

Encoding Points on Hyperelliptic Curves over Finite Fields in Deterministic Polynomial Time

Jean-Gabriel Kammerer^{1,2}, Reynald Lercier^{1,2}, and Guénaél Renault³

¹ DGA MI, La Roche Marguerite, F-35174 Bruz Cedex, France

² Institut de recherche mathématique de Rennes, Université de Rennes 1
Campus de Beaulieu, F-35042 Rennes Cedex, France

`jean-gabriel.kammerer@m4x.org`,

`reynald.lercier@m4x.org`

³ LIP6, Université Pierre et Marie Curie, INRIA/LIP6 SALSA Project-Team
Boite courrier 169, 4 place Jussieu, F-75252 Paris Cedex 05, France

`guenael.renault@lip6.fr`

Abstract. We provide new hash functions into (hyper)elliptic curves over finite fields. These functions aim at instantiating in a secure manner cryptographic protocols where we need to map strings into points on algebraic curves, typically user identities into public keys in pairing-based IBE schemes.

Contrasting with recent Icart’s encoding, we start from “easy to solve by radicals” polynomials in order to obtain models of curves which in turn can be deterministically “algebraically parameterized”. As a result of this strategy, we obtain a low degree encoding map for Hessian elliptic curves, and for the first time, hashing functions for genus 2 curves. More generally, we present for any genus (more narrowed) families of hyperelliptic curves with this property.

The image of these encodings is large enough to be “weak” encodings in the sense of Brier et al. As such they can be easily turned into admissible cryptographic hash functions.

Keywords: deterministic encoding, elliptic curves, Galois theory, hyperelliptic curves.

1 Introduction

Many asymmetric cryptographic mechanisms are based on the difficulty of the discrete logarithm problem in finite groups. Among these groups, algebraic curves on finite fields are of high interest because of the small size of keys needed to achieve good security. Nonetheless it is less easy to encode a message into an element of the group.

Let \mathbb{F}_q be a finite field of odd characteristic p , and $H/\mathbb{F}_q : y^2 = f(x)$ where $\deg f = d$ be an elliptic (if $d = 3$ or 4) or hyperelliptic (if $d \geq 5$) curve, we consider the problem of computing points on H in deterministic polynomial time. In cryptographic applications, computing a point on a (hyper)elliptic curve is a prerequisite for encoding a message into its Jacobian group. In this regard,

pairing-based cryptosystems do not make exception. Boneh-Franklin Identity-Based Encryption scheme [3] requires for instance to associate to any user identity a point on an elliptic curve.

In the case of elliptic curves, we may remark that it is enough to compute one rational point G , since we can have other points tG from integers t (at least if G is of large enough order). To compute such a G , one might test random elements $x \in \mathbb{F}_q$ until $f(x)$ is a square. But without assuming GRH, we have no guarantee of finding a suitable x after a small enough number of attempts, and no deterministic algorithm is known for computing square roots when $p \equiv 1 \pmod 4$. Moreover, encoding t into tG voids the security of many cryptographic protocols [13].

Maybe a more serious attempt in this direction for odd degrees d is due to Atkin and Morain [1]. They remark that if x_0 is any element of \mathbb{F}_q and $\lambda = f(x_0)$, then the point $(\lambda x_0, \lambda^{(d+1)/2})$ is on the curve $Y^2 = \lambda^d f(X/\lambda)$. But again, the latter can be either isomorphic to the curve or its quadratic twist, following that λ is a quadratic residue or not, and we have no way to control this in deterministic time.

In 2006, Shallue and Woestjine [16] proposed the first practical deterministic algorithm to encode points into an elliptic curve, quickly generalized by Ulas [17] to the family of hyperelliptic curves defined by $y^2 = x^n + ax + b$ or $y^2 = x^n + ax^2 + bx$. Icart [13] proposed in 2009 another deterministic encoding for elliptic curves, of complexity $\mathcal{O}(\log^{2+o(1)} q)$, provided that the cubic root function, inverse of $x \mapsto x^3$ on \mathbb{F}_q^* , is a group automorphism. This is equivalent to $q \equiv 2 \pmod 3$. This encoding uses Cardano-Tartaglia’s formulae to parameterize the points $(x : y : 1)$ on any elliptic curve $E : x^3 + ax + b = y^2$.

1.1 Contribution

In this paper, we propose a strategy for finding other families with such properties (Section 2). As an example, we first show how the strategy works for genus 1 curves and come to a new encoding map for Hessian elliptic curves in finite fields of odd characteristic with $q \equiv 2 \pmod 3$ (Section 3.1). Hessian curves are defined by $E_d : x^3 + y^3 + 1 = 3dxy$ with $d \neq 1$, but admit an equivalent model $C_{0,a} : y^3 + xy + ay = x^3$ with $a \neq 0, 1/27$. Our encoding function is 2 : 1: each point of its image admits exactly 2 preimages compared to the 1 to potentially 4 in Icart’s construction.

We then study more carefully genus 2 curves, still over finite fields \mathbb{F}_q of odd characteristic with $q \equiv 2 \pmod 3$, and exhibit several large families (Section 3.2), namely the curves $H_{1,a,b} : y^2 = (x^3 + 3ax + 2)^2 + 8bx^3$ (which we call type A) and $H_{2,\lambda,\mu,a,v,w} : y^2/\lambda = (x^3 + 3\mu x + 2a)^2 + 4b$ (type B). Both of these families have dimension 2 over the moduli space.

Finally for all genus $g \geq 2$, we propose families of hyperelliptic curves which admit an efficient deterministic encoding function (Section 4), again provided some conditions on q (typically q odd, $q \equiv 2 \pmod 3$ and q coprime to $2g + 1$).

Remark 1. In the paper, we use indifferently the words “parameterization” or “encoding”, even if, strictly speaking, we do not have fully parameterized curves.

We are aware that these maps are at least improperly parameterizations since there might correspond more than one parameter to one point. There are numerous points which lie outside the image of our maps too.

Remark 2. Our main goal is to find the biggest possible families for which there exists an efficient deterministic encoding. It might be possible to reduce the number of finite field operations in the description of our algorithms. However, they all use an exponentiation step like every other known encoding which dominates by far the computation time, both from a practical and a complexity viewpoint.

Remark 3. Each of our encodings is a *weak encoding* in the sense of [6]. Combined with a cryptographic hash function, we can thus construct hash functions into the set of rational points of these curves that are indiffereniable from a random oracle.

1.2 Related Work

Other papers about encoding into curves have been independently prepublished soon after the publication of a preversion of this article on the arXiv.org e-Print archive (ref 1005.1454).

Especially, Fouque and Tibouchi proposed an injective encoding over odd hyperelliptic curves, that is curves of the form $y^2 = f(x)$ with $f(-x) = -f(x)$ [12]. For any genus g , this encoding targets a dimension $g - 1$ subspace of hyperelliptic curves, so of dimension 1 for genus 2 curves. Farashahi proposed an other 2 : 1 encoding into Hessian curves too [11].

Because of square or cubic rooting steps, all these encodings have the same asymptotic complexity as ours, namely $\mathcal{O}(\log^{2+o(1)} q)$.

2 A Strategy

Given a genus g , we describe a strategy for finding curves of genus g which admit a deterministic encoding for a large subset of their points.

It's worth noting first that only genus 0 curves are *rationaly* parameterizable. That is, any curve which admits a rational parameterization shall be a conic, see [15, Theorem 4.11]. Encoding maps into higher genus curves shall thus be *algebraic*. We are then reduced to the parameterization of roots of polynomials. Hence, the main idea of our general strategy is to start from polynomials with roots which are easily parameterizable and then deduce curves with deterministic encoding.

2.1 Solvable Polynomials

Classical Galois theory offers a large family of polynomials with easily parameterized roots: polynomials with roots that can be written as radicals, which are polynomials with solvable Galois group. Our strategy is based on these polynomials.

More precisely, let $f_{\underline{a}}(X)$ be a family of parameterized polynomials (where \underline{a} denotes a k -tuple (a_1, a_2, \dots, a_k) of parameters) with solvable Galois group. We are interested in such parametric polynomials but also in the parametric radical expression of their roots $\chi_{\underline{a}}$. For instance $f_A(X) = X^2 + A$ in degree 2, or more interestingly $f_{A,B}(X) = X^3 + AX + B$ in degree 3, are such polynomials with simple radical formulae for their roots. The former verifies $\chi_A = \sqrt{-A}$ and a root of the second one is given by the well-known Cardano-Tartaglia's formulae (see [8]). The application of our general strategy to this family of degree 3 polynomials with the parameterization of its roots is described in Section 3.

