Deterministic Encoding and Hashing to Odd Hyperelliptic Curves

Pierre-Alain Fouque, Mehdi Tibouchi

Deterministic Encoding and Hashing to Odd Hyperelliptic Curves

Pierre-Alain Fouque and Mehdi Tibouchi

École normale supérieure
Département d'informatique, Équipe de cryptographie
45 rue d'Ulm, F-75230 Paris CEDEX 05, France
{pierre-alain.fouque,mehdi.tibouchi}@ens.fr

Abstract. In this paper we propose a very simple and efficient encoding function from $\mathbb{F}_q$ to points of a hyperelliptic curve over $\mathbb{F}_q$ of the form $H\colon y^2 = f(x)$ where f is an odd polynomial. Hyperelliptic curves of this type have been frequently considered in the literature to obtain Jacobians of good order and pairing-friendly curves.

Our new encoding is nearly a bijection to the set of $\mathbb{F}_q$-rational points on H. This makes it easy to construct well-behaved hash functions to the Jacobian J of H, as well as injective maps to $J(\mathbb{F}_q)$ which can be used to encode scalars for such applications as ElGamal encryption.

The new encoding is already interesting in the genus 1 case, where it provides a well-behaved encoding to Joux's supersingular elliptic curves.

Keywords: Hyperelliptic Curve Cryptography, Deterministic Encoding, Hashing.

1 Introduction

Hashing into elliptic and hyperelliptic curves. Many cryptosystems based on discrete log-related hardness assumptions, especially in pairing-based cryptography, involve hashing into a group, usually instantiated as the group of points of an elliptic curve or the Jacobian of a hyperelliptic curve. For example in the Boneh-Franklin IBE scheme [4], the public-key for identity $id \in \{0,1\}^*$ is an element $Q_{id} = H_1(id)$ of the group. This is also the case in many other pairing-based cryptosystems including IBE and HIBE schemes [1,15,17], signature and identity-based signature schemes [3,5,6,10,28] and identity-based signcryption schemes [8,23].

Those cryptosystems are proved to be secure when the hash function is modeled as a random oracle into the group, and it is not obvious how to instantiate such a function in practice (when the group is an elliptic curve or a Jacobian) so that the security proof can go through. As discussed in by Brier *et al.* [9], it is sometimes sufficient to use relatively simple constructions that do not behave like random oracles at all, owing to random self-reducibility properties of the underlying problems, but it is generally desirable to have proper hash functions

M. Joye, A. Miyaji, and A. Otsuka (Eds.): Pairing 2010, LNCS 6487, pp. 265–277, 2010.

that can be plugged into any cryptosystem that requires hashing into elliptic and hyperelliptic curves while not compromising proofs of security in the random oracle model.

Deterministic encodings. The basic building block for constructing such hash functions is an encoding from a set that is easy to enumerate, such as $\{0,1\}^n$ or $\mathbb{F}_q$, into the elliptic or hyperelliptic curve group. If the encoding has suitable properties, combining it with a standard hash function may provide a robust construction for hashing into the group.

Generic encodings, such as $t \mapsto t \cdot G$ where G is a group generator, will not work, since they leak the discrete logarithm (as the hash value in the group is usually obtained as from public data, such as the identity in IBE schemes). Thus, the particular form of the group elements intervenes in the encoding.

In the case of elliptic curves, the classical approach is inherently probabilistic: one will first compute an integer hash value $h(m)$ and add a short counter to get $x = 0^{\log k} \| h(m)$. If x is the abscissa of a point on the elliptic curve $y^2 = x^3 + ax + b$, this gives the desired point; otherwise, one increments the counter and tries again. Each step succeeds with probability about $1/2$, so if k is the security parameter, k steps are heuristically enough to construct a point except with negligible probability.

However, the length of the hash computation depends on the message m, which can lead to side-channel attacks [7], unless all k steps are run for all messages, and Legendre symbols and square roots are computed in constant time, in which case computational cost becomes prohibitive. More importantly for pairing-based cryptography, it is difficult to assess the security of a scheme in which such a "probabilistic" hash function is used, even when the underlying integer hash function h is considered ideal.

Therefore, it has been desirable to devise point construction algorithms on elliptic and hyperelliptic curves that are more robust, easier to analyze, and *deterministic*. Algorithms proposed so far fit in two families:

- SWU-like encodings, similar to those proposed by Shallue and van de Woestijne in [26]. They are based on the construction of explicit rational curves on a surface associated to the target curve.
- Icart-like encodings, similar to Icart's function [18]. They are obtained by writing down a root of the curve equation using radicals of degrees prime to the order of the multiplicative group. This is only possible if the curve equation is solvable.

