FACTA UNIVERSITATIS (NIŠ)
Ser. Math. Inform. Vol. 34, No 5 (2019), 889–902
https://doi.org/10.22190/FUMI1905889Y

EMBEDDING FINITE FIELDS INTO ELLIPTIC CURVES *

Amirmehdi Yazdani Kashani and Hassan Daghigh

Abstract. Many elliptic curve cryptosystems require an encoding function from a finite field $\mathbb{F}_q$ into $\mathbb{F}_q$-rational points of an elliptic curve. We propose a uniform encoding to general elliptic curves over $\mathbb{F}_q$. We also discuss about an injective case of SWU encoing for hyperelliptic curves of genus 2. Moreover we discuss about an injective encoding for elliptic curves with a point of order two over a finite field and present a description for these elliptic curves.

Keywords: elliptic curve; cryptosystem; encoding function; finite field.

1. Introduction

Throughout this article p is a prime number and $q = p^n$ for some $n \in \mathbb{N}$.

Since 1987, when the elliptic curves cryptography was introduced by Koblitz [12], encoding efficiently and deterministically a message by a point on an elliptic curve E has been, and still is, an important question. Several methods have been proposed to solve this problem. Perhaps before 2006 the most common method was the try and increment method . The method consists in taking $x \in \mathbb{F}_q$ and checking whether this value can be abscissa of a point on E. If not, we increment x by 1 until the new value is abscissa of a point on E. The main problem with this algorithm is that the number of steps depends on the input x. The twisted curves method was to apply curve and its twist as suggested in [5]. If E is defined by $y^2{=}x^3{+}ax{+}b$ over $\mathbb{F}_q$, the twist of E is a curve E^d defined by

$$dy^2{=}x^3{+}ax{+}b,$$

where d is a quadratic non-residue in $\mathbb{F}_q$. Then for every x there exists y such that the point (x, y) belongs to E or E^d. The method received little attention, since

Received December 31, 2018; accepted September 04, 2019
2010 *Mathematics Subject Classification.* Primary 14H52; Secondary 11T71
*The authors were supported in part by a grant from University of Kashan (808718/1)

it requires calculation on two curves and this doubles the running time. When $q \equiv 2 \ (mod\ 3)$ the map $x \to x^3$ is a bijection from $\mathbb{F}_q^*$ to itself. If E is defined by the equation $y^2 = x^3 + b$, the map

$$f : u \longrightarrow ((u^2 - b)^{\frac{1}{3}}, u)$$

is a bijection from $\mathbb{F}_q$ to affine points on the curve E. Therefore these curves are supersingular for every b. The MOV attack gives an efficient computable method which enables to reduce the DLP on a supersingular elliptic curve to DLP on a finite field [13]. Therefore in order to avoid this attack, much larger parameters must be used.

In 2006 the first algorithm for encoding to elliptic curves in deterministic polynomial time was proposed by Shallue and Woestijne [14]. The algorithm is based on the Skalba equality which says that there exist four maps $X_1(t), X_2(t), X_3(t), X_4(t)$ such that

$$f(X_1(t))f(X_2(t))f(X_3(t)) = (X_4(t))^2,$$

where $f(x) = X^3 + aX + b$. Then in a finite field for a fixed parameter t, there exists $1 \leqslant j \leqslant 3$ such that $f(X_j(t))$ is a quadratic residue. This implies that $(X_j(t), \sqrt{f(X_j(t))})$ is a point on $E : y^2 = f(x)$. For $q \equiv 3(mod\ 4)$ computing the square root $\sqrt{f(X_j(t))}$ is simply an exponentiation but for $q \equiv 1(mod\ 4)$, no deterministic algorithm has been found for computing the square root. If we have a non quadratic residue in $\mathbb{F}_q$ we can apply Tonelli Shanks algorithm to compute the square root. Using Skalba equality the authors of [14] show that a modification of Tonelli-Shanks algorithm can compute square roots deterministicaly in time $O(\log^4 q)$. Shallue-Woestijne method runs in time $O(\log^4 q)$ for any field size $q = p^n$ and in time $O(\log^3 q)$ when $q \equiv 3 \ (mod\ 4)$.Ulas reduced and generalized maps to hyperelliptic curves in 2007 [16].

In 2009 Icart proposed another method for encoding to elliptic curves [11]. If $q \equiv 2 \ (mod\ 3)$ the map $x \to x^3$ is a bijection in $\mathbb{F}_q$ and cube roots are uniquely defined with $x^{\frac{1}{3}} = x^{\frac{2q-1}{3}}$.Icart defined an encoding as follows:

$$f_{a,b}{:}\mathbb{F}_{p^n} \longrightarrow E_{a,b}$$

$$u \longrightarrow (x, y),$$

where

$$x{=}(v^2{-}b{-}\frac{u^6}{27})^{\frac{1}{3}} + \frac{u^2}{3}, \qquad y{=}ux{+}v, \qquad v{=}\frac{3a{-}u^4}{6u}.$$

He fixed $f_{a,b}(0) = O$, the neutral element of the elliptic curve. Icart proved that for all $p \in E_{a,b}$ the set $f^{-1}(p)$ is computable in polynomial time and $\left|f^{-1}(p)\right| \leqslant 4$, namely a point has at most 4 preimages. He also proved that his algorithm works with complexity $O\left(\log^3 q\right)$ and conjectured that the image of $f_{a,b}$ contains $\frac{5}{8}.\#E(\mathbb{F}_q){+}O(q^{\frac{1}{2}})$. Icart's conjecture was proved by Farashahi, Shparlinski and Voloch[8].

