

Hashing into Generalized Huff Curves

Xiaoyang He^{1,2,3}, Wei Yu^{1,2(✉)}, and Kunpeng Wang^{1,2}

¹ State Key Laboratory of Information Security,
Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
hexiaoyang@iie.ac.cn

² Data Assurance and Communication Security Research Center,
Chinese Academy of Sciences, Beijing, China
yuwei_1-yw@163.com

³ University of Chinese Academy of Sciences, Beijing, China

Abstract. Huff curves are well known for efficient arithmetics to their group law. In this paper, we propose two deterministic encodings from \mathbb{F}_q to generalized Huff curves. When $q \equiv 3 \pmod{4}$, the first deterministic encoding based on Skarpa's equality saves three field squarings and five multiplications compared with birational equivalence composed with Ulas' encoding. It costs three multiplications less than simplified Ulas map. When $q \equiv 2 \pmod{3}$, the second deterministic encoding based on calculating cube root costs one field inversion less than Yu's encoding at the price of three field multiplications and one field squaring. It costs one field inversion less than Alasha's encoding at the price of one multiplication. We estimate the density of images of these encodings with Chebotarev density theorem. Moreover, based on our deterministic encodings, we construct two hash functions from messages to generalized Huff curves indifferentiable from a random oracle.

Keywords: Elliptic curves · Generalized Huff curves · Character sum · Hash function · Random oracle

1 Introduction

Plenty of elliptic/hyperelliptic curve cryptosystems require hashing into algebraic curves. Many identity-based schemes need messages to be hashed into algebraic curves, including encryption schemes [1, 2], signature schemes [3, 4], signcryption schemes [5, 6], and Lindell's universally-composable scheme [7]. The simple password exponential key exchange [10] and the password authenticated key exchange protocols [11] both require a hash algorithm to map the password into algebraic curves.

Boneh and Franklin [8] proposed an algorithm to map elements of \mathbb{F}_q to rational points on an ordinary elliptic curve. This algorithm is probabilistic and

This work is supported in part by National Research Foundation of China under Grant No. 61502487, 61272040, and in part by National Basic Research Program of China (973) under Grant No. 2013CB338001.

fails to return a point at the probability of $1/2^k$, where k is a predetermined bound. One disadvantage of this algorithm is that its total number of running steps depends on the input $u \in \mathbb{F}_q$, hence is not constant. Thus the algorithm may be threaten by timing attacks [9], and the information of the message may leaked out. Therefore, it is significant to find algorithms hashing into curves in constant number of operations.

There exist various algorithms encoding elements of \mathbb{F}_q into elliptic curves in deterministic polynomial time. When $q \equiv 3 \pmod{4}$, Shallue and Woestijne proposed an algorithm [12] based on Skalba's equality [13], using a variation of Tonelli-Shanks algorithm to calculate square roots efficiently as $x^{1/2} = x^{(q+1)/4}$. Fouque and Tibouchi [14] simplified this encoding by applying brief version of Ulas' function [15]. Moreover, they generalized Shallue and Woestijne's method so as to hash into some special hyperelliptic curves. When $q \equiv 2 \pmod{3}$, Icart [16] gave an algorithm based on computing cube roots efficiently as $x^{1/3} = x^{(2q-1)/3}$ in Crypto 2009. Both algorithms encode elements of \mathbb{F}_q into curves in short Weierstrass form.

After initial algorithms listed above, hashing into Hessian curves [17] and Montgomery curves [18] were proposed. Alasha [19] constructed deterministic encodings into Jacobi quartic curves, Edwards curves and Huff curves. Yu constructed a hash function from plaintext to C_{34} -curves by finding a cube root [20].

Huff curves, first introduced by Huff [21] in 1948, were utilized by Joye, Tibouchi and Vergnaud [22] to develop an elliptic curve model over a finite field K where $\text{char}(K) > 2$. They also presented the efficient explicit formulas for adding or doubling points on Huff curves. In 2011, Ciss and Sow [27] introduced generalized Huff curves: $ax(y^2 - c) = by(x^2 - d)$ with $abcd(a^2c - b^2d) \neq 0$, which contain classical Huff curves [22] as special cases. Wu and Feng [23] independently presented another kind of curves they also called generalized Huff curves: $x(ay^2 - 1) = y(bx^2 - 1)$, which is in fact an equivalent variation of Ciss and Sow's construction. Wu and Feng constructed arithmetic and pairing formulas on generalized Huff curves. Generalized Huff curves own an effective group law and unified addition-doubling formula, hence are resistant to side channel attacks [24]. Devigne and Joye also analyzed Huff curves over binary fields [28]: $ax(y^2 + cy + 1) = by(x^2 + cx + 1)$ with $abc(a - b) \neq 0$.