Hyperelliptic curve encodings. While there are now rather general and efficient constructions for elliptic curves (although some important curves remain intractable with current techniques), encodings to hyperelliptic curves are scarce. The first such encoding was proposed by Ulas in [27], for curves of the form $y^2 = x^n + ax + b$ or $y^2 = x^n + ax^2 + bx$. Kammerer, Lercier and Renault, in their recent paper [20], have presented several additional families of hyperelliptic curves for which an Icart-like encoding can be constructed, but the target curves

are still of a special form and may not be convenient to use for cryptographic applications. Efficiency is also a problem for both of these constructions.

Moreover, all of these algorithms construct points on the curve itself, whereas the relevant object in cryptography is the group attached to it, namely its Jacobian variety. Very recently, Farashahi *et al.* [12] have demonstrated how to build a well-behaved hash function to the Jacobian based on a point-construction algorithm to the curve. Their framework apply to the functions proposed by Ulas and Kammerer *et al.*, but with some difficulties and somewhat coarse bounds due to their relatively complex geometric descriptions.

Admissible encodings and indifferentiability. To obtain their well-behaved hash function construction to the Jacobian, Farashahi *et al.* rely on the results by Brier *et al.* [9], which give sufficient conditions for a hash function construction of the form $H(m) = F(h(m))$ to be plugged into any cryptosystem using H as a random oracle provided that h behaves as a random oracle. Basically, Brier *et al.*'s result states that this construction is indistinguishable from a random oracle as soon as F is an *admissible* encoding in the following sense.

A function $F : S \to R$ between finite sets is an admissible encoding if it satisfies the following properties:

1. Computable: F is computable in deterministic polynomial time.
2. Regular: for s uniformly distributed in S, the distribution of $F(s)$ is statistically indistinguishable from the uniform distribution in R.
3. Samplable: there is an efficient randomized algorithm $\mathcal{I}$ such that for any $r \in R$, $\mathcal{I}(r)$ induces a distribution that is statistically indistinguishable from the uniform distribution in $F^{-1}(r)$.

Our contribution. This paper presents a new encoding for hyperelliptic curves of the form $H : y^2 = f(x)$ where f is an odd polynomial over $\mathbb{F}_q$, with $q = 3 \bmod 4$. From this encoding to the curve H, we also deduce efficient injective encodings and well-behaved hash functions to its Jacobian.

The new encoding has the following desirable properties:

- it can be very efficiently computed using one exponentiation and no division, in constant time and without branching;
- the encoding is an efficiently invertible bijection: thus, it is possible to encode messages as points on the curve and recover them. This has numerous applications, e.g. to encryption;
- in genus 1, it provides an encoding to supersingular elliptic curves, similar to Boneh and Franklin's construction [4], but for different base fields;
- in higher genus, many cryptographically interesting curves are of the form H, including the curves considered in [14,16,25];
- many constructions of pairing-friendly hyperelliptic curves yield curves of the form H [21,13];
- since the encoding has a simple geometric description, it is easy to obtain well-behaved hash functions from it, and the corresponding regularity bounds are optimally tight.

2 Odd Hyperelliptic Curves

Let f be an odd monic polynomial over a finite field $\mathbb{F}_q$ with $q \equiv 3 \pmod 4$, which has simple roots in $\overline{\mathbb{F}_q}$. We denote its degree by $2g+1$, and consider the hyperelliptic curve over $\mathbb{F}_q$ defined by:

$$H\colon y^2 = f(x) = x^{2g+1} + a_1 x^{2g-1} + \cdots + a_g x$$

Let us call such curves *odd hyperelliptic curves*. Many hyperelliptic curves relevant to cryptography, and particularly pairing-based cryptography, are of this form. For example:

- the supersingular elliptic curves of Joux [19]: $y^2 = x^3 + ax$;
- the genus 2 curves studied by Furukawa *et al.* [14] and their extension to genus g by Haneda *et al.* [16]: $y^2 = x^{2g+1} + ax$ (for which one can compute the zeta function);
- in particular, the Type II pairing-friendly curves of genus 2 constructed by Kawazoe and Takahashi [21];
- the genus 2 hyperelliptic curves for which Satoh [25] gave an efficient class group counting algorithm: $y^2 = x^5 + ax^3 + bx$;
- in particular, some of the pairing-friendly genus 2 curves constructed by Freeman and Satoh [13] (although the case $q \equiv 1 \pmod 4$ is more common).

Additionally, odd hyperelliptic curves and their Jacobians admit an automorphism of order 4 over $\mathbb{F}_{q^2}$ (namely $(x, y) \mapsto (-x, \sqrt{-1} \cdot y)$) which can be used to map points over $\mathbb{F}_q$ to linearly independent points over $\mathbb{F}_{q^2}$, another useful property for pairings.

Remark 1. A hyperelliptic curve over $\mathbb{F}_q$ is birational to an odd hyperelliptic curve when the set of points in $\mathbb{P}^1$ over which it is ramified is invariant under an automorphism of $\mathbb{P}^1$ of order 2 fixing two of them, both $\mathbb{F}_q$-rational. For example, hyperelliptic curves of the form:

$$H'\colon y^2 = x^6 + ax^5 + bx^4 - bx^2 - ax - 1$$

are birational to odd hyperelliptic curves, since they are ramified over a set of points invariant under $x \mapsto 1/x$ and containing ± 1. One possible change of variables is $x \mapsto (x-1)/(x+1)$.