Brier et al [4] designed a further simplification of Shallue-Woestijne-Ulas algorithm for elliptic curve over $\mathbb{F}_q$ with $q \equiv 3(mod\ 4)$. They showed that a point has at most 8 preimages which can be efficiently computed.

For cryptographic purposes it is important to have an injective encoding into an elliptic curve. Recently Fouque, Jeux and Tibouchi [9] proposed an injective encoding to elliptic curves of the form

$$E_c^\delta : y^2 = x^3 - 4\delta x^2 + \delta(c + \delta/c)^2 x,$$

where $c \in \mathbb{F}_q \backslash \{-1, 0, 1\}, \delta = \pm 1$. Bernstein, Hamburg, Krosnova and Lange [2] proposed an injective encoding for elliptic curves of the form

$$E_{a,b} : y^2 = x(x^2 + ax + b)$$

with $a, b \in \mathbb{F}_q$. We present a description for these elliptic curves over a finite field. we also present a uniform encoding for general elliptic curves over $\mathbb{F}_q$. Finally we discuss about SWU encoding for genus 2 hyperelliptic curves over $\mathbb{F}_q$.

2. Injective encoding

Injective encoding from finite field elements into the points of an elliptic curve is a more challenging problem and needs to be studied more carefully. In this section we express a method for encoding into elliptic curves of the form $E_{a,b}$ over finite fields.

Lemma 2.1. *Let $g(x) = x(x^2 + ax + b)$ where $a \neq 0$ and u be a quadratic non residue in $\mathbb{F}_q$.*

(i)If $x \in \mathbb{F}_q$ satisfies:

$$g(ux) = u^3 g(x) \tag{2.1}$$

then one and only one of the two values x or ux is the abscissa of a point on the curve $y^2 = g(x)$.

(ii)The only solution of (2.1) is

$$x = \frac{-b}{a}(\frac{1+u}{u})$$

Proof. Since u is a quadratic non residue, the equation (2.1) implies that one and only one of the values $g(x)$ or $g(ux)$ is a square in $\mathbb{F}_q$.This proves (i).

(ii) We have also

$$\begin{aligned} g(ux) &= u^3 g(x) \Leftrightarrow \\ ux(u^2x^2 + aux + b) &= u^3 x(x^2 + ax + b) \Leftrightarrow \\ aux + b &= au^2 x + u^2 b \Leftrightarrow \\ x &= \frac{-b}{a}(\frac{1+u}{u}). \end{aligned}$$

□

Theorem 2.1. *If $r, \lambda \in \mathbb{F}_q$, λ is a quadratic non residue and*

$$X_1(r) = \frac{-b}{a}(\frac{1+\lambda r^2}{\lambda r^2}), \qquad X_2(r) = \frac{-b}{a}(1+\lambda r^2)$$

then one and only one of the values $g(X_1(r))$ or $g(X_2(r))$ is a square in $\mathbb{F}_q$.

Proof. Since λ is a quadratic non residue in $\mathbb{F}_q$, by taking $u = \lambda r^2$ we have $X_1(r) = x$ and $X_2(r) = ux$ in previous lemma. □

Note that if we want to use X_1, X_2 for every $r \in {\mathbb{F}_q}^*$ the values $r, -r, \frac{1}{\lambda r}, -\frac{1}{\lambda r}$ give us a same abscissa point on $E_{a,b}$. Since for $r \in \mathbb{F}_q$ we have $X_1(r) = X_1(-r)$ and $X_2(r) = X_2(-r)$ and if $X_1(r_1) = X_2(r_2)$ for $r_1, r_2 \in \mathbb{F}_q$ we have:

$$\frac{-b}{a}(\frac{1+\lambda r_1^2}{\lambda r_1^2}) = \frac{-b}{a}(1+\lambda r_2^2) \Rightarrow r_2 = \pm\frac{1}{\lambda r_1}.$$