Let us note that we might use the classical field machinery to construct new solvable polynomials from smaller ones. Look for instance at De Moivre's polynomials of degree d : we start from the degree 2 field extension $\theta^2 + B\theta - A^d$, followed by the degree d Kummer extension $\gamma^d - \theta = 0$. Then the element $X = \gamma - A/\gamma$ is defined in a degree d subfield of the degree $2d$ extension. The defining polynomial of this extension is given by the minimal polynomial of X , which is equal to the De Moivre's polynomial,

$$X^d + dAX^{d-2} + 2dA^2X^{d-4} + 3dA^3X^{d-6} + \dots + 2dA^{(d-1)/2-1}X^3 + dA^{(d-1)/2}X + B.$$

A more straightforward similar construction is to consider Kummer extensions over quadratic (or small degree) extensions, which yields $X^{2d} + AX^d + B$. From these two specific families of solvable polynomials, we provide, in Section 4.1 and 4.2, hyperelliptic curves for all genus $g \geq 2$ which admit an efficient deterministic encoding function.

2.2 Rational and Deterministic Parameterizations

Given a parameterized family of solvable polynomials $f_{\underline{t}}(X)$, and a genus g , we now substitute a rational function $F_i(Y)$ in some variable Y for each parameter a_i in \underline{a} .

Let $\underline{F}(Y)$ denote the k -tuples of rational functions $(F_1(Y), F_2(Y), \dots, F_k(Y))$. The equation $f_{\underline{F}(Y)}(X)$ now defines a plane algebraic curve C , with variables (X, Y) . The genus of C increases when the degrees of $\underline{F}(Y)$ in Y increase. So if we target some fixed genus g for C , only few degrees for the numerators and denominators of $\underline{F}(Y)$ can occur. Since we can consider coefficients of these rational functions as parameters $\underline{a} = (a_1, \dots, a_{k'})$, this yields a family of curves $C_{\underline{a}}$.

Less easily, it remains then to determine among these $\underline{F}(Y)$ the ones which yield roots $\chi_{\underline{F}(Y)}$ which can be computed in deterministic time. The easiest case is probably when no square root occurs in the computation of $\chi_{\underline{t}}$, since then any choice for $\underline{F}(Y)$ will work, at the expense of some constraint on the finite field. But this is usually not the case, and we might try instead to link these square roots to some algebraic parameterization of an auxiliary algebraic curve

2.3 Minimal Models

In some case (typically hyperelliptic curves), it is worth to derive from the equation for $C_{\underline{a}}$ a minimal model (typically of the form $y^2 = g_{\underline{a}}(x)$). In order to

still have a deterministic encoding with the minimal model, we need explicit birational maps $x = \Lambda_{\underline{a}}(X, Y)$, $y = \Omega_{\underline{a}}(X, Y)$ too. For hyperelliptic curves, the usual way for this is to work with homomorphic differentials defined by $C_{\underline{a}}$. This method is implemented in several computer algebra systems, for instance MAPLE [14] or MAGMA [5]. All in all, we obtain the following encoding for a minimal model $g_{\underline{a}}$:

- Fix some Y as a (non-rational) function of some parameter t so that all the square roots appearing in the expression of $\chi_{\underline{F}}(Y)$ are well defined ;
- Compute $X = \chi_{\underline{F}}(Y)$;
- Compute $x = \Lambda_{\underline{a}}(X, Y)$ and $y = \Omega_{\underline{a}}(X, Y)$.

2.4 Cryptographic Applications

Once we will have found an encoding, it is important for cryptographic applications to study the cardinality of the subset of the curve that we parameterize. This ensures that we obtain convenient weak encodings for hashing into curves primitives (see [6]).

In the degree 3 examples given below, as in the higher genus family given in Section 4.2, we always will be able to deduce from the encoding formulae (sometimes after some resultant computations), a polynomial relation $P_{\underline{a}}(Y, t)$ between any Y of a point of the image and its preimages. Then the number of possible preimages is at most the t -degree of $P_{\underline{a}}(Y, t)$. Factorizing $P_{\underline{a}}(Y, t)$ over \mathbb{F}_q gives then precisely the number of preimages. We detail this process for the genus 1 application of our method in Section 3.1 and sketch how to obtain such a polynomial in other sections.

Hyperelliptic curve cryptography relies on hashing into the Jacobian of a curve. Section 4.3 presents two different ways for accomplishing this task, which rely on encoding into the set of rational points of the curve.

We also need to know *in advance* which values of \mathbb{F}_q cannot be encoded using such functions, in order to deterministically handle such cases. In the genus 1 as in other sections of our paper, this subset is always quite small (never more than several hundred elements) compared to cryptographic sizes, and only depends on the once and for all fixed curve parameters, therefore it can be taken into account and handled appropriately when setting up the cryptosystem. Furthermore, cryptographic encodings of [6] make a heavy use of hash functions onto the finite field before encoding on the curve; the output of the hash function can then be encoded with overwhelming probability.

3 Degree 3 Polynomials

In this section, we consider degree 3 polynomials. After easy changes of variables, any cubic can be written in its “depressed form” $X^3 + 3AX + 2B$, one root of which is

$$\chi_{A,B} = \sqrt[3]{-B + \sqrt{A^3 + B^2}} - \frac{A}{\sqrt[3]{-B + \sqrt{A^3 + B^2}}}.$$

In order to make use of this root while avoiding square roots, aiming at (non-rationally) parameterizing curves of positive genus, we first restrict to finite fields \mathbb{F}_q with q odd and $q \equiv 2 \pmod 3$, so that computing cubic roots can be done thanks to a deterministic exponentiation to the e -th power, $e = 1/3 \pmod{q-1}$. We then need to consider rational functions A and B in Y such that the curve $A(Y)^3 + B(Y)^2 - Z^2$ can be parameterized too.

For non-zero A , let $A(Y) = T(Y)$ for some T and $B(Y) = T(Y)S(Y)$ for some S , this problem is then the same as parameterizing the curve

$$T(Y) + S^2(Y) = Z^2. \tag{3.1}$$

This can be done with rational formulae when this curve is of genus 0, or with non-rational Icart’s formulae when this curve is of genus 1. In the case of irreducible plane curves, this means that T and S are of low degree. Instead of parameterizing an auxiliary curve, we could have directly chosen T and S such that $T(Y) + S(Y)^2 = Z(Y)^2$ for some rational function Z . With comparable degrees for T and S as in the rest of the section, we obtain only genus 0 curves. Thus we have to greatly increase the degree of S and T in order to get higher genus curves. Those curves then have high degree but small genus: they have many singularities.

So, we finally consider in the following degree 3 equations of the form

$$X^3 + 3T(Y)X + 2S(Y)T(Y) = 0. \tag{3.2}$$

We could have considered the case $A = 0$ too, that is polynomials of the form $f_B = X^3 + 2B$. Our experiments in genus 1 and genus 2 yield curves that are isomorphic to hyperelliptic curves of any genus constructed from De Moivre’s polynomials given in Section 4.2. We thus do not study this case further.

3.1 Genus 1 Curves

Parameterization. We made a systematic study of Curves (3.2) of (generic) genus 1 as a function of the degree of the numerators and the denominators of the rational function $S(Y)$ and $T(Y)$. Results are in Tab. 1, where we put altogether columns of compatible degrees. Typically, the first column (S a polynomial of degree at most 2 and T a constant) is a subcase of the second column (S of degree at most 3 and T a constant).

The only case of interest is when $S(Y)$ is a polynomial of degree at most 1 and $T(Y)$ is a polynomial of degree at most 2. When $q \equiv 2 \pmod 3$, these elliptic curves all have a \mathbb{F}_q -rational 3-torsion point, coming from $X = 0$.

Elliptic curves with a \mathbb{F}_q -rational 3-torsion point are known to have very fast addition formulae when given in “generalized” or “twisted” Hessian forms [9, 2]. Since $q \equiv 2 \pmod 3$, we even restrict in the following to classical Hessian elliptic curves.