This remark shows that the coarse moduli space of odd hyperelliptic curves of genus g over $\overline{\mathbb{F}_q}$ is a subvariety of dimension $g-1$ of the dimension $2g-1$ moduli space of genus g hyperelliptic curves.

3 Our New Encoding

3.1 Definition

Let $H\colon y^2 = f(x)$ be an odd hyperelliptic curve over $\mathbb{F}_q$. Denote by $\sqrt{\cdot}$ the usual square root function on the set of quadratic residues in $\mathbb{F}_q$ (exponentiation by $(q+1)/4$), and by $\left(\frac{\cdot}{q}\right)$ the Legendre symbol over $\mathbb{F}_q$.

Over $\mathbb{F}_q$, -1 is a quadratic nonresidue, and for any $t \in \mathbb{F}_q$, we have $f(-t) = -f(t)$, so unless $f(t) = 0$, exactly one of $f(t)$ or $f(-t)$ is a square. In other words, exactly one of t or $-t$ is the abscissa of an $\mathbb{F}_q$-rational point on H.

This observation allows us to define a point encoding function F to $H(\mathbb{F}_q)$ as follows:

$$\begin{aligned} F\colon \mathbb{F}_q &\longrightarrow H(\mathbb{F}_q) \\ t &\longmapsto \left(\varepsilon(t)\cdot t\ ;\ \varepsilon(t)\sqrt{\varepsilon(t)\cdot f(t)}\right) \end{aligned} \tag{1}$$

where $\varepsilon(t) = \left(\frac{f(t)}{q}\right)$. We claim that this function is well-defined and "almost" a bijection.

More precisely, recall that a *Weierstrass point* of H is a point where the rational function y is ramified: these are the points $(x, 0)$ for x a root of f together with the point at infinity ∞. Then, let $W \subset H(\mathbb{F}_q)$ be the set of $\mathbb{F}_q$-rational Weierstrass points on H, and $T \subset \mathbb{F}_q$ the set of roots of f.

Theorem 1. *The function F given by* (1) *is well-defined, maps all points in T to $(0,0) \in W$, and induces a bijection $\mathbb{F}_q \setminus T \to H(\mathbb{F}_q) \setminus W$.*

Proof. For $t \in T$, we have $\varepsilon(t) = 0$, hence $F(t) = (0,0) \in W$. Now let $t \in \mathbb{F}_q \setminus T$, and $x = \varepsilon(t)\cdot t$. Since f is odd and $\varepsilon(t) = \pm 1$, $f(x) = \varepsilon(t)\cdot f(t)$. In particular, recalling that $\left(\frac{-1}{q}\right) = -1$, we can write:

$$\left(\frac{f(x)}{q}\right) = \left(\frac{\varepsilon(t)\cdot f(t)}{q}\right) = \varepsilon(t)\cdot\left(\frac{f(t)}{q}\right) = \varepsilon(t)^2 = 1$$

Thus, the second component $y = \varepsilon(t)\sqrt{\varepsilon(t)\cdot f(t)}$ of $F(t)$ is well-defined, and we have $y^2 = \varepsilon(t)\cdot f(t) = f(x)$, so $F(t)$ is an affine point on $H(\mathbb{F}_q)$ as required. The condition $t \notin T$ further implies that $f(t) \neq 0$, so $y \neq 0$. Therefore, $F(t) \in \mathbb{F}_q \setminus W$.

Let us show that the restriction of F to $\mathbb{F}_q \setminus T$ is injective. Indeed, suppose $F(t) = F(u)$ with $t, u \notin T$. Equating x-coordinates, we get $\varepsilon(t)\cdot t = \varepsilon(u)\cdot u$, hence $u = \pm t$. If $u = -t$, then comparing the y-coordinates, we obtain

$$\begin{aligned} \varepsilon(t)\sqrt{\varepsilon(t)\cdot f(t)} &= \varepsilon(u)\sqrt{\varepsilon(u)\cdot f(u)} \\ &= \varepsilon(-t)\sqrt{\varepsilon(-t)\cdot f(-t)} = -\varepsilon(t)\sqrt{\varepsilon(t)\cdot f(t)} \end{aligned}$$

which is a condraction. Therefore, $t = u$ and F is injective on $\mathbb{F}_q \setminus T$.

