn}}$, and $m = ng$. The Weil descent involves cutting the Weil restriction of J_e by hyperplanes. This is where the second benefit of working with J_e appears. The next result takes the graduation into account when we cut $\mathscr{W}_n(J_e)$ by hyperplanes, and gives an estimate of the weighted degree of the zero-dimensional ideal produced by the Weil Descent.

Proposition 30 *Keeping previous notations, and assume that the field is now $\mathbb{F}_{2^{dn}}$. Let $\mathbb{I}$ be the ideal obtained by a Weil Descent on J_e. Under Genericity assumption 22, we have:*

$$\deg_w \mathbb{I} = C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{(n-1)g+L_1}}.$$

Proof With $m = ng$, Proposition 29 gives:

$$\deg_w \mathscr{W}_n(J_e) = C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{n((n-1)g+L_1)}}.$$

Let $1, t, \ldots, t^{n-1}$ be a a power $\mathbb{F}_{2^d}$-basis of $\mathbb{F}_{2^{dn}}$ and write $e_i = \sum_{j=0}^{n-1} e_{i,j} t^j$. As the graduation involves the characteristic, it extends naturally to the Weil Restriction. To see this, we observe that $e_i^2 = \sum_{j=0}^{n-1} e_{i,j}^2 t^{2j}$ so that only squares of the $e_{i,j}$ will appear in this expression: in other words the graduation is applied on the new variables coming from the Weil restriction. Let $\mathbf{w} = (w_{1,0}, w_{1,1}, \ldots, w_{ng,n-1})$ with $w_{1,0} = w_{1,1} = \ldots = w_{k,n-1} = 1$ and $w_{k+1,0} = w_{k+1,1} = \ldots = w_{ng,n-1} = 2$, and consider $I(\mathscr{W}_n(J_e))$ as an ideal in the $\mathbf{w}$-graded algebra $\mathbb{F}_{2^d}[e_{1,0}, \ldots, e_{ng,n-1}]$. Geometrically, the Weil Descent amounts to cut $\mathscr{W}_n(J_e)$ by the intersection of the (weighted) hyperplanes

$$H = \bigcap_{\substack{1 \leq i \leq ng \\ 1 \leq j \leq n-1}} V(e_{i,j}),$$

with $\deg_{\mathbf{w}} V(e_{i,j}) = 2$ for $g - L_1 + 1 \leq i \leq ng, 1 \leq j \leq n-1$ and thus $\deg_{\mathbf{w}} H = 2^{((n-1)g+L_1)(n-1)}$. Let now $\mathbb{I}$ be the ideal associated to $\mathscr{W}_n(J_e) \cap H$. With the Genericity assumption 22, its dimension is 0, so that the intersection has weighted degree $\deg_{\mathbf{w}} \mathbb{I} = \deg_{\mathbf{w}} \mathscr{W}_n(J_e) \cdot \deg_{\mathbf{w}} H$. The claim follows:

$$\begin{aligned}\deg_{\mathbf{w}} \mathbb{I} &= C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{n((n-1)g+L_1)}} \cdot 2^{((n-1)g+L_1)(n-1)} \\ &= C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{(n-1)g+L_1}}.\end{aligned}$$

□

If I is a zero-dimensional ideal in $K[Y_1, \ldots, Y_n]$ graded by w, then $\deg_w I = \dim_K K[Y_1, \ldots, Y_n]/I$, where the dimension is meant as the dimension as a K-linear space. To see this, consider the injective homomorphism of graded algebras $\varphi : (K[Y_1, \ldots, Y_n], w) \longrightarrow (K[X_1, \ldots, X_n], (1, \ldots, 1))$ defined by $\varphi(Y_i) = X_i^{w_i}$. Then $\dim \varphi(I) = 0$ and by [46, prop. 3.10] we have

$$\deg_w I = \frac{\deg \varphi(I)}{\prod_{i=1}^n w_i} = \frac{\dim_K K[X_1, \ldots, X_n]/\varphi(I)}{\prod_{i=1}^n w_i}.$$

Now the image by φ of a Gröbner basis of I for the w-grevlex order is a Gröbner basis for the grevlex order for $\varphi(I)$. Informally, this means that "going through φ" multiplies the volume under the stair of I by $\prod_{i=1}^n w_i$. From this we obtain $\deg_w I = \dim_K K[Y_1, \ldots, Y_n]/I$, and this means that for zero-dimensional (radical) ideal, the weighted degree also counts the number of elements in the associated variety. Hence for a 0-dimensional ideal I we use the notation $\deg I$ to count its number of solutions, independently of the graduation.