Let us start from $S(Y) = 3(Y + a)/2$, $T(Y) = -Y/3$, that is curves of the type

$$C_{0,a} : Y^2 + XY + aY = X^3, \quad a \neq 0, 1/27. \tag{3.3}$$

Table 1. Degrees of $S(Y)$ and $T(Y)$ for genus 1 plane curves given by Eq. (3.2)

| | | Degrees | | | | | | | | | | |
|--------------------|------|---------|---|---|---|---|---|---|---|---|---|---|
| $S(Y)$ | Num. | 2 | 3 | 2 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 |
| | Den. | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |
| $T(Y)$ | Num. | 0 | 0 | 1 | 1 | 1 | 2 | 2 | 0 | 0 | 0 | 0 |
| | Den. | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 2 | 2 | 3 |
| Genus of Eq. (3.1) | | 1 | 2 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 2 |

Then, the conic $S^2(Y) + T(Y) = 9/4 Y^2 + (9/2 a - 1/3) Y + 9/4 a^2 = Z^2$ can be classically parameterized “by line” as

$$Y = \frac{12 t^2 - 27 a^2}{36 t - 4 + 54 a}, \quad Z = \frac{36 t^2 + (-8 + 108 a) t + 81 a^2}{72 t - 8 + 108 a},$$

so that $X = \Delta/6 + 2Y/\Delta$ where $\Delta = \sqrt[3]{36Y(3Y + 3a + 2Z)}$.

Besides, Curve (3.3) is birationally equivalent to the Hessian model

$$E_d : x^3 + y^3 + 1 = 3 d x y, \quad d \neq 1, \tag{3.4}$$

with $a = (d^2 + d + 1)/3(d + 2)^3$ and

$$x = \frac{3(d + 2)^2(Y(d + 2) + X)}{3(d + 2)^2 X + d^2 + d + 1}, \quad y = -\frac{d^2 + d + 1 + 3(d + 1)(d + 2)^2 X + 3(d + 2)^3 Y}{3(d + 2)^2 X + d^2 + d + 1}. \tag{3.5}$$

The only remaining case is $d = -2$, that is the Hessian curve E_{-2} (the quadratic twist of the curve E_0 , both have their j -invariant equal to 0). This curve is for instance isomorphic to a curve of the type (3.2) with $S = (1 - 7Y)/4$ and $T = -26(3Y^2 + 1)/27$. We might use this to parameterize E_{-2} , but it is much simpler to start from the curve $Y^2 + Y = X^3$, which can be much more easily parameterized with $Y = t, X = \sqrt[3]{t^2 + t}$. This curve is isomorphic to E_{-2} with $x = (X + 1)/(X + Y), y = (-Y + X - 1)/(X + Y)$.

We summarize these calculations in Algorithm 1.

In addition, we have proved what follows.

Theorem 1. *Let \mathbb{F}_q be the finite field with q elements. Suppose q odd and $q \equiv 2 \pmod 3$. Let E_d/\mathbb{F}_q be the elliptic curve defined by Eq. (3.4).*

Then Algorithm 1 computes a deterministic encoding e_d to E_d , from \mathbb{F}_q^ if $d = -2$ and from $\mathbb{F}_q \setminus \left\{ \frac{(2d+1)(d^2+d+7)}{18(d+2)^3} \right\}$ otherwise, in time $\mathcal{O}(\log^{2+o(1)} q)$.*

Remark 4. This encoding is not defined on all of \mathbb{F}_q , however we can map the missing value to the point at infinity on the curve.

Algorithm 1: HessianEncode

```

input : A Hessian elliptic curve
           $E_d/\mathbb{F}_q : x^3 + y^3 + 1 = 3 dxy, d \neq 1, \text{ and } t \in \mathbb{F}_q.$ 
output: A point  $(x_t : y_t : 1)$  on  $E_d.$ 

if  $d = -2$  then /*  $t \neq 0$  */
   $Y := t; X := (t + t^2)^{1/3 \bmod q-1};$ 
   $x_t := (X + 1)/(X + Y); y_t := (-Y + X - 1)/(X + Y);$ 
  return  $(x_t : y_t : 1)$ 
 $a := \frac{d^2 + d + 1}{3(d + 2)^3};$  /*  $t \neq \frac{(2d + 1)(d^2 + d + 7)}{18(d + 2)^3}$  */
if  $t = \pm 3a/2$  then
   $Y := 0; X := 0;$ 
else /*  $Y \neq 0$  */
   $Y := \frac{12t^2 - 27a^2}{36t + 54a - 4}; \Delta := (36Y(2t + 3a))^{1/3 \bmod q-1};$ 
   $X := \Delta/6 + 2Y/\Delta;$ 
   $x_t := \frac{3(d + 2)^2(Y(d + 2) + X)}{3(d + 2)^2X + d^2 + d + 1};$ 
   $y_t := -\frac{3(d + 1)(d + 2)^2X + 3(d + 2)^3Y + d^2 + d + 1}{3(d + 2)^2X + d^2 + d + 1};$ 
return  $(x_t : y_t : 1)$ 

```

Fig. 1. Encoding on Hessian elliptic curves

Number of curves. A way quantifying the number of curves defined by Eq. (3.4) is to compute their j -invariant. Here, we obtain

$$j_{E_d} = 27 d^3 \frac{(d + 2)^3 (d^2 - 2d + 4)^3}{(d - 1)^3 (d^2 + d + 1)^3}. \tag{3.6}$$

When $q \equiv 2 \pmod 3$, there are exactly $\lfloor q/2 \rfloor$ distinct such invariants. Additionally, one can show that there exists $q - 1$ distinct \mathbb{F}_q -isomorphic classes of Hessian elliptic curves (see [9]).

Cardinality of the image. It is obvious to see that $|\text{Im } e_{-2}| = q - 1$, simply because $Y = t \neq 0$. Now, determining $|\text{Im } e_d|$ for $d \neq 1, -2$ needs some more work, but can still be evaluated exactly.

Theorem 2. *Let $d \neq 1, -2$, then $|\text{Im } e_d| = (q + 1)/2$ if $(d - 1)/(d + 2)$ is a quadratic residue in \mathbb{F}_q and $|\text{Im } e_d| = (q - 1)/2$ otherwise.*

Proof. Let $(x : y : 1)$ be a point on E_d , then there exists a unique point $(X : Y : 1)$ on $C_{0,a}$ sent by Isomorphism (3.5) to $(x : y : 1)$.

Viewed as a polynomial in t , the equation $12t^2 - 36Yt - 54Ya - 27a^2 + 4Y$ has 0 or 2 solutions except when $27Y^2 + (-4 + 54a)Y + 27a^2 = 0$. The latter has no root in Y if $1 - 27a = (d - 1)^3/(d + 2)^3$ is a quadratic non-residue, and

two distinct roots denoted Y_0 and Y_1 otherwise (if $a = 1/27$, the curve $C_{0,a}$ degenerates into a genus 0 curve).

Let us summarize when $(d - 1) / (d + 2)$ is a quadratic residue in \mathbb{F}_q .

- (1 element) If $t \in \left\{ \frac{(2d+1)(d^2+d+7)}{18(d+2)^3} \right\}$, then t is not encodable by e_d ;
- (2 elements) If $t \in \left\{ \pm \frac{d^2+d+1}{2(d+2)^3} \right\}$, then $e_d(t) = (0 : -1 : 1)$;
- (2 elements) If t_i is a (double) root of $12t^2 - (36t - 4 + 54a)Y_i - 27a^2$ with $i = 0, 1$, we obtain two distinct points $e_d(t_i) = (x_{t_i} : y_{t_i} : 1)$;
- ($q - 5$ elements) Else, for each remaining t , there exists exactly one other t' such that $e_d(t) = e_d(t') = (x_t : y_t : 1)$.

We thus obtain $(q - 5)/2 + 2 + 1 = (q + 1)/2$ distinct rational points on the curve. Similarly if $(d - 1) / (d + 2)$ is a quadratic non-residue in \mathbb{F}_q , we obtain $(q - 1)/2$ distinct rational points on E_d . □

Related work. Compared to Icart’s formulae [13], this encoding has two drawbacks of limited practical impact:

- it does not work for any elliptic curves, but only for Hessian curves;
- the subset of the curve which can be parameterized is slightly smaller than in Icart’s case: we get $\simeq q/2$ points against approximately $5/8\#E \pm \lambda\sqrt{q}$.

Nonetheless, it has three major practical advantages:

- recovering the parameter t from a given point $(x : y : 1)$ is much easier: we only have to find the roots of a degree 2 equation instead of a degree 4 one;
- the parameter t only depends on y : we can save half of the bandwidth of a protocol by sending only y and not the whole point $(x : y : 1)$;
- Y is computable using only simple (rational) finite field operations: no exponentiation is required, but it carries the whole information on the encoded point. It is thus preferable for encoding purposes to work on the $C_{0,a}$ model rather than on the Hessian model¹.

3.2 Genus 2 Curves

Parameterizations. In the same spirit as in Section 3.1, we made a systematic study of Curves (3.2) of (generic) genus 2 as a function of the degree of the numerators and the denominators of the rational function $S(Y)$ and $T(Y)$. Results are in Tab. 2.