Finally, $F(\mathbb{F}_q \setminus T) = H(\mathbb{F}_q) \setminus W$. To see this, take $(x, y) \in H(\mathbb{F}_q) \setminus W$ and let $t = \delta\cdot x$, where $\delta = \pm 1$ is defined by $y = \delta\sqrt{f(x)}$. We have

$$\varepsilon(t) = \left(\frac{f(\delta x)}{q}\right) = \left(\frac{\delta\cdot f(x)}{q}\right) = \delta\cdot\left(\frac{f(x)}{q}\right) = \delta$$

since $f(x) = y^2$ is a square. Thus:

$$F(t) = \left(\delta^2\cdot x\ ;\ \delta\sqrt{\delta\cdot f(\delta x)}\right) = \left(x\ ;\ \delta\sqrt{f(x)}\right) = (x; y)$$

as required. □

Corollary 1. *The cardinal of $H(\mathbb{F}_q)$ is $q+1$.*

Proof. From the above, we get $\#H(\mathbb{F}_q) = \#(\mathbb{F}_q \setminus T) + \#W = q - \#T + \#W$. But W consists of the point at infinity on H, and all points of the form $(x, 0)$, $x \in T$. Thus, $\#W = \#T + 1$, and $\#H(\mathbb{F}_q) = q + 1$. □

Remark 2. – Since F is an efficiently computable bijection between all of $\mathbb{F}_q$ and $H(\mathbb{F}_q)$ except at most $2g+2$ points on both sides, with an efficiently computable inverse (namely $(x, y) \mapsto \left(\frac{y}{q}\right) x$), it is a very well-behaved encoding function.

In particular, it is clear that if t is uniformly distributed in $\mathbb{F}_q$, the distribution of $F(t)$ in $H(\mathbb{F}_q)$ is statistically indistinguishable from the uniform distribution. According to the results of Brier *et al.* [9], it follows that if $m \mapsto h(m)$ is a hash function to $\mathbb{F}_q$ modeled as a random oracle, then $F(h(m))$ is a function into $H(\mathbb{F}_q)$ that is indifferentiable from a random oracle. When the genus of H is at least 2, however, one is usually interested in hashing to the Jacobian of H rather than H itself. This will be discussed in §4.

The fact that F is injective, unlike most other constructions, makes it possible to also use it for other purposes than hashing, such as encoding a message to be encrypted, for example with ElGamal.

– Since $\#T = \#(W \setminus \{\infty\})$, it is in fact easy to modify the definition of F to obtain a bijection $F' \colon \mathbb{F}_q \to H(\mathbb{F}_q) \setminus \{\infty\}$ which misses only one rational point on H. It is slightly less efficient to compute, however, and using one or the other makes no difference in practice (as one is not concerned with a few exceptional points), so we shall stick to F as defined by (1).

– When H is in fact an elliptic curve E (i.e. $\deg f = 3$), Corollary 1 says that E is supersingular. These are in fact the supersingular elliptic curves $y^2 = x^3 + ax$ discussed by Joux in [19]. Thus, the function F provides a convenient way to encode points into supersingular elliptic curves over $\mathbb{F}_q$ with $q \equiv 3 \pmod 4$. This is an interesting addition to the original encoding of Boneh and Franklin [4], which applies to supersingular curves of the form $y^2 = x^3 + b$ over fields $\mathbb{F}_q$ with $q \equiv 2 \pmod 3$. In particular, our encoding can be used in characteristic 3.

– In the general case, we see that $\#H(\mathbb{F}_{q^n}) = q^n + 1$ for any odd extension degree n. This gives some constraints on the zeta function of H, but in genus $g \geq 2$, many isogeny classes are possible for the Jacobian J of H nonetheless, so the proposed encoding applies to a wide range of curves. It is not always easy to determine the order of $J(\mathbb{F}_q)$: an approach is given by Satoh in [25] for $g = 2$.

3.2 Efficient Computation

The definition of F involves a generalized Legendre symbol and one square root, which suggests that its computation might be costly, especially if it is to be done in constant time, an important property in settings where side-channel

attacks are a concern. However, it is actually possible to compute F with a single exponentiation, a few multiplications and no division, making it one of the most efficient deterministic encoding function proposed to date. One such implementation is described as Algorithm 1. Note that this implementation is also branch-free, contrary to what happens for encodings such as the one by Shallue and van de Woestijne [26]; this also prevents certain active side-channel attacks.

Algorithm 1. Constant-time, single-exponentiation implementation of the encoding F. The constant r is $(q-3)/4$ if $q \equiv 3 \pmod 8$, $(q-3)/4 + (q-1)/2$ otherwise.

```
1: function F(t)
2:     α ← f(t)
3:     β ← α^r
4:     return (αβ²t, αβ)
5: end function
```

To see that this implementation is correct, consider α and β as defined in Algorithm 1. For $t \in T$, we have $\alpha = 0$, hence the procedure returns $F(t) = (0, 0)$ as required. Now let $t \notin T$. We have

$$\beta^2 = \alpha^{\frac{q-3}{2}} = \frac{1}{\alpha}\left(\frac{\alpha}{q}\right) = \frac{\varepsilon(t)}{\alpha}$$

In particular, $\alpha\beta^2 t = \varepsilon(t) \cdot t$ is indeed the abscissa of $F(t)$.