In order to have an injective encoding we focus our attention to a subset of $\mathbb{F}_q$ such that it has exactly one of the values x or $-x$.
Let $\{1, \alpha, \alpha^2, \ldots, \alpha^{n-1}\}$ be a basis for $\mathbb{F}_q$ over $\mathbb{F}_p$. We know that every element of the field $\mathbb{F}_p$ can be expressed as a nonnegative integer smaller than p. So usual order relation can be considered as a order relation for elements of $\mathbb{F}_p$. We define an order relation for elements of $\mathbb{F}_q$ as follows:
If $a, b \in \mathbb{F}_q$ and

$$\begin{aligned} a &= a_0 + a_1\alpha + \ldots + a_{n-1}\alpha^{n-1} \quad a_i \in \mathbb{F}_p, \\ b &= b_0 + b_1\alpha + \ldots + b_{n-1}\alpha^{n-1} \quad b_i \in \mathbb{F}_p \end{aligned}$$

then $a > b$ if the last non-zero entry of the vector

$$(a_0 - b_0, a_1 - b_1, \cdots, a_{n-1} - b_{n-1})$$

is positive. What remains is how we choose half of elements of $\mathbb{F}_q$ with respect to the mentioned order. We define:

$$\begin{aligned} A_0 &= \{(a_0, a_1, \ldots, a_{n-1}) : 0 < a_0 \leq \tfrac{p-1}{2}, \forall 1 \leq i \leq n-1 \quad a_i = 0\}, \\ A_1 &= \{(a_0, a_1, \ldots, a_{n-1}) : 0 < a_1 \leq \tfrac{p-1}{2}, \forall 2 \leq i \leq n-1 \quad a_i = 0\}, \\ &\vdots \\ A_{n-1} &= \{(a_0, a_1, \ldots, a_{n-1}) : 0 < a_{n-1} \leq \tfrac{p-1}{2}\}. \end{aligned}$$

We have:

$$\#A_0 + \#A_1 + \ldots + \#A_{n-1} = \frac{p-1}{2} + \frac{p(p-1)}{2} + \ldots + \frac{p^{n-1}(p-1)}{2} = \frac{p^n - 1}{2}.$$

Algorithm 1 Injective encoding

Require: Finite field $\mathbb{F}_q$, $r \in A$ and a quadratic non-residue $\lambda \in \mathbb{F}_q$
Ensure: $(x, y) \in E_{a,b}(\mathbb{F}_q)$ where $E_{a,b} : y^2 = x(x^2 + ax + b)$.
if $r = 0$ then return $(0, 0)$
end if
$u = \lambda r^2$
if $1 + u = 0$ then return O
end if
$X_1 = \frac{-b}{a}\frac{1+u}{u}$ $\quad X_2 = uX_1$
$g_1 = X_1(X_1^2 + aX_1 + b)$ $\quad g_2 = X_2(X_2^2 + aX_2 + b)$
if g_1 is a square, return $(X_1, \sqrt{g}_1)$, **else** return $(X_2, -\sqrt{g}_2)$
end if

So we have determined half of elements of $\mathbb{F}_q$ with respect to the mentioned order. In fact we restrict ourselves to the set

$$A = A_1 \cup A_2 \cup \cdots \cup A_{n-1}$$

which has cardinality $\frac{p^n-1}{2}$ and for each $x \in \mathbb{F}_q$ exactly one of x or $-x$ is in A.

Theorem 2.2. *The following properties hold:*

(i) Algorithm 1 defines an injective encoding e from half of $\mathbb{F}_q$ *to* $E_{a,b}(\mathbb{F}_q)$.

(ii) Algorithm 1 and its inverse (that is computing the preimage of outpot of algorithm 1) can be executed in deterministic polynomial time. More precisely with $O(\log^3 q)$ *running time.*

(iii) Almost half of the points on the curve $E_{a,b}(\mathbb{F}_q)$ *appear in the image of e. In fact*

$$\left| \frac{\#Im(e)}{\#E_{a,b}(\mathbb{F}_q)} - \frac{1}{2} \right| \leqslant \frac{1}{\sqrt{q} - 1}.$$

Proof. (i). To compute the preimages of a point $P = (X_P, Y_P)$ we must solve the equations $X_1(r) = X_P, X_2(r) = X_P$. Each of these equations have at most one solution in A. The minus sign in the final step of algorithm assures us that set of points obtained in the two cases in the last step of the algorithm are separated.

(ii). the number of multiplication steps is equal to the number of binary digits q plus the number of ones in the binary representation of q minus 1. So, it is at most equal to $2\,[|\log q\,|] + 1$. Since one multiplication takes time $O(\log^2 q)$ [11] the time of the powering algorithm is $O(\log^3 q)$.

The inverse algorithm is easy to implement. Let (x, y) be a point in the image of encoding. If $\sqrt{g(x)} \equiv y(mod\ q)$ we must solve $X_1\,(r) = x$ and if $\sqrt{g(x)} \equiv -y(mod\ q)$ we must solve $X_2\,(r) = x$ to obtain $r \in \mathbb{F}_q$. Since it is a powering algorithm, time complexity is $O(\log^3 q)$.