We propose two deterministic encodings directly from \mathbb{F}_q to generalized Huff curves: brief Shallue-Woestijne-Ulas (SWU) encoding and cube root encoding. Based on Skalba's equality [13], brief SWU encoding costs three field squarings and five multiplications less than birational equivalence from short Weierstrass curve to generalized Huff curve composed with Ulas' original encoding [15]. It saves three squarings less than birational equivalence from short Weierstrass curve to generalized Huff curve composed with simplified Ulas map [26]. To prove our encoding's B-well-distributed property, we estimate the character sum of an arbitrary non-trivial character defined over generalized Huff curves through brief SWU encoding. We also estimate the size of image of brief SWU encoding. Based on calculating cube root of elements in \mathbb{F}_q , cube root encoding saves one field inversion compared with Yu's encoding function at the price

of one field multiplication. It saves one field inversion compared with Alasha's encoding at the price of one field squaring and three field multiplications. We estimate the relevant character sum and the size of image of cube root encoding in similar way.

Based on brief SWU encoding and cube root encoding, we construct two hash functions efficiently mapping binary messages into generalized Huff curves, which are both indifferentiable from random oracle.

We do experiments over 192-bit prime field $\mathbb{F}_{P_{192}}$ and 384-bit prime field $\mathbb{F}_{P_{384}}$ recommended by NIST in the elliptic curve standard [25]. On both fields, there exist efficient algorithms to calculate the square root and cube root for each element. On $\mathbb{F}_{P_{192}}$, our cube root encoding f_I saves 13.20 % running time compared with Alasha's encoding function f_A , 8.97 % with Yu's encoding f_Y , on $\mathbb{F}_{P_{384}}$, f_I saves 7.51 % compared with f_A and 4.40 % with f_Y . Our brief SWU encoding f_S also runs faster than f_U , birational equivalence composed with Ulas' encoding function and f_E , birational equivalence composed with Fouque and Tibouchi's brief encoding. Experiments show that f_S saves 9.19 % compared with f_U and 7.69 % with f_E on $\mathbb{F}_{P_{192}}$, while it saves 5.92 % compared with f_U and 5.17 % with f_E on $\mathbb{F}_{P_{384}}$.

Organization of the Paper. In Sect. 2, we recall some basics of generalized Huff curves. In Sect. 3, we introduced brief SWU encoding, prove its B-well-distributed property by estimating the character sum of this encoding, and calculate the density of image of the encoding. In Sect. 4, we proposed the cube root encoding, also prove its B-well-distributed property and calculate the density of image of the encoding by similar methods. In Sect. 5, we construct 2 hash functions indifferentiable from random oracle. In Sect. 6, time complexity of given algorithms is analysed, and we presented the practical results. Section 7 is the conclusion of the paper.

2 Generalized Huff Curves

Suppose \mathbb{F}_q is a finite field whose characteristic is greater than 2.

Definition 1 ([27]). *Generalized Huff curve can be written as:*

$$ax(y^2 - c) = by(x^2 - d),$$

where $a, b, c, d \in \mathbb{F}_q$ with $abcd(a^2c - b^2d) \neq 0$.

For generalized Huff curve E , if $c = \gamma^2, d = \delta^2$ are squares of \mathbb{F}_q , let $(x, y) = (\delta x', \gamma y')$, we find that E is \mathbb{F}_q -isomorphic to classical Huff curve $(a\delta\gamma^2)x'(y'^2 - 1) = (b\delta^2\gamma)y'(x'^2 - 1)$. If c or d is not a square of \mathbb{F}_q , there exists no relevant classical Huff curve which is \mathbb{F}_q -isomorphic to E . Therefore, generalized Huff curves contain classical Huff curves as a proper subset.