We can see that there are three cases of interest:

- $S(Y)$ and $T(Y)$ be both a rational function of degree 1 ;
- $S(Y)$ be a rational function of degree 2 and $T(Y)$ be a constant ;
- $S(Y)$ be a constant and $T(Y)$ be a rational function of degree 2.

¹ For example, we could imagine that a limited power device computes the encoded y and sends it to an other device specialized in curve operations, which in turn computes the associated x and realizes the group operations.

Table 2. Degrees of $S(Y)$ and $T(Y)$ for genus 2 plane curves given by Eq. (3.2)

| | | Degrees | | | | | | | | | | | | | | | | | | |
|--------------------|------|---------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $S(Y)$ | Num. | 2 | 0 | 1 | 2 | 2 | 2 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |
| | Den. | 1 | 2 | 2 | 2 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| $T(Y)$ | Num. | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 2 | 2 |
| | Den. | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 1 | 2 | 2 |
| Genus of Eq. (3.1) | | 1 | 1 | 1 | 1 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 3 | 1 | 1 | 1 | 1 | |

We now study the two first cases. We omit the third one because it turns out that it yields curves already obtained in the second case.

$S(Y)$ and $T(Y)$ rational functions of degree 1. Let $S(Y) = (\alpha Y + \beta)/(\gamma Y + \delta)$ and $T(Y) = (\varepsilon Y + \varphi)/(\mu Y + \nu)$, then Curve (3.2) is birationally equivalent to curves of the form $y^2/d^2 = (x^3 + 3ax + 2c)^2 + 8bx^3$ where

$$a = \frac{\delta\varepsilon - \gamma\varphi}{\delta\mu - \gamma\nu}, \quad b = \frac{(\alpha\delta - \gamma\beta)(\mu\varphi - \varepsilon\nu)}{(\delta\mu - \gamma\nu)^2}, \quad c = \frac{\beta\varepsilon - \alpha\varphi}{\delta\mu - \gamma\nu} \quad \text{and} \quad d = (\delta\mu - \gamma\nu).$$

Many of these curves are isomorphic to each other and, without any loss of generality, we can set $c = 1$ and $d = 1$. We thus finally restrict to $S(Y) = -Y$, $T(Y) = (a^2Y + a)/(aY + b + 1)$, so that, when $4a^6b^3 - b^3(b^2 + 20b - 8)a^3 + 4b^3(b + 1)^3 \neq 0$, Curve (3.2) is birationally equivalent to the Weierstrass model of a genus 2 curve,

$$H_{1,a,b} : y^2 = (x^3 + 3ax + 2)^2 + 8bx^3, \tag{3.7}$$

with $x = X$ and $y = -4aY + X^3 + 3aX - 2$.

Besides, Curve

$$S^2(Y) + T(Y) = Y^2 + (a^2Y + a)/(aY + 1 + b) = Z^2 \tag{3.8}$$

is birationally equivalent to the Weierstrass elliptic curve

$$V^2 = U^3 + (-a^6 + 2(b + 1)(2b - 1)a^3 - (b + 1)^4) \frac{U}{3} + \frac{1}{27} (2a^9 + 3(2 - 2b + 5b^2)a^6 - 6(2b - 1)(b + 1)^3a^3 + 2(b + 1)^6). \tag{3.9}$$

The latter can now be parameterized with Icart’s method. This yields

$$U = \frac{1}{6} \sqrt[3]{\frac{2\delta}{t^2}} + \frac{t^2}{3}, \quad V = \frac{1}{6} \sqrt[3]{2\delta t} + \frac{t^3}{6} + \frac{1}{6t} (-a^6 + 2(b + 1)(2b - 1)a^3 - (b + 1)^4)$$

with

$$\delta = -t^8 + (-12(b + 1)(2b - 1)a^3 + 6a^6 + 6(b + 1)^4)t^4 + (12(2b - 5b^2 - 2)a^6 - 8(b + 1)^6 - 8a^9 + 24(2b - 1)(b + 1)^3a^3)t^2 + 3(a^6 - 2(b + 1)(2b - 1)a^3 + (b + 1)^4)^2$$

Algorithm 2: Genus2TypeAEncode

input : A curve $H_{1,a,b}$ defined by Eq. (3.7) on \mathbb{F}_q , an element $t \in \mathbb{F}_q \setminus \mathcal{S}_1$
output: A point $(x_t : y_t : 1)$ on $H_{1,a,b}$
 $\delta := -t^8 + (-12(b+1)(2b-1)a^3 + 6a^6 + 6(b+1)^4)t^4 + (12(2b-5b^2-2)a^6 - 8(b+1)^6 - 8a^9 + 24(2b-1)(b+1)^3a^3)t^2 + 3(a^6 - 2(b+1)(2b-1)a^3 + (b+1)^4)^2$;
 $U := ((2\delta/t^2)^{1/3 \bmod q-1} + 2t^2)/6$;
 $V := (2\delta t)^{1/3 \bmod q-1}/6 + t^3/6 + (-a^6 + 2(b+1)(2b-1)a^3 - (b+1)^4)/6t$;
 $W := -3Ua + a((b+1)^2 + a^3)$; $Y := (3(b+1)U + (2b-1)a^3 - (b+1)^3)/W$; $Z := 3V/W$;
 $T := (a^2Y + a)/(aY + b + 1)$; $\Delta := (T(Z + Y))^{1/3 \bmod q-1}$;
 $x_t := \Delta - T/\Delta$; $y_t := -4aY + X^3 + 3aX - 2$;
return $(x_t : y_t : 1)$

Fig. 2. Encoding on genus 2 curves (of the type A)

Now, back by the birational change of variables between Curve (3.9) and Curve (3.8), we get Y and Z from U and V (cf. Algorithm 2 for precise formulae). Let now $\Delta = \sqrt[3]{T(Y)(Z - S(Y))}$, then $X = \Delta - T(Y)/\Delta$.

So, we obtain the following theorem.

Theorem 3. *Let \mathbb{F}_q be the finite field with q elements. Suppose q odd and $q \equiv 2 \pmod 3$. Let $H_{1,a,b}/\mathbb{F}_q$ be the hyperelliptic curve of genus 2 defined by Eq. (3.7).*

Then, Algorithm 2 computes a deterministic encoding $e_{1,a,b} : \mathbb{F}_q^ \setminus \mathcal{S}_1 \rightarrow H_{1,a,b}$, where \mathcal{S}_1 is a subset of \mathbb{F}_q of size at most 35, in time $\mathcal{O}(\log^{2+o(1)} q)$.*

Proof. The previous formulae define a deterministic encoding provided that t , W , $aY + b + 1$ and Δ are not 0.

The condition $W = 0$ yields a polynomial in t of degree 8, we thus have at most 8 values for which $W = 0$. Similarly, the condition $aY + b + 1 = 0$ yields at most 8 additional values for which $W = 0$.

Now $\Delta = 0$ if and only if $T = 0$ or $Z = -Y$. The condition $T = 0$ yields 8 additional values. Similarly, the condition $Z + Y = 0$ yields a polynomial in t of degree 10, we thus have in this case at most 18 values for which $\Delta = 0$.

The total number of field elements which cannot be encoded finally amounts to at most 35. □

Cardinality of the image. Let (X, Y) be a rational point on a $C_{1,a,b,c}$ curve, let t be a possible preimage of (X, Y) by our encoding $e_{1,a,b}$. Then there exists a polynomial relation in Y and t of degree at most 8 in t (cf. Algorithm 2). Hence (X, Y) has at most 8 preimages by $e_{1,a,b}$. Therefore, $|\text{Im } e_{1,a,b}| \geq (q - 35)/8$.

Number of curves. Igusa invariants of these curves are equal to

$$\begin{aligned}
 J_2 &= 2^6 3 (-9a^3 + 4b^2 + 4b - 9), \\
 J_4 &= 2^{10} 3 (-9b(4b - 15)a^3 + 4b(b + 1)(2b^2 + 2b - 27)), \\
 J_6 &= 2^{14} (729a^6b^2 - 216b^2(2b^2 + 3b + 21)a^3 + 16b^2(4b^2 + 4b + 81)(b + 1)^2), \\
 J_8 &= 2^{18} 3 (-6561a^9b^2 + 2916b^2(-7 + b^2 + 13b)a^6 \\
 &\quad - 144b^2(4b^4 + 63b^3 + 450b^2 - 149b - 810)a^3 \\
 &\quad + 64b^2(b^4 + 2b^3 + 154b^2 + 153b - 729)(b + 1)^2), \\
 J_{10} &= 2^{28} 3^6 (4a^6b^3 - b^3(b^2 + 20b - 8)a^3 + 4b^3(b + 1)^3).
 \end{aligned}$$

The geometric locus of these invariants is a surface of dimension 2 given by a homogeneous equation of degree 90 (which is far too large to be written here). Consequently, Eq. (3.7) defines $\mathcal{O}(q^2)$ distinct curves over \mathbb{F}_q .