From the point of view of FGLM's algorithm, this says that the complexity of the change-order step can be expressed in term of the weighted degree of I. The next result formulates this observation in the context of Decomposition attacks, and sums up this Section.

Corollary 31 *Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be a genus g hyperelliptic curve defined over $\mathbb{F}_{2^{dn}}$, and fix $R \in \mathrm{Jac}(\mathscr{H})$ of weight g. Let L_1 be the length of h_1. The PDP$_{ng}$ instance related to R can be solved by computing a lexicographical Gröbner Basis for a zero-dimensional ideal $\mathbb{I}$ of degree $C_1^n \cdot \frac{\deg \mathscr{W}_n(I_e)}{2^{(n-1)g+L_1}}$.*

Remark 32 *If Conjecture 21 is true, then $\deg V(I_e) = \deg V(\mathbb{S}_{m,R}) = 2^{m-g}$ and $\deg \mathscr{W}_n(I_e) = 2^{n(n-1)g}$. In this case Proposition 29 rewrites as $\deg_w V(J_e) = C_1 \cdot 2^{-L_1}$, and Proposition 30 then tells that $\deg \mathbb{I} = C_1^n \cdot 2^{(n-1)^2 g - L_1}$.*

4.3 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 coefficients of the Decomposition polynomial, the closer $\deg I'$ is to $\deg I_e$ and C_1 is to 1. No square appears 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 30 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_1	$\deg \mathbb{I}$	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 4 First step of degree reduction for genus 2 binary hyperelliptic curves.

Higher genus The value of the constant C_1 seems to be linked with the length of h_1, or more accurately, to the rank of the 2-torsion in $\mathrm{Jac}(\mathscr{H})$. 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 curve 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 $L_1 = 1$. If h_1 is monic of degree 2 and irreducible, we observe $C_1 = 4$, and $L_1 = 2$.
- 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 $L_1 = 2$. When h_1 is monic and irreducible of degree 3, then $C_1 = 8$ with $L_1 = 3$. 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 33 *Let $\mathscr{H} : y^2 + h_1(x)y = h_0(x)$ be a hyperelliptic curve of genus g defined over $\mathbb{F}_{2^{dn}}$. Assume h_1 is not irreducible of degree g and of length L_1. Then the degree ratio C_1 defined in Proposition 29 is a power of 2 that only depends on the polynomial h_1. More precisely, we have:*

$$C_1 = 2^{L_1}.$$

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

$$\deg \mathbb{I} = 2^{(n-1)((n-1)g+L_1)}.$$

Remark 34 *If this Conjecture is true, then we find the following bounds for the first reduction step:*

$$2^{(n-1)^2 g} \leq \deg \mathbb{I} \leq 2^{(n-1)(ng-1)}.$$

While there is no known classification for binary hyperelliptic curves in general when $g \geq 3$, the (squarefree part of the) polynomial h_1 determines the 2-rank of $\mathrm{Jac}(\mathscr{H})$. It may be possible to classify all the possible degree reductions based on the squarefree decomposition of h_1.

5 Comparisons of methods and practical impact

We proposed an new method to solve PDP_m instances using Summation polynomials instead of Nagao's approach. The natural question is now to compare them to estimate which one is the most efficient for a given task. We start by a quick comparison in odd characteristic, and then turn to even characteristic. The timings we obtain shows that overall Nagao with our degree reduction algorithm is a better approach. We also see that the Summation approach with degree reduction is way faster than the standard Nagao's approach. This illustrates the power of the degree reduction in polynomial system solving. The Section and the article ends with a description on how we handle the realistic computation for a genus 2 Type II class group with 2^{184} elements.

5.1 Nagao vs Summation in Odd characteristic

Experiments were done on $\mathbb{F}_{q^n}$ with $\log q = 16$, $n = 2,3$, and imaginary genus 2 curves given by general Weierstrass equation $y^2 = h(x)$. This means we look for $2n$-decompositions of a given R of weight g. 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 Summation set for a given R of weight 2, that is to say, eliminating variables from a parametrization of the corresponding $\mathscr{V}_{m,R}$. Implementation were done with Magma 2.19 [2], so that DRL Gröbner basis and elimination basis are computed with $F4$, on the same computer as the previous experiments of this article.