Moreover, suppose $q \equiv 3 \pmod 8$. Then $(q+1)/4$ is odd and $\varepsilon(t) = \pm 1$, so we have

$$\begin{aligned}\alpha\beta &= \alpha^{\frac{q-3}{4}+1} = \varepsilon(t) \cdot \varepsilon(t) \cdot f(t)^{\frac{q+1}{4}} \\ &= \varepsilon(t) \cdot \big(\varepsilon(t) \cdot f(t)\big)^{\frac{q+1}{4}} = \varepsilon(t)\sqrt{\varepsilon(t) \cdot f(t)}\end{aligned}$$

so the algorithm is correct.

Similarly, when $q \equiv 7 \pmod 8$, $(q+1)/4$ is odd and we obtain

$$\begin{aligned}\alpha\beta &= \alpha^{\frac{q-1}{2}+\frac{q-3}{4}+1} = \varepsilon(t) \cdot f(t)^{\frac{q+1}{4}} \\ &= \varepsilon(t) \cdot \big(\varepsilon(t) \cdot f(t)\big)^{\frac{q+1}{4}} = \varepsilon(t)\sqrt{\varepsilon(t) \cdot f(t)}\end{aligned}$$

which concludes.

4 Mapping to the Jacobian

In the previous section, we have constructed a function $F\colon \mathbb{F}_q \to H(\mathbb{F}_q)$ which is efficiently computable and has a number of desirable properties. For cryptographic purposes, however, we are usually interested in obtaining elements of

a group attached to the curve, namely the Jacobian, rather than points on the curve itself. In the case of elliptic curves, the curve and its Jacobian are isomorphic so no further work is needed, but for curves of genus $g \geq 2$, they are quite different objects.

In the following, we always denote the Jacobian of H by J, and we regard H as embedded in J via the map $H \to J$ sending a point P to the class of the degree 0 divisor $(P) - (\infty)$. In particular, if P, Q are points in $H(\mathbb{F}_q)$, $P + Q$ denotes the class of $(P) + (Q) - 2(\infty)$.

We propose two constructions of maps to $J(\mathbb{F}_q)$ to accommodate for different use cases: an injective map with large image, which can be used to encode scalars as group elements (e.g. for encryption), and a map defining an essentially uniform distribution on $J(\mathbb{F}_q)$, to obtain well-behaved hash functions.

4.1 Injective Encoding to the Jacobian

Let us first recall a few facts about hyperelliptic curves, for which we refer for example to [24]. Elements of $J(\mathbb{F}_q)$ are classes of $\mathbb{F}_q$-divisors on H and admit a canonical representation as so-called *reduced divisors* defined over $\mathbb{F}_q$. Let $\widetilde{\cdot}$ denote the hyperelliptic involution on H, $(x, y) \mapsto (x, -y)$. A divisor $D = P_1 + \cdots + P_r$ (where the P_i are not necessarily distinct points in $H(\overline{\mathbb{F}_q})$) is said to be reduced when r is less than or equal to the genus g of H, and $P_i \neq \widetilde{P_j}$ for all $i \neq j$. The reduced divisors D and D' defined by $P_1, \ldots, P_r$ and $P'_1, \ldots, P'_r$ are distinct and non-equivalent as soon as the multisets $\{P_1, \ldots, P_r\}$ and $\{P'_1, \ldots, P'_r\}$ are different. Each divisor class in $J(\mathbb{F}_q)$ contains a unique reduced divisor defined over $\mathbb{F}_q$.

Now, with the notations of §3, the encoding $F \colon \mathbb{F}_q \to H(\mathbb{F}_q)$ defined by (1) satisfies that for all $t \in \mathbb{F}_q \setminus T$, the only u such that $F(u) = \widetilde{F(t)}$ is $u = -t$. Therefore, if $(t_1, \ldots, t_g)$ is any tuple of g elements of $\mathbb{F}_q \setminus T$ (g being the genus of H) such that $t_i + t_j \neq 0$ for all i, j, then $F(t_1) + \cdots + F(t_g)$ is a reduced divisor. In particular, consider the set X of g-element subsets of $\mathbb{F}_q \setminus T$ not containing any two opposite elements. Then it is immediate from the facts above that the map:

$$\begin{aligned} F_{\mathrm{inj}} \colon X &\longrightarrow J(\mathbb{F}_q) \\ \{t_1, \ldots, t_g\} &\longmapsto F(t_1) + \cdots + F(t_g) \end{aligned}$$

is injective. We have

$$\#X = 2^g \binom{(q - \#T)/2}{g} = \frac{1 - o(1)}{g!} \cdot q^g \geq c_g \cdot \#J(\mathbb{F}_q)$$

for some constant $c_g > 0$ depending only on g. Thus, F_{inj} is an injective mapping to $J(\mathbb{F}_q)$ covering a large portion of all points. It is also very easy to compute since points in the image are directly given as reduced divisors, so no actual arithmetic on the Jacobian is needed.