S(Y) be a rational function of degree 2. Let now $S(Y) = (\alpha Y^2 + \beta Y + \gamma) / (\delta Y^2 + \varepsilon Y + \varphi)$ and $T(Y) = \kappa$, then Curve (3.2) is birationally equivalent to curves of the form $y^2/\lambda = (x^3 + 3\mu x + 2a)^2 + 4b$ where

$$\lambda = \varepsilon^2 - 4\varphi\delta, \quad \mu = \kappa, \quad a = \frac{\kappa}{\lambda}(\varepsilon\beta - 2\delta\gamma - 2\varphi\alpha) \quad \text{and} \quad b = \frac{\kappa^2}{\lambda}(\beta^2 - 4\alpha\gamma) - a^2.$$

Many of these curves are isomorphic to each other and, without any loss of generality, we can set λ and μ to be either any quadratic residues (for instance $\lambda, \mu = 1$) or any non-quadratic residues (for instance $\lambda, \mu = -3$ because $q \equiv 2 \pmod 3$).

We finally arrive to

$$S(Y) = \frac{\lambda(a - u)Y^2 - 4vY - 4(a + u)}{\mu(\lambda Y^2 - 4)} \quad \text{and} \quad T(Y) = \mu,$$

where $u = \mu^3/2w - w/2 - a$ for some $w \in \mathbb{F}_q^*$. Then, when $b^3\lambda^{10}(\mu^6 + 2\mu^3a^2 - 2b\mu^3 + a^4 + 2ba^2 + b^2) \neq 0$, Curve (3.2) is birationally equivalent to the Weierstrass model of a genus 2 curve,

$$H_{2,\lambda,\mu,a,v,w} : y^2/\lambda = (x^3 + 3\mu x + 2a)^2 + 4b, \tag{3.10}$$

where $b = v^2/\lambda - u^2$ for some v in \mathbb{F}_q , $x = X$ and $y = \lambda(X^3/2 + 3\mu X/2 + a - u)Y - 2v$.

We may remark that computing v and w from b is the same as computing a point $(v : w : 1)$ on the elliptic curve $v^2/\lambda - (\mu^3/2w - w/2 - a)^2 - b = 0$. This can be done in deterministic time from Icart’s formulae when one can exhibit a \mathbb{F}_q -rational bilinear change of variable between this curve and a cubic Weierstrass model, typically when $\lambda = 1$ (but no more when $\lambda = -3$).

Besides, let $z = w/2 + r^3/2w$ and thus $(u + a)^2 + r^3 = z^2$, then

$$\begin{aligned} \mu^2(\lambda Y^2 - 4)^2(S(Y)^2 + T(Y)) &= -\lambda^2(4ua - z^2)Y^4 - 8\lambda v(a - u)Y^3 \\ &\quad - 8\lambda(4\mu^3 - 3z^2 - 2b + 6ua + 4a^2)Y^2 + 32v(u + a)Y + 16z^2 = Z^2 \end{aligned} \tag{3.11}$$

is birationally equivalent to the Weierstrass elliptic curve

$$\begin{aligned} V^2 &= U^3 + 2^8\lambda^2(-\mu^6 + (b - 2a^2)\mu^3 - (a^2 + b)^2)U/3 + \\ &\quad 2^{12}\lambda^3(2\mu^9 + (6a^2 - 3b)\mu^6 - 3(a^2 + b)(b - 2a^2)\mu^3 + 2(a^2 + b)^3)/3^3. \end{aligned} \tag{3.12}$$

The latter can now be parameterized with Icart’s method. This yields

$$U = \frac{1}{6} \sqrt[3]{\frac{2\delta}{t^2} + \frac{t^2}{3}}, \quad V = \frac{1}{6} \sqrt[3]{2\delta t} + \frac{t^3}{6} + 128(-\mu^6 + (b - 2a^2)\mu^3 - (b + a^2)^2) \frac{\lambda^2}{3t}$$

with

$$\begin{aligned} \delta &= -t^8 + 2^9 3(\mu^6 + (-b + 2a^2)\mu^3 + (a^2 + b)^2)\lambda^2 t^4 + \\ &\quad 2^{14}(-2\mu^9 - (6a^2 - 3b)\mu^6 + 3(a^2 + b)(b - 2a^2)\mu^3 - 2(a^2 + b)^3)\lambda^3 t^2 + \\ &\quad 2^{16} 3(\mu^{12} + (-2b + 4a^2)\mu^9 + (3b^2 + 6a^4)\mu^6 + 2(a^2 + b)^2(-b + 2a^2)\mu^3 + (a^2 + b)^4)\lambda^4. \end{aligned} \tag{3.13}$$

Again, back by a birational change of variables between Curves (3.12) and (3.11), we get Y and Z from U and V (cf. Algorithm 3 for precise formulae). Let now $\Delta = \sqrt[3]{T(Y)(Z/\mu(\lambda Y^2 - 4) - S(Y))}$, then $X = \Delta - T(Y)/\Delta$.

Algorithm 3: Genus2TypeBEncode

```

input : A curve  $H_{2,\lambda,\mu,a,v,w}$  defined by Eq. (3.10) on  $\mathbb{F}_q$ , an element  $t \in \mathbb{F}_q \setminus \mathcal{S}_2$ .
output: A point  $(x_t : y_t : 1)$  on  $H_{2,\lambda,\mu,a,v,w}$ 
 $u := -(2aw + w^2 - r^3)/2w$ ;  $b := v^2/t - u^2$ ;  $z := (w^2 + r^3)/2w$ ;
 $\delta := -t^8 + 2^9 3(\mu^6 + (-b + 2a^2)\mu^3 + (a^2 + b)^2)\lambda^2 t^4 +$ 
 $\quad 2^{14}(-2\mu^9 - (6a^2 - 3b)\mu^6 + 3(a^2 + b)(b - 2a^2)\mu^3 - 2(a^2 + b)^3)\lambda^3 t^2 +$ 
 $\quad 2^{16} 3(\mu^{12} + (-2b + 4a^2)\mu^9 + (3b^2 + 6a^4)\mu^6 + 2(a^2 + b)^2(-b + 2a^2)\mu^3 + (a^2 + b)^4)\lambda^4$ ;
 $U := ((2\delta/t^2)^{1/3} \bmod q^{-1} + 2t^2)/6$ ;
 $V := (2\delta t)^{1/3} \bmod q^{-1}/6 + t^3/6 + 128(-\mu^6 + (b - 2a^2)\mu^3 - (b + a^2)^2)\lambda^2/3t$ ;
 $W := -9U^2 - 48\lambda(-3z^2 - 2b + 6ua + 4a^2 + 4\mu^3)U + 256(-4\mu^6 + (6z^2 + a^2 - 12ua + 4b)\mu^3 +$ 
 $\quad (b + a^2)(5a^2 + 6ua - b - 3z^2))\lambda^2$ ;
 $Y := (-288v(u + a)U - 72zV + 1536\lambda v(bu + a^3 - 2\mu^3u + ab + a\mu^3 + ua^2))/W$ ;
 $Z := -( -324zU^4 + (6912\lambda\mu^3z + 1728\lambda z(-3z^2 - 2b + 6ua + 4a^2))U^3 - 2592v(u + a)U^2V$ 
 $\quad + (-27648\lambda^2z(b + a^2)(2a^2 + 6ua - 4b - 3z^2) + 193536\lambda^2z\mu^6 - 27648\lambda^2z(-5a^2 - 12ua + 6z^2 + 7b)\mu^3)U^2$ 
 $\quad + (27648\lambda v(-2u + a)\mu^3 + 27648\lambda v(b + a^2)(u + a))UV + (49152\lambda^3z(36a^3u - 18a^2z^2 + 12a^4 + 9z^2b + 30b^2$ 
 $\quad - 12a^2b - 18aub)\mu^3 + 49152\lambda^3z(-6b + 18ua + 12a^2 - 9z^2)\mu^6 + 49152\lambda^3z(b + a^2)^2(4a^2 + 18ua$ 
 $\quad - 14b - 9z^2) + 196608\lambda^3\mu^9z)U + (-73728v\lambda^2(b + a^2)^2(u + a) - 73728v\lambda^2(4u - 8a)\mu^6 - 73728v\lambda^2$ 
 $\quad (-4bu + 9z^2a - 7a^3 - 13ua^2 + 2ab)\mu^3)V - 7340032\lambda^4\mu^{12}z - 262144\lambda^4z(60ua - 56b + 85a^2 - 30z^2)\mu^9$ 
 $\quad - 262144\lambda^4z(b + a^2)(31a^4 + 72a^3u - 10a^2b - 36a^2z^2 + 18aub + 13b^2 - 9z^2b)\mu^3 - 262144\lambda^4z(b + a^2)^3$ 
 $\quad (a^2 + 6ua - 5b - 3z^2) - 262144\lambda^4z(15b^2 + 87a^4 - 63a^2z^2 + 45z^2b - 90aub - 33a^2b + 126a^3u)\mu^6)/W^2$ ;
 $S := (-u + a)Y^2\lambda - 4vY - 4a - 4u$ ;  $\Delta := \sqrt[3]{(Z - S)/(\lambda Y^2 - 4)}$ ;
 $x_t := \Delta - \mu/\Delta$ ;  $y_t := \lambda(X^3/2 + 3\mu X/2 + a - u)Y - 2v$ ;
return  $(x_t : y_t : 1)$ 