Table 5 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 compute Summation sets 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 classic strategy.

Even if we assume that a symbolic 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 many quadratic equations as variables, whereas Summation's approach needs less variables but gives equations of greater degree.

5.2 Nagao vs Summation for binary genus **2** curves

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 Table 1 and Section 4. 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.

The timings highlighted by exclamations marks are abnormally long. Since, once computed, the lexicographical bases are not in Shape position, this suggests a problem in Magma 2.19 implementation[5] 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

[5] We did not try more recent version.

Table 6 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.		

$n(n-1)g = 12$ 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.

We again used computational strategies 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 first step, then the last is eliminated. This strategy leads to important speed-ups in our experiments for the elimination, but this step was still the bottleneck in Reduced Summation approach. Table 6 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.

5.3 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 such that $t^3 + \alpha t + \beta = 0$ with $\alpha = 7\text{BCEB1AC}$ and $\beta = 50\text{F6CCC4}$. These values are obtained by considering α, β as polynomial in ω, evaluate them at 2 and converting the integer we obtained in hexadecimal. Let also $\mathbb{F}_{2^{93}} = \mathbb{F}_{2^{31\cdot 3}} \simeq \mathbb{F}_{2^{31}}[t]$. We solve PDP_6 instances using our refined Nagao modelling.

Type II curve: Let $\mathscr{H} : y^2 + xy = x^5 + f_3x^3 + x^2 + f_0$, with

$$\begin{aligned} f_3 &= \text{A814B6C09256168AC93ABA1}, \\ f_0 &= \text{16400CBCC65A5EE5F67165AC}, \\ \#\mathscr{H}(\mathbb{F}_{2^{93}}) &= 9903520314283080096056319534 \geq 2^{93} \end{aligned}$$

These parameters were obtained with several tries with Magma, until the cardinality of the class group was large enough. Using Magma 2.19 implementation Vercauteren's version [45] of Kedlaya's algorithm for counting points, it takes approximately 24 seconds to verify that the class group has order

$$\#\text{Jac}(\mathscr{H}) = 2 \times 3 \times 1634661910256954370788166730322099364314237310743193865 3,$$

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.

The systems coming from the univariate polynomial among the defining equations 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

$$N_1(a_4) \in \mathbb{F}_{2^{31}} \Leftrightarrow a_{4,i} = u'_{1,i} \text{ for } i = 1, 2.$$

Hence those values are directly known 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 $R \in \mathrm{Jac}(\mathscr{H})$ to decompose is computed using a pseudo-random walk as proposed by Gaudry [24]. 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 2.2. The resulting system has resp. 4 (resp. 6) linear (resp. quadratic) equations in 10 variables, and is solved following the classic 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 [15] algorithm. We can check that $\sqrt{\mathscr{S}_2}$ has 64 solutions.
- With Sparse-FGLM algorithm [18], 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 [43]. This is done in $2.22 \cdot 10^{-3}$sec.

Overall, solving one PDP_6 instance over $\mathscr{H}$ take $3.2 \cdot 10^{-3}$sec., and finding the roots of the degree 64 univariate polynomial becomes the bottleneck of the computation. This is because we did not try to use any optimizations to speed-up this particular step. If such optimizations were to be used, it is believable that the harvesting time could be slightly reduced. Memory-wise the whole process is really efficient as approximately 1.1 MB 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 [3][5] 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. A sparse linear algebra algorithm — usually a block Wiedemann — is expected to run in around 2^{63} operations.

This can be compared to the size of the matrices obtained after the filtering step in the record factorizations of a RSA-768 modulus [32] or a 1061 bits number [6], and more interestingly, to the recent computation of a discrete logarithm in a finite field of size 768 bits reported in [33]. There the authors harvested around 10 billions relations in 4000 core years. After an efficient and dedicated filtering, the linear algebra was done on a matrix with roughly 25 millions rows and an average of 134 non-zero elements by row. Computation of the kernel was done modulo a 767 bits integer in around 920 core years. By comparison, the harvesting could be run much longer in our context: for example for 6 months, which is less than the harvesting duration of [33], it can be hoped that around 2^{35} relations could be obtained. The linear algebra in our setting would be modulo a 184 bits integer. Assuming a dedicated filtering could be designed, we may hope that the reduced matrix is small enough (for example, 50 millions row) so that the computations can be done in comparable time with the 768 bits finite field DLP.