In the case that is most relevant for cryptographic applications, namely $g = 2$, we can define an even simpler injective encoding, from the set Y of 2-element subsets of $\mathbb{F}_q \setminus T$, which may be easier to manipulate than X:

$$\begin{aligned} F'_{\text{inj}} \colon Y &\longrightarrow J(\mathbb{F}_q) \\ \{t_1, t_2\} &\longmapsto F(t_1) + F(-t_2) \end{aligned}$$

This function injective, easy to compute, and reaches roughly one half of all points in $J(\mathbb{F}_q)$.

4.2 Indifferentiable Hashing to the Jacobian

One can also use F to construct well-behaved hash functions to $J(\mathbb{F}_q)$. For this purpose, Brier *et al.* [9] have shown how one could use functions to $J(\mathbb{F}_q)$ with good regularity properties, and Farashahi *et al.* [12] have proposed a framework based on character sums to prove such regularity properties for functions of the form:

$$\begin{aligned} F^{\otimes s} \colon (\mathbb{F}_q)^s &\longrightarrow J(\mathbb{F}_q) \\ (t_1, \ldots, t_s) &\longmapsto F(t_1) + \cdots + F(t_s) \end{aligned}$$

Since F is so simple, we do not really need to rely on the entire framework of [12]. Indeed, the following bound, which in the terminology of Farashahi *et al.* says that F is a $(2g - 2 + \varepsilon)$-well-distributed encoding, can be proved using classical results on characters on algebraic curves. Note that this bound is very tight: it gives a better well-distributedness bound for F in genus up to 6 than can be established for Icart's function in genus 1.

Lemma 1. *For any character χ of the abelian group $J(\mathbb{F}_q)$, let*

$$S(\chi) = \sum_{t \in \mathbb{F}_q} \chi(F(t))$$

Then, whenever χ is nontrivial, we have

$$|S(\chi)| \leq (2g - 2)\sqrt{q} + 4g + 3$$

Proof. A nontrivial character χ of $J(\mathbb{F}_q)$ is also a nontrivial, unramified Artin character of H (see [22, §2] or [12, §4]). In particular, the Riemann hypothesis for the L-function on H associated with χ gives:

$$\left| \sum_{P \in H(\mathbb{F}_q)} \chi(P) \right| \leq (2g - 2)\sqrt{q}$$

The result then follows from the observation that

$$\begin{aligned} \sum_{t \in \mathbb{F}_q} \chi(F(t)) &= \#T \cdot \chi((0,0)) + \sum_{P \in H(\mathbb{F}_q) \setminus W} \chi(P) \\ &= \#T \cdot \chi((0,0)) - \sum_{P \in W} \chi(P) + \sum_{P \in H(\mathbb{F}_q)} \chi(P) \end{aligned}$$

since $\#T + \#W \leq 4g + 3$. □

We can then proceed like in [12] and deduce from this lemma a bound on the statistical distance between the distribution defined on $J(\mathbb{F}_q)$ by $F^{\otimes s}$ and the uniform distribution.

For any $D \in J(\mathbb{F}_q)$, let $N_s(D)$ denote the number of preimages of D under $F^{\otimes s}$:

$$N_s(D) = \#\big\{(t_1, \ldots, t_s) \in (\mathbb{F}_q)^s \mid D = F(t_1) + \cdots + F(t_s)\big\}$$

Then we have the following result:

Theorem 2. *The statistical distance between the distribution defined by $F^{\otimes s}$ and the uniform distribution on $J(\mathbb{F}_q)$ is bounded as:*

$$\sum_{D \in J(\mathbb{F}_q)} \left| \frac{N_s(D)}{q^s} - \frac{1}{\#J(\mathbb{F}_q)} \right| \leq \frac{\big(2g + 2 + (4g+3)q^{-1/2}\big)^s \sqrt{\#J(\mathbb{F}_q)}}{q^{s/2}}$$

Proof. This results from [12, Theorem 2]. We can give a quick recap of the proof for the reader's convenience.

Note first that one can write $N_s(D)$ in terms of the character sums $S(\chi)$ as follows:

$$\begin{aligned} N_s(D) &= \sum_{t_1,\ldots,t_s \in \mathbb{F}_q} \frac{1}{\#J(\mathbb{F}_q)} \sum_{\chi} \chi\left(F(t_1) + \cdots + F(t_s) - D\right) \\ &= \sum_{\chi} \frac{\chi(-D)}{\#J(\mathbb{F}_q)} \sum_{t_1,\ldots,t_s \in \mathbb{F}_q} \chi\left(F(t_1) + \cdots + F(t_s)\right) \\ &= \sum_{\chi} \frac{\chi(-D)}{\#J(\mathbb{F}_q)} S(\chi)^s \end{aligned}$$