```

Fig. 3. Encoding on genus 2 curves (of the type B)

So, we obtain the following theorem.

Theorem 4. *Let \mathbb{F}_q be the finite field with q elements. Suppose q odd and $q \equiv 2 \pmod 3$. Let $H_{2,\lambda,\mu,a,v,w}/\mathbb{F}_q$ be the hyperelliptic curve of genus 2 defined by Eq. (3.10).*

Then, Algorithm 3 computes a deterministic encoding $e_{2,\lambda,\mu,a,v,w} : \mathbb{F}_q^ \setminus \mathcal{S}_2 \rightarrow H_{2,\lambda,\mu,a,v,w}$, where \mathcal{S}_2 is a subset of \mathbb{F}_q of size at most 233, in time $\mathcal{O}(\log^{2+o(1)} q)$.*

Proof. The previous formulae defines a deterministic encoding provided that t , W , $\lambda Y^2 - 4$ and $Z - S$ are not 0.

The condition $W = 0$ yields a polynomial in U of degree 2, we thus have at most 2 values for U for which $W = 0$. Each value of U then yields a polynomial in t , derived from δ , of degree 8. We thus have at most 16 values for t to avoid in this case.

The condition $\lambda Y^2 - 4 = 0$ similarly yields 2 values for Y . Each such value yields in return a polynomial of degree 2 in U , and degree 1 in V , which can be seen as a curve in t and $\tau = \sqrt[3]{2t\delta}$ of degree at most 6. Besides $\tau^3 = 2t\delta$ is a curve of degree at most 9. Bezout's theorem yields thus a maximal number of $2 \times 6 \times 9 = 108$ intersection points, or equivalently values for t , to avoid in this case.

Finally, the condition $Z = S$ can be seen as a curve in t and τ of degree 12. Thus, this yields a maximal number of $12 \times 9 = 108$ values too.

So, the total number of field elements which cannot be encoded finally amounts to at most $1 + 16 + 2 \times 108 = 233$. □

Cardinality of the image. Let (X, Y) be a rational point on $H_{2,\lambda,\mu,a,v,w}$ and t a preimage by $e_{2,\lambda,\mu,a,v,w}$. Then we have seen in the proof of Theorem 4 that t and $\tau = \sqrt[3]{2t\delta}$ are defined as intersection points of two curves, one of degree 7 parameterized by Y and the other one of degree 9 from the definition of δ . In full generality, this might yield for some curves and some of their points a total number of at most 54 t 's. Therefore, $|\text{Im } e_{1,a,b}| \geq (q - 233)/63$.

Number of curves. Igusa invariants of these curves are equal to

$$\begin{aligned} J_2 &= -2^6 3 \lambda^2 (9 \mu^3 + 9 a^2 + 10 b), \\ J_4 &= 2^9 3 b \lambda^4 (297 \mu^3 + 54 a^2 + 55 b), \\ J_6 &= 2^{14} b^2 \lambda^6 (-6480 \mu^3 + 81 a^2 + 80 b), \\ J_8 &= -2^{16} 3 b^2 \lambda^8 (31347 \mu^6 - 134136 \mu^3 a^2 - 158310 b \mu^3 + 11664 a^4 + 23940 ba^2 + 12275 b^2), \\ J_{10} &= -2^{24} 3^6 b^3 \lambda^{10} (\mu^6 + 2 \mu^3 a^2 - 2 b \mu^3 + a^4 + 2 ba^2 + b^2). \end{aligned}$$

Here, the geometric locus of these invariants is a surface of dimension 2 given by a homogeneous equation of degree 30,

$$\begin{aligned} &11852352 J_2^5 J_{10}^2 + 196992 J_2^5 J_4 J_6 J_{10} - 362998800 J_2^3 J_4 J_{10}^2 + 64 J_2^6 J_6^3 - 636672 J_2^4 J_6^2 J_{10} \\ &- 895349625 J_2^2 J_6 J_{10}^2 - 64097340625 J_{10}^3 - 373248 J_2^4 J_4^3 J_{10} - 4466016 J_2^3 J_4^2 J_6 J_{10} \\ &+ 2903657625 J_2 J_4^2 J_{10}^2 - 3984 J_2^4 J_4 J_6^3 + 606810 J_2^2 J_4 J_6^2 J_{10} + 3383973750 J_4 J_6 J_{10}^2 + 1647 J_2^3 J_6^4 \\ &+ 49583475 J_2 J_6^3 J_{10} + 11290752 J_2^2 J_4^4 J_{10} + 38072430 J_2 J_4^3 J_6 J_{10} + 76593 J_2^2 J_4^2 J_6^3 \\ &- 115457700 J_4^2 J_6^2 J_{10} + 20196 J_2 J_4 J_6^4 - 530604 J_6^5 - 85386312 J_4^5 J_{10} - 468512 J_4^3 J_6^3. \end{aligned}$$

This shows that Eq. (3.10) defines $\mathcal{O}(q^2)$ distinct curves over \mathbb{F}_q .

4 Hyperelliptic Curves of Any Genus

In this section, we present two families of parametric polynomials which provide deterministic parameterizable hyperelliptic curves of genus $g \geq 2$.

4.1 Quasiquadratic Polynomials

Curves of the form $y^2 = f(x^d)$ where f is a family of solvable polynomials whatever is its constant coefficient may yield parameterizable hyperelliptic curves. Typically, we may consider polynomials f of degree 2, 3 or 4 or some solvable

families of higher degree polynomials. Here, we restrict ourselves to quadratic polynomials since it yields non trivial hyperelliptic curves for any genus.

We define quasiquadratic polynomials as follows.

Definition 1 (Quasiquadratic polynomials). *Let \mathbb{K} be a field and d be an integer coprime with $\text{char } \mathbb{K}$. The family of quasiquadratic polynomials $q_{a,b}(x) \in \mathbb{K}[x]$ of degree $2d$ is defined for $a, b \in \mathbb{K}$ by $q_{a,b}(x) = x^{2d} + ax^d + b$.*

Quasiquadratic polynomials define an easily parameterized family of hyperelliptic curves $y^2 = q_{a,b}(x)$ (see Algorithm 4). When d does not divide $q-1$ and when $a \neq 0$, these curves are isomorphic to curves $y^2 = q_{1,a}(x)$ by the variable substitution $x \rightarrow a^{1/d}x$. When $a = 0$, we are reduced to the unique well-known curve $y^2 = x^{2d} + b$ which can be parameterized by $t \mapsto \left(\sqrt[d]{(-b + t^2)/(2t)}, (b + t^2)/(2t) \right)$.