Consider the point sets on projective plane $(X : Y : Z) \in \mathbb{P}^2(\overline{\mathbb{F}_q})$, generalized Huff curve can be written as:

$$aX(Y^2 - cZ^2) = bY(X^2 - dZ^2).$$

Generalized Huff curve has 3 infinity points: $(1 : 0 : 0), (0 : 1 : 0), (a : b : 0)$. We give a picture of generalized curve $3x(y^2 - 1) = -5y(x^2 - 2)$ as shown in Fig. 1 (over \mathbb{R}):

According to [23], a generalized Huff curve over \mathbb{F}_q contains a copy of $\mathbb{Z}/2\mathbb{Z} \times \mathbb{Z}/2\mathbb{Z}$. In fact, every elliptic curve with 3 points of order 2 is \mathbb{F}_q -isomorphism to a generalized Huff curve. In particular, $ax(y^2 - c) = by(x^2 - d)$ is \mathbb{F}_q -isomorphic to $y^2 = x(x + a^2c)(x + b^2d)$.

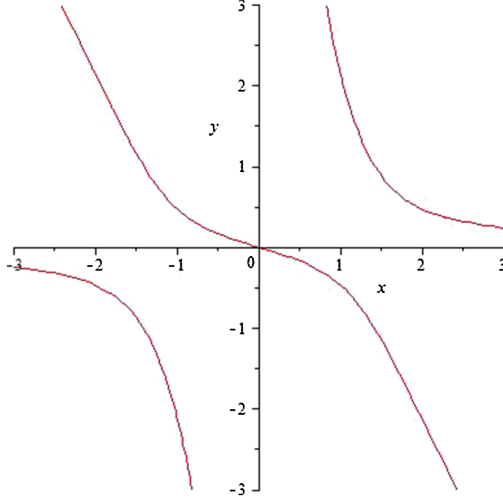


Fig. 1. Generalized Huff Curve $3x(y^2 - 1) = -5y(x^2 - 2)$

3 Brief SWU Encoding

For $q \equiv 3 \pmod{4}$, Ulas presented an encoding function from \mathbb{F}_q to curve $y^2 = x^n + ax^2 + bx$ [15]. We construct our deterministic encoding function f_S by generalizing his method, mapping $u \in \mathbb{F}_q$ to $(x, y) \in E(\mathbb{F}_q)$.

3.1 Algorithm

Input: a, b, c, d and $u \in \mathbb{F}_q$.

Output: A point $(x, y) \in E(\mathbb{F}_q)$.

1. If $u = 0$ then return $(0, 0)$.
2. $X(u) = \frac{a^2b^2cd}{a^2c + b^2d}(u^2 - 1)$.
3. Calculate $g(X(u))$ where $g(s) = s^3 + (a^2c + b^2d)s^2 + a^2b^2cds$.
4. $Y(u) = -\frac{a^2b^2cd}{a^2c + b^2d} \cdot (1 - \frac{1}{u^2})$.

5. Calculate $g(Y(u))$.
6. If $g(X(u))$ is a quadratic residue, then $(s, t) = \left(X(u), -\sqrt{g(X(u))}\right)$,
else $(s, t) = \left(Y(u), \sqrt{g(Y(u))}\right)$.
7. $(x, y) = \left(\frac{bd(s + a^2c)}{t}, \frac{ac(s + b^2d)}{t}\right)$.

According to [14], there exists a function $U(u) = u^3g(Y(u))$, such that the equality

$$U(u)^2 = -g(X(u))g(Y(u)) \quad (1)$$

holds. Thus either $g(X(u))$ or $g(Y(u))$ is a quadratic residue. Choose the one which has square roots in \mathbb{F}_q . Note that $q \equiv 3 \pmod{4}$, we can efficiently calculate the standard square root by $\sqrt{a} = a^{(q+1)/4}$. Hence the mapping $u \mapsto (s, t)$ satisfying $t^2 = g(s)$ is constructed. Then in step 7, we transfer (s, t) to $(x, y) \in E(\mathbb{F}_q)$ by a birational equivalence. It is easy to check that this birational equivalence is one-to-one and onto when it is extended to a map between projective curves. The image of $(0, 0), (-a^2c, 0), (-b^2d, 0)$ are infinite points $(a : b : 0), (0 : 1 : 0), (1 : 0 : 0)$ respectively while the image of $(0 : 1 : 0)$ is $(0, 0)$ on E . Denote the map $u \mapsto (s, t)$ by ρ , and denote the map $(s, t) \mapsto (x, y)$ by ψ , we call the composition $f_S = \psi \circ \rho$ brief SWU encoding. Therefore given $(s, t) \in \text{Im}(\rho)$, either $t = \sqrt{g(s)}$ hence s is the image of $Y(u)$ and has at most 2 preimages, or $t = -\sqrt{g(s)}$ hence s is the image of $X(u)$ and has still at most 2 preimages. Moreover, it is easy to check that ψ is one-to-one. Therefore for each finite point on $E(\mathbb{F}_q)$, and for the infinite point $(a : b : 0)$, f_S has at most 2 preimages, but for the rest 2 infinite points of $E(\mathbb{F}_q)$, whose projective coordinates are $(1 : 0 : 0)$ and $(0 : 1 : 0)$, f_S has at most 4 preimages since the corresponding t vanishes.