(iii). Since encoding function is injective, the size of image of encoding function is $\frac{q-1}{2}$. Using Hasse theorem [15] we have:

$$\frac{\#Im(e)}{\#E_{a,b}(\mathbb{F}_q)} \leqslant \frac{\frac{q-1}{2}}{q+1-2\sqrt{q}} = \frac{q-1}{2(q-1)+4(1-\sqrt{q})} = \frac{\sqrt{q}+1}{2(\sqrt{q}+1)-4}$$
$$= \frac{\sqrt{q}-1+2}{2(\sqrt{q}-1)} = \frac{1}{2} + \frac{1}{\sqrt{q}-1} \Rightarrow |\frac{\#Im(e)}{\#E(\mathbb{F}_q)} - \frac{1}{2}| \leqslant \frac{1}{\sqrt{q}-1}.$$

This completes the proof. □

Remark 2.1. *In the last step of algorithm 1 we need to compute a square root. If $q \equiv 3(mod\ 4)$ then $\sqrt{g_i} = g_i^{\frac{q+1}{4}}$. If $q \equiv 1(mod\ 4)$ since λ is a quadratic non-residue, Tonelli-Shanks [6] algorithm can compute the square root deterministicly in polynomial time.*

Definition 2.1. A Montgomery curve over a field $\mathbb{F}$ is defined by the equation

$$E_{M,A,B} : By^2 = x^3 + Ax^2 + x,$$

where $A, B \in \mathbb{F}$ and $B(A^2 - 4) \neq 0$.

Definition 2.2. A twisted Edwards curve over a field $\mathbb{F}$ with $char(\mathbb{F}) \neq 2$ is defined by the equation:

$$E_{TE,a,d} : ax^2 + y^2 = 1 + dx^2y^2,$$

where a, d are distinct non-zero elements of $\mathbb{F}$.

Remark 2.2. *We know [1] that twisted Edwards curves are birationally equivalent to Montgomery curves via the map*

$$\phi : E_{TE,a,d} \longrightarrow E_{M,\frac{2(a+d)}{a-d},\frac{4}{a-d}}$$
$$\phi(x,y) = (\frac{1+y}{1-y}, \frac{1+y}{x(1-y)}).$$

Morover, the Montgomery curve $E_{M,A,B}$ is birationally equivalent to the twisted Edwards curve $E_{TE,a,d}$ where $a = \frac{(A+2)}{B}$ and $d = \frac{(A-2)}{B}$. Therefore this encoding also can be used for twisted Edwards curves.

2.1. Description of the target curves

In the previous section we discussed about an injective encoding to a family of elliptic curves over $\mathbb{F}_q$ with the point of order 2. This family is

$$E_{a,b} = \{y^2 = x(x^2 + ax + b) : a, b \in \mathbb{F}_q, a^2 - 4b \neq 0, b \neq 0\}.$$

From Hasse theorem [15] we know that $\#E_{a,b}$ lies in interval $[q+1-2\sqrt{q}, q+1+2\sqrt{q}]$. If $N = q + 1 - t$ is not equal to the number of points of an elliptic curve, then

$\gcd(t,q) \neq 1$ and at most 5 values of t can exist such that $\gcd(t,q) \neq 1$ and $N = q+1-t$ occures as number of points of some elliptic curves over $\mathbb{F}_q$ [17] .Therefore there are $2\sqrt{q}(1-\frac{1}{p})$ numbers $N \equiv 0(mod\ 2)$ that occurs as the number of points $E_{a,b}(\mathbb{F}_q)$. Since there are $(q-1)^2$ elliptic curves $E_{a,b}(\mathbb{F}_q)$ we can find $\frac{(q-1)^2}{2\sqrt{q}(1-\frac{1}{p})}$ values of $a, b \in \mathbb{F}_q$ such that $\#E_{a,b}(\mathbb{F}_q)$ is equal to a given N.

Theorem 2.3. *If a^2-4b is a quadratic non-residue and b is a quadratic residue then only the point $(0,0)$ is divisible by 2 and the rational 4-torsion subgroup of $E_{a,b}$ is $\mathbb{Z}_4$.*

If a^2-4b is a quadratic residue and b is a quadratic non-residue then the point $(0,0)$ is not divisible by 2 and the rational 4-torsion subgroup of $E_{a,b}$ is $\mathbb{Z}_2 \times \mathbb{Z}_2$ or $\mathbb{Z}_2 \times \mathbb{Z}_4$.

Proof. Since in the first case, the discriminant of x^2+ax+b is a quadratic non residue, then $(0,0)$ is the only point of order two over $E_{a,b}(\mathbb{F}_q)$. We show that there is a point $Q \in E_{a,b}(\mathbb{F}_q)$ so that $2Q = (0,0)$. If $y = cx$ is a passing line from the point $(0,0)$ the other intersection points with $E_{a,b}(\mathbb{F}_q)$ are obtained from $c^2x = x^2+ax+b$. If this line is tangent, the disciminant of this equation is equal to zero. Therefore $(a-c^2)^2 = 4b$ and we have $c^2 = a \pm 2\sqrt{b}$. Since

$$(a-2\sqrt{b})(a+2\sqrt{b}) = a^2-4b$$

is a quadratic non residue exactly one of the values $a-2\sqrt{b}$ or $a+2\sqrt{b}$ is square. Therefore the above equation has two solutions for only one of the two possible signs. It gives two answers for Q. In the same way we can find out that the point $(0,0)$ is not divisible by 2 in the second case. It is easy to see that rational 4 torsion is $\mathbb{Z}_4$ in the first case or $\mathbb{Z}_2 \times \mathbb{Z}_2$ or $\mathbb{Z}_2 \times \mathbb{Z}_4$ in the second case. In fact it depends to this subject that the points of order 2 except (0,0) is divisible by two. □