Algorithm 4: QuasiQuadraticEncode

input : A curve $H_a : x^{2d} + x^d + a = y^2$, and $t \in \mathbb{F}_q \setminus \{1/2\}$.
output: A point $(x_t : y_t : 1)$ on H_a

$\alpha := (t^2 - a)/(1 - 2t);$
 $x_t := \alpha^{1/d}; y_t := (-a + t - t^2)/(1 - 2t);$
return $(x_t : y_t : 1)$

Fig. 4. Encoding on quasiquadratic curves

Theorem 5. *Let \mathbb{F}_q be the finite field with q elements. Suppose q odd and $q \neq 2, 3$ and d coprime with $q - 1$. Let $H_a/\mathbb{F}_q : y^2 = x^{2d} + x^d + a$ be an hyperelliptic curve where a is such that the quasiquadratic polynomial $q_{1,a}$ has a non-zero discriminant over \mathbb{F}_q .*

Algorithm 4 computes a deterministic encoding $e_a : \mathbb{F}_q^ \setminus \{1/2\} \rightarrow H_a$ in time $\mathcal{O}(\log^{2+o(1)} q)$.*

Genus of H_a . Let $q_{1,a} \in \mathbb{F}_q[X]$ and $H_a : q_{1,a}(x) = y^2$, where $q_{1,a}$ has degree $2d$. We have requested that the discriminant of $q_{1,a}$ is not 0. This implies that $q_{1,a}$ has exactly $2d$ distinct roots. Thus H_a has genus $d - 1$ provided H_a has no singularity except at the point at infinity.

It remains to study the points of the curve where both derivatives in x and y are simultaneously 0. This implies $y = 0$. Thus the only singular points are the common roots of $q_{1,a}(x)$ and its derivative. Since we request that the discriminant of $q_{1,a}$ is not 0, there are no singular points.

For $d = 3$, H_a is the well known family of genus 2 curves with automorphism group D_{12} [7]. The geometric locus of these curves is a one-dimensional variety in the moduli space. Moreover, when $x \rightarrow x^d$ is invertible over \mathbb{F}_q , these curves all have exactly $q + 1$ \mathbb{F}_q -points (but they have a much better distributed number of \mathbb{F}_{q^2} -points).

The encoding. The parameterization is quite simple. Let $H_a : x^{2d} + x^d + a = y^2$ be a quasiquadratic hyperelliptic curve. Setting $x = \alpha^{1/d}$ reduces the parameterization of H_a to the parameterization of the conic $\alpha^2 + \alpha + a - y^2 = 0$, which easily gives $\alpha = (-a + t^2)/(1 - 2t)$ and $y = (-a + t - t^2)/(1 - 2t)$ for some parameter t . We finally obtain Algorithm 4.

Cardinality of the image.

Theorem 6. *Given a rational point $(x : y : 1)$ on $H_a : q_{1,a}(x) = y^2$, the equation $e_a(t) = (x : y : 1)$ has exactly 1 solution. Thus, $|\text{Im } e_a| = q - 1$*

Proof. Let $\alpha = x^d$, then t is a solution of the degree 1 equation $y + \alpha = ta/(a - 2t)$. □

4.2 De Moivre’s Polynomials

This well-known family of degree 5 polynomials was first introduced by De Moivre for the study of trigonometric equalities and its study from a Galoisian point of view was done by Borger in [4]. This definition can be easily generalized for any odd degree.

Definition 2 (De Moivre’s polynomials). *Let \mathbb{K} be a field and d be an odd integer coprime with $\text{char } \mathbb{K}$. The family of De Moivre’s polynomials $p_{a,b}(x) \in \mathbb{K}[x]$ of degree d is defined for $a, b \in \mathbb{K}$ by*

$$p_{a,b}(x) = x^d + da x^{d-2} + 2da^2 x^{d-4} + 3da^3 x^{d-6} + \dots + 2da^{(d-1)/2-1} x^3 + da^{(d-1)/2} x + b.$$

Examples. De Moivre’s polynomials of degree 5 are $x^5 + 5ax^3 + 5a^2x + b$. De Moivre’s polynomials of degree 13 are $x^{13} + 13ax^{11} + 26a^2x^9 + 39a^3x^7 + 39a^4x^5 + 26a^5x^3 + 13a^6x + b$.

Borger proved in [4] that De Moivre’s polynomials of degree 5 are solvable by radical, the same is true for De Moivre’s polynomials of any degree.

Lemma 1 (Resolution of De Moivre’s polynomials). *Let $p_{a,b}$ be a De Moivre’s polynomial of degree d , let θ_0 and θ_1 be the roots of $q_{a,b}(\theta) = \theta^2 + b\theta - a^d$, then the roots of $p_{a,b}$ are*

$$(\omega_k \theta_0^{1/d} + \omega_k^{d-1} \theta_1^{1/d})_{0 \leq k < d}$$

where $(\omega_k)_{0 \leq k < d}$ are the d -th roots of unity.

Proof. As in the case of degree 5 (see [4]), we do the variable substitution $x = \gamma - a/\gamma$, then γ^d is a root of the polynomial $q_{a,b}(\theta)$. □

De Moivre’s polynomials also define a family of deterministically parameterized hyperelliptic curves for any genus.

Algorithm 5: DeMoivreEncode

input : A curve $H : p_{a,b}(x) - y^2 = 0$ and $t \in \mathbb{F}_q^* \setminus \mathcal{S}$.
output: A point $(x_t : y_t : 1)$ on H

if $a = 0$ **then**
 return $((t^2 - b)^{1/d \bmod q-1} : t : 1)$

$\delta := -(3a^d + b^2 + t^4)/6t - 2b^3/27 - a^d b/3 - t^6/27$; $A := \delta^{1/3 \bmod q-1} + t^2/3$;
 $Y := tA - (3a^d + b^2 + t^4)/(6t)$;
 $\alpha := 3a^d/(-3A + b)$;
 $y_t := -3Y/(-3A + b)$; $x_t := \alpha^{1/d \bmod q-1} + (-a^d/\alpha)^{1/d \bmod q-1}$;
return $(x_t : y_t : 1)$

Fig. 5. Encoding on De Moivre’s curves

Theorem 7. *Let \mathbb{F}_q be the finite field with q elements. Suppose q odd and $q \equiv 2 \pmod 3$ and d coprime with $q - 1$. Let $H_{a,b}/\mathbb{F}_q : y^2 = p_{a,b}(x)$ be the hyperelliptic curve where $p_{a,b}$ is a De Moivre polynomial defined over \mathbb{F}_q with non-zero discriminant.*

Algorithm 5 computes a deterministic encoding $e_{a,b} : \mathbb{F}_q^ \setminus \mathcal{S} \rightarrow H_{a,b}$, where \mathcal{S} is a subset of \mathbb{F}_q of size at most 7, in time $\mathcal{O}(\log^{2+o(1)} q)$.*

Conversely, given a point on H we study how many elements in \mathbb{F}_q yield this point.

Theorem 8. *Given a point $(x : y : 1) \in H_{a,b}(\mathbb{F}_q)$, we can compute the solutions s of the equation $e_{a,b}(s) = (x : y : 1)$ in time $\mathcal{O}(\log^{2+o(1)} q)$. There are at most 8 solutions to this equation.*

We give below proofs of these two theorems.

Genus and dimension of $H_{a,b}$. As in Section 4.1, since we request the discriminant of $q_{a,b}$ to be nonzero, there is no singularity except the point at infinity. Thus the genus of $H_{a,b}$ is $(d - 1)/2$.

The encoding. Thanks to Lemma 1, parameterizing rational points on $H_{a,b} : p_{a,b}(x) = y^2$ amounts to finding roots of $\theta^2 + (b - y^2)\theta - a^d$. Let them be α, α' , then we have $x = \alpha^{1/d} + \alpha'^{1/d}$, $\alpha\alpha' = -a^d$ and $\alpha + \alpha' = y^2 - b$. Thus $\alpha^2 - a^d = \alpha y^2 - b\alpha$. This is a genus 1 curve with variable α, y which is birationally equivalent to $Y^2 = A^3 + (-a^d - \frac{1}{3}b^2)A + \frac{2}{27}b^3 + \frac{1}{3}a^d b$, with $\alpha = 3a^d/(-3A + b)$ and $y = -3Y/(-3A + b)$.

This curve can be parameterized with Icart’s method. This yields $A = \sqrt[3]{\delta} + t^2/3$, $Y = tA - (3a^d + b^2 + t^4)/6t$ where $\delta = (-53a^d + b^2 + t^4)/6t - 2b^3/27 - a^d b/3 - t^6/27$. We finally obtain Algorithm 5.

Restrictions. Previous necessary conditions on an encoding are also sufficient to give an encoding for $t \in \mathbb{F}_q$ provided that every variable substitution is computable.

In order to compute A and Y using the encoding from [13], we need $t \neq 0$. Then computing y and α from A and Y we also request $-3A + b \neq 0$, that is $\delta \neq (b/3 - t^2/3)^3$. This amounts to a degree 7 equation, thus at most 7 elements of \mathbb{F}_q are not encodable.