Putting the trivial character aside, this yields:

$$\frac{N_s(D)}{q^s} - \frac{1}{\#J(\mathbb{F}_q)} = \frac{\chi(-D)}{q^s \#J(\mathbb{F}_q)} \sum_{\chi \neq 1} S(\chi)^s$$

Then, we consider the sum of squares of this expression as D varies along $J(\mathbb{F}_q)$. Let

$$V_s = \sum_{D \in J(\mathbb{F}_q)} \left| \frac{N_s(D)}{q^s} - \frac{1}{\#J(\mathbb{F}_q)} \right|^2$$

We have

$$\begin{aligned} V_s &= \sum_{D} \frac{1}{q^{2s} \#J(\mathbb{F}_q)^2} \sum_{\chi, \chi' \neq 1} \chi(-D)\overline{\chi'}(-D) \cdot S(\chi)^s \cdot \overline{S(\chi')^s} \\ &= \frac{1}{q^{2s} \#J(\mathbb{F}_q)^2} \sum_{\chi, \chi' \neq 1} \left(\sum_{D} \chi(D)\overline{\chi'}(D) \right) S(\chi)^s \cdot \overline{S(\chi')^s} \\ &= \frac{1}{q^{2s} \#J(\mathbb{F}_q)} \sum_{\chi \neq 1} \big|S(\chi)\big|^{2s} \leq \frac{\big((2g+2)\sqrt{q} + 4g + 3\big)^{2s}}{q^{2s}} \end{aligned}$$

since the sum over D of $\chi(D)\overline{\chi'}(D)$ vanishes if $\chi \neq \chi'$. Finally, the Cauchy-Schwarz inequality gives:

$$\sum_{D \in J(\mathbb{F}_q)} \left| \frac{N_s(D)}{q^s} - \frac{1}{\#J(\mathbb{F}_q)} \right| \leq \sqrt{V_s} \cdot \sqrt{\#J(\mathbb{F}_q)}$$
$$\leq \frac{\left(2g + 2 + (4g+3)q^{-1/2}\right)^s \sqrt{\#J(\mathbb{F}_q)}}{q^{s/2}}$$

as required. □

Note that $\#J(\mathbb{F}_q) \sim q^g$, so that the bound we get on the statistical distance is in $O(q^{(g-s)/2})$. Therefore, as soon as $s > g$, the distribution defined by $F^{\otimes s}$ on $J(\mathbb{F}_q)$ is statistically indistinguishable from the uniform distribution. In particular, in the terminology of Brier *et al.* [9] which we recalled in the introduction, the encoding $F^{\otimes(g+1)}$ to $J(\mathbb{F}_q)$ is regular. It is also obviously computable and samplable, so $F^{\otimes(g+1)}$ is an admissible encoding to $J(\mathbb{F}_q)$.

This provides a simple, well-behaved hash function construction to the Jacobian of H. Indeed, it follows that the function

$$m \mapsto F(h_1(m)) + \cdots + F(h_{g+1}(m))$$

is indifferentiable from a random oracle if $h_1, \ldots, h_{g+1}$ are seen as independent random oracles into $\mathbb{F}_q$.

5 Conclusion

In this paper, we provide a very efficient construction of a deterministic encoding into odd hyperelliptic curves. Odd hyperelliptic curves are a simple and relatively large class of hyperelliptic curves, compared to the families of curves covered by previous deterministic encodings. They also include many curves of cryptographic interest (because of efficient point-counting on the Jacobian, or pairing-friendliness), even in the elliptic curve case.

This encoding is almost a bijection, which can be useful for a number of applications, such as encryption, and allows us to construct the first efficient injections with large image to the Jacobians of odd hyperelliptic curves, as well as indifferentiable hash functions to these Jacobians with particularly tight regularity bounds.

Acknowledgments. We are grateful to Reza Farashahi and anonymous referees for useful comments, and to Masayuki Abe, Jean-Sébastien Coron and Thomas Icart for earlier discussions that inspired this paper. This work was partly supported by the French ANR-07-TCOM-013-04 PACE Project and by the European Commission through the IST Program under Contract ICT-2007-216646 ECRYPT II.