3.2 Theoretical Analysis of Time Cost

Let S denote field squaring, M denote field multiplication, I field inversion, E_S the square root, E_C the cube root, D the determination of the square residue. Suppose $a, b, c, d \in \mathbb{F}_q$. In this paper we make the assumption that $S = M$, $I = 10M$ and $E_S = E_C = E$.

The cost of f_S can be calculated as follows:

1. Calculating u^2 costs S , multiplying $u^2 - 1$ by $\frac{a^2b^2cd}{a^2c + b^2d}$ costs M , and it is enough to calculate $X(u)$.
2. To compute $Y(u)$, we need to calculate the inversion of u^2 for $I + M$.
3. When s is known, computing $g(s) = s(s^2 + (a^2c + b^2d)s + a^2b^2cd) = s(s + a^2c)(s + b^2d)$ takes $2M$. To make sure that the algorithm be run in constant time, both $g(X(u))$ and $g(Y(u))$ must be calculated and it requires $4M$.

4. In general case, exact one of $g(X(u))$ and $g(Y(u))$ is a quadratic residue. We only need to check once and it takes D , then compute the square root E_S of the quadratic residue. Then values of s and t are known.
5. Finally, we calculate the inverse of t , which requires I . Then multiplying the inverse by $s + a^2c$ and $s + b^2d$ costs $2M$, then calculating x and y costs $2M$, hence it requires $I + 4M$ in this step.

Therefore, f_S requires $E_S + 2I + 10M + S + D = E + 31M + D$ in all.

3.3 B-Well-Distributed Property of Brief SWU Encoding

Definition 2 (Character Sum). Suppose f is an encoding from \mathbb{F}_q into a smooth projective elliptic curve E , and $J(\mathbb{F}_q)$ denotes the Jacobian group of E . Assume that E has an \mathbb{F}_q -rational point O , by sending $P \in E(\mathbb{F}_q)$ to the deg 0 divisor $(P) - (O)$, we can regard f as an encoding to $J(\mathbb{F}_q)$. Let χ be an arbitrary character of $J(\mathbb{F}_q)$. We define the character sum

$$S_f(\chi) = \sum_{s \in \mathbb{F}_q} \chi(f(s)).$$

We say that f is B-well-distributed if for any nontrivial character χ of $J(\mathbb{F}_q)$, the inequality $|S_f(\chi)| \leq B\sqrt{q}$ holds [29].

Lemma 1 (Corollary 2, Sect. 3, [29]). If f is a B-well-distributed encoding into a curve E , then the statistical distance between the distribution defined by $f^{\otimes s}$ on $J(\mathbb{F}_q)$ and the uniform distribution is bounded as:

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

where

$$f^{\otimes s}(u_1, \dots, u_s) = f(u_1) + \dots + f(u_s),$$

$$N_s(D) = \#\{(u_1, \dots, u_s) \in (\mathbb{F}_q)^s \mid D = f(u_1) + \dots + f(u_s)\},$$

i.e., $N_s(D)$ is the size of preimage of D under $f^{\otimes s}$. In particular, when s is greater than the genus of E , the distribution defined by $f^{\otimes s}$ on $J(\mathbb{F}_q)$ is statistically indistinguishable from the uniform distribution. Especially, in the elliptic curves' case, $g_E = 1$, let $s = g_E + 1 = 2$, the hash function construction

$$m \mapsto f^{\otimes 2}(h_1(m), h_2(m))$$

is indifferentiable from random oracle if h_1, h_2 are seen as independent random oracles into \mathbb{F}_q (See [29]).