Complexity. Our encoding function uses one Icart’s encoding, of complexity $\mathcal{O}(\log^{2+o(1)} q)$ operations in \mathbb{F}_q , two exponentiations for computing d -th roots and a constant number of field operations. The total amounts to $\mathcal{O}(\log^{2+o(1)} q)$ running time.

Computation of $e_{a,b}^{-1}$. Let $(x : y : 1)$ be a point on $H_{a,b}$. The polynomial $\beta^2 + x\beta - \sqrt[3]{(-a^d)}$ has at most two roots. Let β be one, and $\alpha = \beta^5$. Let then $A = 1 - 3(b\alpha - 3a^d)/\alpha$ and $Y = -ya^d/\alpha$, we are reduced to finding the solutions of an Icart’s encoding. It admits at most 4 solutions per α , thus there are at most 8 solutions to the equation $e_{a,b}(t) = (x : y : 1)$.

Genus 2 case. In this case we are interested in the dimension of the family of curves defined by De Moivre’s polynomials, $H : y^2 = x^5 + 5ax^3 + 5a^2x + b$. We have computed their Igusa invariants,

$$J_2 = 700 a^2, \quad J_4 = 13750 a^4, \quad J_6 = -2500 a(3 a^5 + 32 b^2),$$

$$J_8 = -15625 a^3(3109 a^5 + 896 b^2), \quad J_{10} = 800000 (4 a^5 + b^2)^2,$$

from which it is easy to derive numerous algebraic relations. This reduces the set of curves from an expected q^2 because of the two parameters a and b to a set of cardinality $\mathcal{O}(q)$.

4.3 Encoding into the Jacobian of an Hyperelliptic Curve

Let H be a genus g hyperelliptic curve defined over a finite field \mathbb{F}_q arising from the families defined in the previous sections 3.2, 4.1 and 4.2. We provide deterministic functions e_H which construct rational points on H from elements in $\mathbb{F}_q \setminus \mathcal{S}$, where \mathcal{S} is a small subset of \mathbb{F}_q which depends on the definition of H . In this section, we present two straightforward strategies for encoding divisors in $\mathcal{J}_H(\mathbb{F}_q)$ the Jacobian of H . This problem was also studied by Farashahi et al. [10].

Recall that each class in $\mathcal{J}_H(\mathbb{F}_q)$ can be uniquely represented by a reduced divisor. A divisor D is said to be reduced when it is a formal sum of points $\sum_{i=1}^r P_i - rP_\infty$ with $r \leq g$, $P_i \neq -P_j$ for $i \neq j$ and this sum is invariant under the action of the Galois group $\text{Gal}(\overline{\mathbb{F}_q}/\mathbb{F}_q)$.

Encoding 1-smooth reduced divisors. There is a particular subset, denoted by \mathcal{D}_1 , of reduced divisors which are called 1-smooth. These divisors are the ones with only rational points in their support. From our encoding function e_H , one easily deduces a function providing elements in \mathcal{D}_1 : in a first step, a set of $r \leq g$ points

(none of these points in this set is the opposite of another one) is produced then a divisor is constructed from this set. This first step can be done deterministically by computing g points with e_H and eliminating possible collisions after negation. When q is large enough, the proportion of \mathcal{D}_1 in $\mathcal{J}_H(\mathbb{F}_q)$ is $\approx 1/g!$ moreover, since e_H is not surjective, this function may be not surjective too. If one wants to construct more general reduced divisors, another strategy has to be used.

Extension of the base field and encoding. In the definition of the encoding e_H , we assume specific conditions on the base field \mathbb{F}_q so that some power functions are deterministically bijective. If one wants to directly encode in the Jacobian of an hyperelliptic curve H defined over \mathbb{F}_q , one can change the conditions in the following way. These specific conditions are now assumed for the extension field \mathbb{F}_{q^g} (and thus no more on \mathbb{F}_q). The function e_H becomes an encoding e'_H from $\mathbb{F}_{q^g} \setminus \mathcal{S}'$ (where the set \mathcal{S}' can be computed in the same manner as \mathcal{S}) to the set of \mathbb{F}_{q^g} -rational points of H . From this new function e'_H one can compute a set of k points in $H(\mathbb{F}_{q^g})$ such that the sum of their degree over \mathbb{F}_q is less than g . By constructing the \mathbb{F}_q -conjugates of these points and eliminating the possible collision after negation, we deduce a reduced divisor of $\mathcal{J}_H(\mathbb{F}_q)$. This second strategy is more general than the former but it does not assume the same conditions on the field \mathbb{F}_q .

Like the previous encodings, these two presented here are clearly “weak encodings” in the sense of [6].

5 Conclusion and Future Work

We have almost extensively studied families of genus 1 and 2 curves which admit a deterministic algebraic encoding using the resolution of a degree 3 polynomial. We come to a new encoding map for Hessian elliptic curves and we give, for the first time to our knowledge, encoding maps for large families of genus 2 curves. We have also sketched families of higher genus hyperelliptic curves whose deterministic algebraic parameterization is based on solvable polynomials of higher degree arising from Kummer theory.

On-going work is being done to extend these families to finite fields of small characteristic. A natural question is to generalize the method to solvable degree 5 polynomials too, in the hope to first find a deterministic algebraic parameterization of every genus 2 curve, then of families of higher genus curves.

Acknowledgments

This work was partially supported by the French ANR under the ANR-09-BLAN-0020-01 CHIC, ANR-09-BLAN-0371-01 EXACTA and ANR-09-JCJ-0064-01 CAC projects.

References

- [1] Atkin, A.O.L., Morain, F.: Elliptic curves and primality proving. *Mathematics of Computation* 61(203), 29–68 (1993)
- [2] Bersntein, D.J., Kohel, D., Lange, T.: Twisted Hessian curves, <http://www.hyperelliptic.org/EFD/g1p/auto-twistedhessian.html>
- [3] Boneh, D., Franklin, M.: Identity-Based Encryption from the Weil Pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001)
- [4] Borger, R.L.: On De Moivre’s quintic. *The American Mathematical Monthly* 15(10), 171–174 (1908)
- [5] Bosma, W., Cannon, J., Playoust, C.: The Magma Algebra System I: The user language. *J. Symb. Comput.* 24(3/4), 235–265 (1997)
- [6] Brier, E., Coron, J.-S., Icart, T., Madore, D., Randriam, H., Tibouchi, M.: Efficient indiffereniable hashing into ordinary elliptic curves. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 237–254. Springer, Heidelberg (2010), <http://eprint.iacr.org/2009/340/>
- [7] Cardona, G., Quer, J.: Curves of genus 2 with group of automorphisms isomorphic to D_8 or D_{12} . *Trans. Amer. Math. Soc.* 359, 2831–2849 (2007)
- [8] Cox, D.A.: Galois theory. In: *Pure and Applied Mathematics (New York)*. Wiley-Interscience [John Wiley & Sons], Hoboken (2004)
- [9] Farashahi, R.R., Joye, M.: Efficient Arithmetic on Hessian Curves. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 243–260. Springer, Heidelberg (2010)
- [10] Farashahi, R.R., Fouque, P.-A., Shparlinski, I.E., Tibouchi, M., Felipe Voloch, J.: Indifferentiable deterministic hashing to elliptic and hyperelliptic curves. Preprint (2010), <http://www.ma.utexas.edu/users/voloch/Preprints/welldistributed.pdf>
- [11] Farashahi, R.R.: Hashing into hessian curves. *Cryptology ePrint Archive*, Report 2010/373 (2010), <http://eprint.iacr.org/>
- [12] Fouque, P.-A., Tibouchi, M.: Deterministic encoding and hashing to odd hyperelliptic curves. *Cryptology ePrint Archive*, Report 2010/382 (2010), <http://eprint.iacr.org/>
- [13] Icart, T.: How to Hash into Elliptic Curves. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 303–316. Springer, Heidelberg (2009)
- [14] Waterloo Maple Incorporated. Maple. Waterloo, Ontario, Canada, <http://www.maplesoft.com/>
- [15] Sendra, J.R., Winkler, F., Prez-Diaz, S.: *Rational Algebraic Curves: A Computer Algebra Approach*. Springer Publishing Company, Incorporated, Heidelberg (2007)
- [16] Shallue, A., van de Woestijne, C.: Construction of Rational Points on Elliptic Curves over Finite Fields. In: Hess, F., Pauli, S., Pohst, M.E. (eds.) ANTS 2006. LNCS, vol. 4076, pp. 510–524. Springer, Heidelberg (2006)
- [17] Ulas, M.: Rational points on certain hyperelliptic curves over finite fields. *Bull. Polish Acad. Sci. Math.* (55), 97–104 (2007)