We know that if $g \in \mathbb{F}[X]$ and $deg(g) = n$, then g has a splitting field $\mathbb{K}$ over $\mathbb{F}$ with $[\mathbb{K} : \mathbb{F}] \leq n!$. Therefore every elliptic curve E with Weierstrass form $y^2 = x^3 + a_1x + b_1$ can be written as :

$$E_{a,b,c} : y^2 = (x-a)(x-b)(x-c)$$

over $\mathbb{F}_{q^\alpha}$, where α is $1, 2, 3$ or 6.

Theorem 2.4. *If $a, b, c \in \mathbb{F}_q$ then*

$$\sum_{a,b,c \in \mathbb{F}_q} trace(E_{a,b,c}) = 0.$$

Proof. we know

$$\sum_{a,b,c \in \mathbb{F}_q} \#E_{a,b,c}(\mathbb{F}_q) = \sum_{a,b,c \in \mathbb{F}_q} q+1-trace(E_{a,b,c}).$$

Therefore it is enough to show

$$\sum_{a,b,c\in\mathbb{F}_q} \#E_{a,b,c}(\mathbb{F}_q) = (q+1)\begin{pmatrix} q \\ 3 \end{pmatrix}$$

We assume S_0 is the number of finite points on all elliptic curves of the form $E_{0,b,c}$ and S_1 is the number of finite points on all elliptic curves of the form $E_{1,b,c}$. In the same way we define $S_2, \ldots, S_{q-2}$. We also assume $S_{0,1}$ is the number of finite points on all elliptic curves of the form $E_{0,1,c}$ and $S_{0,2}$ is number of finite points on all elliptic curves of the form $E_{0,2,c}$. In the same way we define $S_{0,3}, \ldots, S_{q-1,q-2}$. We have:

$$\begin{aligned} S_0 &= S_{0,1} + S_{0,2} + \ldots + S_{0,q-1}. \\ S_1 &= S_{1,0} + S_{1,2} + \ldots + S_{1,q-1}. \\ &\vdots \\ S_{q-1} &= S_{q-1,0} + S_{q-1,1} + \ldots S_{q-1,q-2}. \end{aligned}$$

Let $(\frac{\cdot}{\mathbb{F}_q})$ be the legendre symbol over $\mathbb{F}_q$. For example to calculate $S_{0,1}$ we observe:

$$\begin{aligned} T_{0,1} &= |\{(x,y,c) \in \mathbb{F}_q^3 : y^2 = x(x-1)(x-c)\}| = q^2, \\ T_{0,1,0} &= |\{(x,y) \in \mathbb{F}_q^2 : y^2 = x^2(x-1)\}| = q - (\tfrac{-1}{\mathbb{F}_q}), \\ T_{0,1,1} &= |\{(x,y) \in \mathbb{F}_q^2 : y^2 = x(x-1)^2\}| = q-1. \end{aligned}$$

Since in $S_{0,1}$ we have $c \neq 0, 1$ then

$$S_{0,1} = T_{0,1} - T_{0,1,0} - T_{0,1,1} = q^2 - 2q + (\frac{-1}{\mathbb{F}_q}) + 1.$$

In the same way for calculating $S_{0,2}$ we have :

$$\begin{aligned} T_{0,2} &= |\{(x,y,c) \in \mathbb{F}_q^3 : y^2 = x(x-2)(x-c)\}| = q^2, \\ T_{0,2,0} &= |\{(x,y) \in \mathbb{F}_q^2 : y^2 = x^2(x-2)\}| = q - (\tfrac{-2}{\mathbb{F}_q}), \\ T_{0,2,2} &= |\{(x,y) \in \mathbb{F}_q^2 : y^2 = x(x-2)^2\}| = q - (\tfrac{2}{\mathbb{F}_q}). \end{aligned}$$

Therefore we have:

$$S_{0,2} = T_{0,2} - T_{0,2,0} - T_{0,2,2} = q^2 - 2q + (\frac{-2}{\mathbb{F}_q}) + (\frac{2}{\mathbb{F}_q}).$$

Following this process we have:

$$S_{0,q-1} = q^2 - 2q + (\frac{-(q-1)}{\mathbb{F}_q}) + (\frac{q-1}{\mathbb{F}_q}).$$