Hence, it is of great importance to estimate the character sum of an encoding into an elliptic curve, and we will study the case of generalized Huff curves.

Definition 3 (Artin Character). Let E be a smooth projective elliptic curve, $J(\mathbb{F}_q)$ be Jacobian group of E . Let χ be a character of $J(\mathbb{F}_q)$. Its extension is a multiplicative map $\bar{\chi} : \text{Div}_{\mathbb{F}_q}(E) \rightarrow \mathbb{C}$,

$$\bar{\chi}(n(P)) = \begin{cases} \chi(P)^n, & P \in S, \\ 0, & P \notin S. \end{cases}$$

Here P is a point on $E(\mathbb{F}_q)$, S is a finite subset of $E(\mathbb{F}_q)$, usually denotes the ramification locus of a morphism $Y \rightarrow X$. Then we call $\bar{\chi}$ an Artin character of X .

Theorem 1. Let $h : \tilde{X} \rightarrow X$ be a nonconstant morphism of projective curves, and χ is an Artin character of X . Suppose that $h^*\chi$ is unramified and nontrivial, φ is a nonconstant rational function on \tilde{X} . Then

$$\left| \sum_{P \in \tilde{X}(\mathbb{F}_q)} \chi(h(P)) \left(\frac{\varphi(P)}{q} \right) \right| \leq (2\tilde{g} - 2 + 2 \deg \varphi) \sqrt{q},$$

where $\left(\frac{\cdot}{q} \right)$ denotes Legendre symbol, and \tilde{g} is the genus of \tilde{X} .

Proof. See Theorem 3, [29].

Theorem 2. Let f_S be the brief SWU encoding encoding from \mathbb{F}_q to generalized Huff curve E , $q \equiv 2 \pmod{3}$. For any nontrivial character χ of $E(\mathbb{F}_q)$, the character sum $S_{f_S}(\chi)$ satisfies:

$$|S_{f_S}(\chi)| \leq 16\sqrt{q} + 45.$$

Proof. Let $S = \{0\} \cup \{\text{roots of } g(X(u)) = 0\} \cup \{\text{roots of } g(Y(u)) = 0\}$ where $X(\cdot)$ and $Y(\cdot)$ are defined as in Sect. 3.1. For any $u \in \mathbb{F}_q \setminus S$, $X(u)$ and $Y(u)$ are both well defined and nonzero. Let $C_X = \{(u, s, t) \in \mathbb{F}_q^3 \mid s = X(u), t = -\sqrt{g(X(u))}\}$, $C_Y = \{(u, s, t) \in \mathbb{F}_q^3 \mid s = Y(u), t = \sqrt{g(Y(u))}\}$ be the smooth projective curves. It is trivial to see there exist one-to-one map $P_X : u \mapsto (u, s \circ \rho_X(u), t \circ \rho_X(u))$ from $\mathbb{P}^1(\mathbb{F}_q)$ to $C_X(\mathbb{F}_q)$ and $P_Y : u \mapsto (u, s \circ \rho_Y(u), t \circ \rho_Y(u))$ from $\mathbb{P}^1(\mathbb{F}_q)$ to $C_Y(\mathbb{F}_q)$. Let h_X and h_Y be the projective maps on C_X and C_Y satisfying $\rho_X(u) = h_X \circ P_X(u)$ and $\rho_Y(u) = h_Y \circ P_Y(u)$. Let $g_X = P_X^{-1}$, $g_Y = P_Y^{-1}$, $S_X = g_X^{-1}(S \cup \{\infty\}) = P_X(S) \cup P_X(\infty)$, $S_Y = g_Y^{-1}(S \cup \{\infty\}) = P_Y(S) \cup P_Y(\infty)$.