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 which 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 with degree having a small factor should in general be avoided for curves with short h_1.

Acknowledgments We want to thank the anonymous reviewers for their useful suggestions and insights towards the improvement of this article, as well as pointing out valuable references.

References

1. M. Bardet, J-C. Faugère, B. Salvy On the complexity of the $F5$ Gröbner basis algorithm. *J. Symbolic Comput.*, p. 1–24, September 2014.
2. W. Bosma, J. Cannon, and C. Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. Computational algebra and number theory (London, 1993).
3. C. Bouvier. The filtering step of discrete logarithm and integer factorization algorithms. Preprint, 22 pages, http://hal.inria.fr/hal-00734654, 2013.
4. B. Byramjee and S. Duquesne. Classification of genus 2 curves over $\mathbb{F}_{2^n}$ and optimization of their arithmetic. Cryptology ePrint Archive, Report 2004/107, 2004
5. The CADO-NFS Development Team. CADO-NFS, An Implementation of the Number Field Sieve Algorithm, http://cado-nfs.gforge.inria.fr/, Release 2.2.0, 2015.
6. G. Childers. Factorization of a 1061-bit number by the Special Number Field Sieve. Cryptology ePrint Archive, Report 2012/444, 2012.
7. Ping Ngai Chung and Craig Costello and Benjamin Smith. Fast, uniform, and compact scalar multiplication for elliptic curves and genus 2 Jacobians with applications to signature schemes. In *CoRR*, abs/1510.03174, 2015
8. D. A. Cox, J. Little, D. O'Shea. Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra, 3/e (Undergraduate Texts in Mathematics), 2007 Springer-Verlag New York, Inc.
9. C. Diem. An index calculus algorithm for plane curves of small degree. In *Algorithmic number theory*, volume 4076 of *Lecture Notes in Comput. Sci.*, pages 543–557. Springer, 2006
10. C. Diem. On the discrete logarithm problem in elliptic curves. In *Compositio Mathematica*, volume 147, pages 75–104, 2011
11. C. Diem. The GHS Attack in odd Characteristic. In *J. Ramanujan Math. Soc. 18*, No.1, 1-32 (2003)
12. J.-C. Faugère. FGb: A Library for Computing Gröbner Bases. In Komei Fukuda, Joris Hoeven, Michael Joswig, and Nobuki Takayama, editors, *Mathematical Software ICMS 2010*, volume 6327 of Lecture Notes in Computer Science, pages 84-87, Berlin, Heidelberg, September 2010. Springer Berlin / Heidelberg.
13. J-C. Faugère, P. M. Gianni, D. Lazard, T. Mora Efficient Computation of Zero-Dimensional Gröbner Bases by Change of Ordering. In *J. Symb. Comput.*, volume 16, nb. 4, pages 329–344, 1993.
14. J.-C. Faugère. A new efficient algorithm for computing Gröbner bases (F4). In *Journal of Pure and Applied Algebra*, vol. 139 (1): 61–88, June 1999.
15. J.C. Faugère. A New Efficient Algorithm for Computing Gröbner Bases Without Reduction to Zero (F5). Proceedings of *the 2002 International Symposium on Symbolic and Algebraic Computation, ISSAC '02*, 2002.
16. J-C. Faugère, P. Gaudry, L. Huot, G. Renault. Using Symmetries in the Index Calculus for Elliptic Curves Discrete Logarithm. In *J. Cryptology*, volume 27, nb. 4, pages 595–635, 2014.
17. J-C. Faugère, L. Huot, A. Joux, G. Renault, 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, Proceedings.* pages 40–57, 2014.
18. J.C. Faugère, C. Mou. Fast algorithm for change of ordering of zero-dimensional Gröbner bases with sparse multiplication matrices. In *Symbolic and Algebraic Computation, International Symposium, ISSAC, 2011 (co-located with FCRC, San Jose, CA, USA, June 7-11.* Proceedings, pages 115–122, 2011.
19. G. Frey, M. Müller, H-G. Rück, The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems. In *IEEE Transactions on Information Theory, volume 45, pages 1717–1719*, 1999.
20. G. Frey, H-G. Rück, A remark concerning m-divisibility and the discrete logarithm problem in the divisor class group of curves. In *Math. Comp.* **62**(1994), 865-874r.
21. S. D. Galbraith, P. Gaudry. Recent progress on the elliptic curve discrete logarithm problem In *Des. Codes Cryptography*, 78-1:51–72, 2016
22. S. D. Galbraith,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*. Proceedings, pages 409–427, 2014
23. P. Gaudry. Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem. In *J. Symb. Comput.*, volume 44, nb. 12, pages 1690–1702, 2009.
24. P. Gaudry. An algorithm for solving the discrete log problem on hyperelliptic curves. In *Advances in cryptology—EUROCRYPT 2000*, volume 1807 of *Lecture Notes in Comput. Sci.*, pages 19–34. Springer, 2000.
25. Pierrick Gaudry. Fast genus 2 arithmetic based on Theta functions. In *J. Mathematical Cryptology*, volume 1-3, pages 243–265, 2007
26. Pierrick Gaudry and David Lubicz. The arithmetic of characteristic 2 Kummer surfaces and of elliptic Kummer lines. In *Finite Fields and Their Applications*, volume 15-2, pages 246–260, 2009.
27. 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.
28. 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.
29. 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*, volume 7237 of *Lecture Notes in Comput. Sci.*, pages 9–26. Springer, 2012.