References

1. Baek, J., Zheng, Y.: Identity-based threshold decryption. In: Bao, et al. (eds.) [2], pp. 262–276
2. Bao, F., Deng, R., Zhou, J. (eds.): PKC 2004. LNCS, vol. 2947. Springer, Heidelberg (2004)
3. Boldyreva, A.: Threshold signatures, multisignatures and blind signatures based on the gap-diffie-hellman-group signature scheme. In: Desmedt (ed.) [11], pp. 31–46
4. Boneh, D., Franklin, M.K.: Identity-based encryption from the weil pairing. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 213–229. Springer, Heidelberg (2001)
5. Boneh, D., Gentry, C., Lynn, B., Shacham, H.: Aggregate and verifiably encrypted signatures from bilinear maps. In: EUROCRYPT, pp. 416–432 (2003)
6. Boneh, D., Lynn, B., Shacham, H.: Short signatures from the weil pairing. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 514–532. Springer, Heidelberg (2001)
7. Boyd, C., Montague, P., Nguyen, K.Q.: Elliptic curve based password authenticated key exchange protocols. In: Varadharajan, V., Mu, Y. (eds.) ACISP 2001. LNCS, vol. 2119, pp. 487–501. Springer, Heidelberg (2001)
8. Boyen, X.: Multipurpose identity-based signcryption (a swiss army knife for identity-based cryptography). In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 383–399. Springer, Heidelberg (2003)
9. Brier, E., Coron, J.-S., Icart, T., Madore, D., Randriam, H., Tibouchi, M.: Efficient indifferentiable hashing into ordinary elliptic curves. In: Rabin, T. (ed.) CRYPTO 2010. LNCS, vol. 6223, pp. 237–254. Springer, Heidelberg (2010)
10. Cha, J.C., Cheon, J.H.: An identity-based signature from gap diffie-hellman groups. In: Desmedt (ed.) [11], pp. 18–30
11. Desmedt, Y. (ed.): PKC 2003. LNCS, vol. 2567. Springer, Heidelberg (2002)
12. Farashahi, R.R., Fouque, P.-A., Shparlinski, I., Tibouchi, M., Voloch, F.: Indifferentiable deterministic hashing to elliptic and hyperelliptic curves. Preprint (2010), `http://www.di.ens.fr/~tibouchi/research.html`
13. Freeman, D.M., Satoh, T.: Constructing pairing-friendly hyperelliptic curves using weil restriction. Cryptology ePrint Archive, Report 2009/103 (2009), `http://eprint.iacr.org/`
14. Furukawa, E., Kawazoe, M., Takahashi, T.: Counting points for hyperelliptic curves of type $y^2 = x^5 + ax$ over finite prime fields. In: Matsui, M., Zuccherato, R.J. (eds.) SAC 2003. LNCS, vol. 3006, pp. 26–41. Springer, Heidelberg (2003)
15. Gentry, C., Silverberg, A.: Hierarchical id-based cryptography. In: Zheng (ed.) [29], pp. 548–566
16. Haneda, M., Kawazoe, M., Takahashi, T.: Suitable curves for genus-4 HCC over prime fields. In: Caires, L., Italiano, G.F., Monteiro, L., Palamidessi, C., Yung, M. (eds.) ICALP 2005. LNCS, vol. 3580, pp. 539–550. Springer, Heidelberg (2005)
17. Horwitz, J., Lynn, B.: Toward hierarchical identity-based encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 466–481. Springer, Heidelberg (2002)
18. Icart, T.: How to hash into elliptic curves. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 303–316. Springer, Heidelberg (2009)
19. Joux, A.: The weil and tate pairings as building blocks for public key cryptosystems. In: Fieker, C., Kohel, D.R. (eds.) ANTS 2002. LNCS, vol. 2369, pp. 20–32. Springer, Heidelberg (2002)

20. Kammerer, J.-G., Lercier, R., Renault, G.: Encoding points on hyperelliptic curves over finite fields in deterministic polynomial time. CoRR, abs/1005.1454 (2010)
21. Kawazoe, M., Takahashi, T.: Pairing-friendly hyperelliptic curves with ordinary jacobians of type $y^2 = x^5 + ax$. In: Galbraith, S.D., Paterson, K.G. (eds.) Pairing 2008. LNCS, vol. 5209, pp. 164–177. Springer, Heidelberg (2008)
22. Kohel, D.R., Shparlinski, I.: On exponential sums and group generators for elliptic curves over finite fields. In: Bosma, W. (ed.) ANTS 2000. LNCS, vol. 1838, pp. 395–404. Springer, Heidelberg (2000)
23. Libert, B., Quisquater, J.-J.: Efficient signcryption with key privacy from gap diffie-hellman groups. In: Bao, et al. (eds.) [2], pp. 187–200
24. Menezes, A.J., Wu, Y.-H., Zuccherato, R.J.: An elementary introduction to hyperelliptic curves. In: Koblitz, N. (ed.) Algebraic Aspects of Cryptography. Algorithms and Computation in Mathematics, vol. 3, pp. 155–178. Springer, Heidelberg (1998)
25. Satoh, T.: Generating genus two hyperelliptic curves over large characteristic finite fields. In: Joux, A. (ed.) EUROCRYPT 2009. LNCS, vol. 5479, pp. 536–553. Springer, Heidelberg (2010)
26. 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)
27. Ulas, M.: Rational points on certain hyperelliptic curves over finite fields. Bull. Polish Acad. Sci. Math. 55(2), 97–104 (2007)
28. Zhang, F., Kim, K.: Id-based blind signature and ring signature from pairings. In: Zheng (ed.) [29], pp. 533–547
29. Zheng, Y. (ed.): ASIACRYPT 2002. LNCS, vol. 2501. Springer, Heidelberg (2002)