Therefore

$$S_0 = S_{0,1} + S_{0,2} + \ldots + S_{0,q-1} = (q-1)(q^2 - 2q).$$

Similarly we can see that

$$S_0 = S_1 = S_2 = \ldots = S_{q-1} = q(q-1)(q-2).$$

Therefore:

$$S_0 + S_1 + \ldots + S_{q-2} + S_{q-1} = q^2(q-1)(q-2).$$

Since for every triple (a, b, c) the rearrangements will give the same elliptic curve $E_{a,b,c}$ and considering the points at infinity we have:

$$\sum_{a,b,c \in \mathbb{F}q} \#E_{a,b,c}(\mathbb{F}_q) = \frac{q^2(q-1)(q-2)}{3!} + \left(\begin{array}{c} q \\ 3 \end{array} \right) = (q+1) \left(\begin{array}{c} q \\ 3 \end{array} \right).$$

□

3. Uniform encoding

As mentined in the introduction Brier et al. [4] proposed a method based on a variant of the Shallue-Woestijne-Ulas (SWU) function where $q \equiv 3(mod\ 4)$.

Algorithm 2 Simlified SWU encoding Algorithm

Require: Finite field $\mathbb{F}_q$, $t \in \mathbb{F}_q$
Ensure: $(x, y) \in E(\mathbb{F}_q)$ where $E : y^2 = x^3 + ax + b$.
$u = -t^2$
if $u^2 + u = 0$ return O
end if
$X_1 = -\frac{b}{a}(1 + \frac{1}{u^2+u})$ $\quad X_2 = u.X_1$
$g_1 = X_1^3 + aX_1 + b$ $\quad g_2 = X_2^3 + aX_2 + b$
if g_1 is a square return $(X_1, g_1^{\frac{q+1}{4}})$ **else** return $(X_2, g_2^{\frac{q+1}{4}})$
end if

Definition 3.1. Let E be an elliptic curve over a finite field $\mathbb{F}_q$. Let S be a subset of $\mathbb{F}_q$ and let f be an encoding $f : S \longrightarrow E(\mathbb{F}_q)$. Let $R = f(S)$. The encoding f is uniform on S if the followings hold:
1. Computable: f is computable in deterministic polynomial time;
2. ℓ-to-1: for any $r \in R$, $\#f^{-1}(r) = \ell$;
3. Samplable: there exists a probabilistic polynomial time algorithm such that for any $r \in R$ returns a random element in $f^{-1}(r)$.

Brier et al. [4] indicated that if $f : S \longrightarrow E(\mathbb{F}_q)$ is a uniform encoding, then the construction $H(m) = f(h(m))$ is indifferentiable from a random oracle in the random oracle model for $h : \{0,1\}^* \longrightarrow S$. In order to have a uniform encoding we alter simplified SWU method [4] such that any point has exactly 4 preimages. Notice that A is the set defined in section 2..

Algorithm 3 Uniform encoding

Require: Finite field $\mathbb{F}_q$, $t \in \mathbb{F}_q$ and a quadratic non-residue $\eta \in \mathbb{F}_q$.
Ensure: $(x, y) \in E(\mathbb{F}_q)$ where $E : y^2 = x^3 + ax + b$.

$u = \eta t^2$
if $u^2 + u = 0$ return O
end if
$X_1 = -\frac{b}{a}(1 + \frac{1}{u^2+u})$ $\quad X_2 = u.X_1$
$g_1 = X_1^3 + aX_1 + b$ $\quad g_2 = X_2^3 + aX_2 + b$
if g_1 is a square
  if $u \in A$ return $(X_1, \sqrt{g_1})$ **else** return $(X_1, -\sqrt{g_1})$
  end if
else
  if $\frac{1}{u} \in A$ return $(X_2, \sqrt{g_2})$ **else** return $(X_2, -\sqrt{g_2})$
  end if
end if

Theorem 3.1. *Algorithm 3 defines a uniform encoding from $\mathbb{F}_q$ to $E(\mathbb{F}_q)$.*

Proof. It's easy to see that, except for the point at infinity and points of order 2, all points appear four times in the output of the algorithm. The minus signs in the final step of algorithm assures that set of points obtained of forms $(X_1, \sqrt{g_1})$ and $(X_1, -\sqrt{g_1})$ are separated. In the same way the points of forms $(X_2, \sqrt{g_2})$ and $(X_2, -\sqrt{g_2})$ are seperated. Without lose of generality let (x, y) be a point that is generated with the form $(X_1, \sqrt{g_1})$. We notice that there is only one u such that $u \in A$ and $X_1(u) = x$. Because if u_1, u_2 are such that $X_1(u_1) = X_1(u_2)$ then we have $u_1 + u_2 = -1$. It is easy to see that one and only one of u_1 and u_2 is in A. The point (x, y) is also generated by $(X_2, \sqrt{g_2})$ for the value $w = \frac{1}{u}$. because

$$X_2(w) = X_2(1/u) = X_1(u) = x$$

and $\frac{1}{w} = u \in A$. Condition $\frac{1}{w} \in A$ ensures that w is the only value that can generate this point by X_2. Since each of u, w comes by two values $\pm t$ therefore (x, y) has exactly 4 preimages. □