To estimate $S_{f_S}(\chi)$,

$$\begin{aligned} S_{f_S}(\chi) &= \left| \sum_{u \in \mathbb{F}_q \setminus S} (f_S^* \chi)(u) + \sum_{u \in S} (f_S^* \chi)(u) \right| \\ &\leq \left| \sum_{u \in \mathbb{F}_q \setminus S} (f_S^* \chi)(u) \right| + \#S, \end{aligned}$$

we deduce as follows,

$$\begin{aligned} \left| \sum_{u \in \mathbb{F}_q \setminus S} (f_S^* \chi)(u) \right| &= \left| \sum_{\substack{P \in C_Y(\mathbb{F}_q) \setminus S_Y \\ \left(\frac{t(P)}{q}\right)=1}} (h_Y^* \psi^* \chi)(P) + \sum_{\substack{P \in C_X(\mathbb{F}_q) \setminus S_X \\ \left(\frac{t(P)}{q}\right)=-1}} (h_X^* \psi^* \chi)(P) \right| \\ &\leq \#S_Y + \#S_X + \left| \sum_{\substack{P \in C_Y(\mathbb{F}_q) \\ \left(\frac{t(P)}{q}\right)=+1}} (h_Y^* \psi^* \chi)(P) \right| + \left| \sum_{\substack{P \in C_X(\mathbb{F}_q) \\ \left(\frac{t(P)}{q}\right)=-1}} (h_X^* \psi^* \chi)(P) \right|, \end{aligned}$$

and

$$\begin{aligned} &2 \left| \sum_{\substack{P \in C_Y(\mathbb{F}_q) \\ \left(\frac{t(P)}{q}\right)=+1}} (h_Y^* \psi^* \chi)(P) \right| \\ &= \left| \sum_{P \in C_Y(\mathbb{F}_q)} (h_Y^* \psi^* \chi)(P) + \sum_{P \in C_Y(\mathbb{F}_q)} (h_Y^* \psi^* \chi)(P) \cdot \left(\frac{t(P)}{q}\right) \right. \\ &\quad \left. - \sum_{\substack{P \in C_Y(\mathbb{F}_q) \\ \left(\frac{t(P)}{q}\right)=0}} (h_Y^* \psi^* \chi)(P) \right| \\ &\leq \left| \sum_{P \in C_Y(\mathbb{F}_q)} (h_Y^* \psi^* \chi)(P) \right| + \left| \sum_{P \in C_Y(\mathbb{F}_q)} (h_Y^* \psi^* \chi)(P) \cdot \left(\frac{t(P)}{q}\right) \right| \\ &\quad + \#\{\text{roots of } g(Y(u)) = 0\}. \end{aligned}$$

From the covering $\psi \circ h_Y : C_Y \rightarrow E$, $Y(u) = s \circ \psi^{-1}(x, y)$, which implies

$$\begin{aligned} T(u) &= (a^3cy - b^3dx)u^2 - (acx - bdy)ab = 0. \\ \Leftrightarrow u^2 &= \frac{ab(acx - bdy)}{a^3cy - b^3dx}. \end{aligned}$$

Indeed, $\psi \circ h_Y$ is ramified if and only if $T(u)$ has multiple roots, which occurs when $u = 0$ or at infinity. Hence by Riemann-Hurwitz formula,

$$2g_{C_Y} - 2 = 0 + 1 + 1 = 2.$$

Hence curve C_Y is of genus 2. Similarly, C_X is also of genus 2.

Observe that

$$\deg t = [\mathbb{F}_q(s, t, u) : \mathbb{F}_q(t)] = [\mathbb{F}_q(s, t, u) : \mathbb{F}_q(s, t)] [\mathbb{F}_q(s, t) : \mathbb{F}_q(t)] = 2 \cdot 3 = 6.$$

Further more, by Theorem 3 in [29], $\left| \sum_{P \in C_Y(\mathbb{F}_q)} (h_Y^* \psi^* \chi)(P) \right| \leq (2g_{C_Y} - 2)\sqrt{q} = 2\sqrt{q}$, $\left| \sum_{P \in C_Y(\mathbb{F}_q)} (h_Y^* \psi^* \chi)(P) \cdot \left(\frac{t(P)}{q} \right) \right| \leq (2g_{C_Y} - 2 + 2 \det t)\sqrt{q} = 14\sqrt{q}$, and $g(Y(u)) = 0$ is sextic polynomial, we can derive

$$\left| \sum_{\substack{P \in C_Y(\mathbb{F}_q) \\ \left(\frac{t(P)}{q} \right) = +1}} (h_Y^* \psi^* \chi)(P) \right| \leq 8\sqrt{q} + 3.$$

And

$$\left| \sum_{\substack{P \in C_X(\mathbb{F}_q) \\ \left(\frac{t(P)}{q} \right) = -1}} (h_X^* \psi^* \chi)(P) \right|$$

has the same bound.