30. A. Joux, V. Vitse. Elliptic Curve Discrete Logarithm Problem over Small Degree Extension Fields - Application to the Static Diffie-Hellman Problem on $E(\mathbb{F}_{q^5})$. In *J. Cryptology*, volume 26, pages 119–143, 2013.
31. G. Kemper Hilbert Series and Dimension. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.
32. 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, 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*. Proceedings, pages 333–350, 2010.
33. T. Kleinjung, C. Diem, A. K. Lenstra, C. Priplata, C. Stahlke Computation of a 768 bits prime field discrete logarithm In *Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30- May 4, Proceedings.*
34. Arjen K. Lenstra and Henrik W. Lenstra Jr. and Mark S. Manasse and John M. Pollard. The Number Field Sieve, In *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing,* May 13-17, 1990, Baltimore,pages 564–572, 1990.
35. R. Lidl, H. Niederreiter. *Finite Fields*, Encyclopedia of Mathematics and its Applications 20 (Second ed., 1997), Cambridge University Press, ISBN 0-521-39231-4, Zbl 0866.11069
36. D. Lubicz, D. Robert Arithmetic of Abelian and Kummer varieties Finite fields and their applications
37. Gary L. Mullen, D. Panario. *Handbook of Finite Fields*, CRC Press, ISBN 978-1-4398-7378-6
38. K-I. 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*. Proceedings, pages 285–300, 2010.
39. J. Renes, P. Schwabe, B. Smith, L. Batina. μ-Kummer: efficient hyperelliptic signatures and key exchange on microcontrollers. CoRR, abs/1604.06059, 2016
40. I. Semaev. Summation polynomials and the discrete logarithm problem on elliptic curves. IACR Cryptology ePrint Archive, 2004
41. V. Shoup. Lower Bounds for Discrete Logarithms and Related Problems. In *Advances in Cryptology - EUROCRYPT '97, International Conference on the Theory and Application of Cryptographic Techniques, Konstanz, Germany, May 11-15*. Proceedings, pages 256–266, 1997.
42. Y. Choie, D. Yun. Isomorphism classes of hyperelliptic curves of genus 2 over $\mathbb{F}_{2^n}$. In *ACISP 2002. LNCS, vol. 2384*. Proceedings, pages 190–202, 2002.
43. V. Shoup. *NTL: A Library for doing Number Theory.* Courant Institute, New York University, 2005.
44. C. Tran. Formules d'addition sur les jacobiennes de courbes hyperelliptiques : application à la cryptographie Ph. D. Thesis, 2014
45. F. Vercauteren. Computing zeta functions of hyperelliptic curves over finite fields of characteristic 2. In *Advances in Cryptology—CRYPTO 2002*, volume 2442 of LNCS, pages 369–384. Springer, Berlin, 2002.
46. T. Verron. Régularisation du calcul de bases de Gröbner pour des systèmes avec poids et déterminantiels, et applications en imagerie médicale. Ph. D. Thesis, 2016
47. D. Wiedemann. Solving sparse linear equations over finite fields. In *IEEE Trans. Information Theory*, vol. 32, 1-54–62, 1986.