4. Encoding to hyperelliptic curves

Ulas [16] simplified and generalized the proposed method by Shallu and Woestijne [14] to encode $\mathbb{F}_q$ to hyperelliptic curves of the forms $y^2 = x^n + ax + b$ and $y^2 =$

$x^n + ax^2 + bx$. Foque and Tibouchi [10] proposed a deterministic encoding in to hyperelliptic curves of the form

$$y^2 = x^{2g+1} + a_1 x^{2g-1} + \cdots + a_g x,$$

where g is the genus of the curve.
In this section we recall SWU algorithm for genus 2 hyperelliptic curves of the form $H_{a,b} : y^2 = x^5 + ax^2 + bx$ where $q \equiv 3(mod\ 4)$. We show that this algorithm is an injective encoding for finite fields of characteristic 3.

Lemma 4.1. *Let $g(x) = x^5 + ax^2 + bx$. Let u be a quadratic non-residue in $\mathbb{F}_q$ such that for some $x \in \mathbb{F}_q$ we have*

$$g(ux) = u^5 g(x). \tag{4.1}$$

then either x or $u.x$ is the abscissa of a point on the $y^2 = g(x)$. Moreover for each u the value

$$x = \frac{-b}{a}\frac{(1-u^4)}{u-u^4}$$

satisfies (4.1).

Proof. Since u is not a quadratic residue, if x satisfies 4.1 then either $g(u.x)$ or $g(x)$ must be a square in $\mathbb{F}_q$. Therefor either x or $u.x$ must be abscissa of a point on the curve $y^2 = g(x)$. Moreover we have:

$$\begin{aligned} g(ux) = u^5 g(x) &\iff (ux)^5 + a(ux)^2 + bx = u^5(x^5 + ax^2 + bx) \\ &\iff aux + b = au^4 x + bu^4 \\ &\iff x = \frac{-b}{a}\frac{(1-u^4)}{u-u^4}. \end{aligned}$$

□

Since -1 is a quadratic non residue in $\mathbb{F}_q$, for every $t \in \mathbb{F}_q$, $u = -t^2$ is a quadratic non-residue in $\mathbb{F}_q$. So we have algorithm 4 in the next page.

Algorithm 4 new simplified SWU algorithm

Require: Finite field $\mathbb{F}_q$ such that $q \equiv 3(mod\ 4)$, $t \in \mathbb{F}_q$
Ensure: $(x, y) \in H_{a,b}(\mathbb{F}_q)$ where $H_{a,b} : y^2 = x^5 + ax^2 + bx$.
$u = -t^2$
$X_1 = \frac{-b}{a}\frac{(1-u^4)}{(u-u^4)}$ $\quad X_2 = u.X_1$
$g_1 = X_1^5 + aX_1^2 + bX_1$ $\quad g_2 = X_2^5 + aX_2^2 + bX_2$
if g_1 is a square, return $(X_1, g_1^{\frac{q+1}{4}})$, **else** return $(X_2, -g_2^{\frac{q+1}{4}})$
end if

Remark 4.1. *In order to compute the pre-images of a point $P = (X_P, Y_P)$ we should solve the equations $X_1(t) = X_P$ and $X_2(t) = X_P$. Since $degX_1(t) = 6$ and $degX_2(t) = 6$ each equation has at most 6 solutions. The minus sign in the final step of the algorithm makes the set of points obtained of form $(X_1, g_1^{\frac{q+1}{4}})$ and set of points obtained of form $(X_2, -g_2^{\frac{q+1}{4}})$ separated. Hence a point has at most 6 pre-images.*

According to the previous remark, a point in the outpot of algorithm 4 is generated by at most 6 elements of $\mathbb{F}_q$. Since $u = -t^2$, this point is generated by at most 3 quadratic non-residues in $\mathbb{F}_q$. we want to find the quadratic non-residues that for them there is no partner in the production of a point. We notice that if $X_1(u_1) = X_1(u_2)$ we have:

$$u_1^4u_2^4 - u_1^4u_2^3 - u_1u_2^4 + u_2^3 + u_1 - 1 = 0.$$

By dividing by $1 - u_1$ and $1 - u_2$ and $u_1 - u_2$ we have:

$$u_1^2 + (u_2 + 1)u_1 + (u_2^2 + u_2 + 1) = 0. \tag{4.2}$$

The discriminant of equation 4.2 is $\Delta_1 = -3u_2^2 - 2u_2 - 3$. This equation has no solution whenever Δ_1 is a quadratic non-residue.
Also if $X_2(u_1) = X_2(u_2)$ we have:

$$u_1^4 + u_1^4u_2^3 + u_2^3 - u_2^4 + u_1^3u_2^4 - u_1^3 = 0.$$

Similarly by dividing by $1 - u_1$ and $1 - u_2$ and $u_1 - u_2$ we have:

$$(u_2^2 + u_2 + 1)u_1^2 + (u_2^2 + u_2)u_1 + u_2^2 = 0. \tag{4.3}$$

The discriminant of equation 4.3 is $\Delta_2 = u_2^2(-3u_2^2 - 2u_2 - 3)$. This equation has no solution whenever Δ_2 is a quadratic non-residue. By looking at equations $\Delta_1 = -3u_2^2 - 2u_2 - 3$ and $\Delta_2 = u_2^2(-3u_2^2 - 2u_2 - 3)$, we find out that they are quadratic non-residues if for any u as quadratic non-residue the value $\Delta = -3u^2 - 2u - 3$ is a quadratic non-residue.

Corollary 4.1. *If we consider $H_{a,b}$ over finite fields of characteristic 3, the Δ value is always a quadratic non residue. Therefore if A is the set defined in section 2., algorithm 4 is an injective encoding from A into points $H_{a,b}$ for these finite fields.*

Remark 4.2. *We know that the set of points on $H_{a,b}$ is not a group. Therefore if for cryptographic purposes we need to be in a group, we can map $H_{a,b}$ to the jacobian J of $H_{a,b}$ which is an abelian group.*

REFERENCES

1. D. J. Bernstein, P. Birkner, M. Joye, T. Lange and C. Peters: Twisted edwards curves. In: International Conference on Cryptology in Africa, Springer, Berlin, Heidelberg, 2008, p. 389-405.
2. D. J. Bernstein, M. Hamburg, A. Krosnova and T. Lange: Elliptic-curve points indistinguishable from uniform random strings. In: Proceedings of the 2013 ACM SIGSAC conference on Computer and communications security, ACM, 2013, p. 967-980.
3. D. Boneh, M. Franklin: Identity-based encryption from the Weil pairing. In: Annual international cryptology conference, Springer, Berlin, Heidelberg, 2001, p. 213-229.
4. E. Brier, J. -S. Coron, T. Icart, D. Madore, H. RandriaM and M. Tibouchi: Efficient indifferentiable hashing into ordinary elliptic curves. In: Annual Cryptology Conference, Springer, Berlin, Heidelberg, 2010, p. 237-254.
5. O. Chevassut, P. A. Fouque, P. GaudRY and D. Pointcheval: The twist-augmented technique for key exchange. In: International Workshop on Public Key Cryptography, Springer, Berlin, Heidelberg, 2006, p. 410-426.
6. H. Cohen: A course in computational algebraic number theory. volume 138 of Graduate Texts in Mathematics, Springer-Verlag, Berlin, 1993.
7. R. R. Farashahi: Hashing into Hessian curves. In: International Conference on Cryptology in Africa, Springer, Berlin, Heidelberg, 2011, p. 278-289.
8. R. R. Farashahi, I. E. Shparlinski and J. F. Voloch: On hashing into elliptic curves. Journal of Mathematical Cryptology, 2009, 3.4: 353-360.
9. P. A. Fouque, A. Joux and M. Tibouchi: Injective encodings to elliptic curves. In: Australasian Conference on Information Security and Privacy, Springer, Berlin, Heidelberg, 2013, p. 203-218.
10. P.A. Fouque, M. Tibouchi: Deterministic encoding and hashing to odd hyperelliptic curves. In: International Conference on Pairing-Based Cryptography, Springer, Berlin, Heidelberg, 2010, p. 265-277.
11. T. Icart: How to hash into elliptic curves. In: Advances in Cryptology-CRYPTO 2009, Springer, Berlin, Heidelberg, 2009, p. 303-316.
12. N. Koblitz : Elliptic curve cryptosystems. Mathematics of computation, 1987, 48.177: 203-209.
13. A. J. Menezes, T. Okamoto and S. A. Vanstone: Reducing elliptic curve logarithms to logarithms in a finite field. iEEE Transactions on information Theory, 1993, 39.5: 1639-1646.
14. A. Shallue, C. E. Van De Woestijne: Construction of rational points on elliptic curves over finite fields. In: International Algorithmic Number Theory Symposium, Springer, Berlin, Heidelberg, 2006, p. 510-524.
15. J.H. Silverman: Advanced Topics in the Arithmetic of Elliptic Curves. Grad. Texts in Math., vol. 151, Springer, New York, 1994.
16. M. Ulas: Rational points on certain hyperelliptic curves over finite fields. Bull. Polish Acad. Sci. Math., 55(2), 2007: 97-104.
17. W.C. Waterhouse, Abelian Varieties over Finite Fields, Annales scientifiques de l'École Normale Supérieure 2, no. 4 (1969).

Amirmehdi Yazdani Kashani
Faculty of Mathematical Sciences
Department of Pure Mathematics
University of Kashan, Iran
yazdani@grad.kashanu.ac.ir

Hassan Daghigh
Faculty of Mathematical Sciences
Department of Pure Mathematics
University of Kashan, Iran
hassan@kashanu.ac.ir