Hence $|S_{f_S}(x)| \leq 16\sqrt{q} + 6 + \#S_Y + \#S_X + \#S$. Note that $g(X(u)) = 0$ and $g(Y(u)) = 0$ have common roots, we can deduce that $\#S \leq 1 + 6 = 7$. Thus $\#S_X \leq 2(\#S + 1) \leq 16$. By the same reason, $\#S_Y \leq 16$. Then $|S_{f_S}(x)| \leq 16\sqrt{q} + 45$. Thus f_S is well-distributed encoding using the Theorem 3 in [29]. ■

3.4 Calculating the Density of the Image

In the case of dealing with short Weierstrass curves, Icart conjectured that the density of image $\frac{\#Im(f)}{\#E(\mathbb{F}_q)}$, is near $\frac{5}{8}$, see [16]. Fouque and Tibouchi proved this conjecture [14] using Chebotarev density theorem. Now we apply this theorem onto generalized Huff curves, and give their sizes of images of deterministic encodings.

Theorem 3 (Chebotarev, [31]). *Let K be an extension of $\mathbb{F}_q(x)$ of degree $n < \infty$ and L a Galois extension of K of degree $m < \infty$. Assume \mathbb{F}_q is algebraically closed in L , and fix some subset φ of $\text{Gal}(L/K)$ stable under conjugation. Let $s = \#\varphi$ and $N(\varphi)$ the number of places v of K of degree 1, unramified in L , such that the Artin symbol $\left(\frac{L/K}{v} \right)$ (defined up to conjugation) is in φ . Then*

$$|N(\varphi) - \frac{s}{m}q| \leq \frac{2s}{m}((m + g_L) \cdot q^{1/2} + m(2g_K + 1) \cdot q^{1/4} + g_L + nm)$$

where g_K and g_L are genera of the function fields K and L .

Theorem 4. *Let E be the generalized Huff curve over \mathbb{F}_q defined by equation $ax(y^2 - c) = by(x^2 - d)$, $abcd(a^2c - b^2d) \neq 0$, f_S is the corresponding brief SWU encoding function. Then*

$$|\#Im(f_S) - \frac{1}{2}q| \leq 4q^{1/2} + 6q^{1/4} + 27.$$

Proof. K is the function field of E which is the quadratic extension of $\mathbb{F}_q(x)$, hence $d = 2$, and by the property of elliptic curve, $g_K = 1$.

$\text{Gal}(L/K) = S_2$, hence $m = \#S_2 = 2$. φ is the subset of $\text{Gal}(L/K)$ consisting a fixed point, which is just (1)(2), then $s = 1$.

Let W be the preimage of the map ψ , $W(\mathbb{F}_q)$ be the corresponding rational points on W . By the property that ψ is one-to-one rational map, $\#Im(f_S) = \#Im(\psi^{-1} \circ f) = I_X + I_Y + I_0$, where $I_X = \#\{(s, t) \in W(\mathbb{F}_q) | \exists u \in \mathbb{F}_q, s = X(u), y = -\sqrt{g(X(u))} \neq 0\}$, $I_Y = \#\{(s, t) \in W(\mathbb{F}_q) | \exists u \in \mathbb{F}_q, s = Y(u), t = \sqrt{g(Y(u))} \neq 0\}$, $I_0 = \#\{(s, 0) \in W(\mathbb{F}_q) | g(X(u)) = 0 \text{ or } g(Y(u)) = 0\}$. It is trivial to see that $I_0 \leq 3$.

Let N_X denote the number of rational points on the curve W with an s-coordinate of the form $X(u)$ and N_Y denote the number of rational points on the curve W with an s-coordinate of the form $Y(u)$, we have

$$\begin{aligned} 2I_X &\leq N_X \leq 2I_X + I_0 \leq 2I_X + 3, \\ 2I_Y &\leq N_Y \leq 2I_Y + 3. \end{aligned}$$

$$\text{Hence } I_X + I_Y \leq \frac{1}{2}(N_X + N_Y) \leq I_X + I_Y + 3.$$

Since the place v of K of degree 1 correspond to the projective unramified points on $E(\mathbb{F}_q)$, hence $|N_X - N(\varphi)| \leq 12 + 3 = 15$, where 3 represents the number of infinite points, 12 represents the number of ramified points. Then we have

$$\begin{aligned} |N_X - \frac{1}{2}q| &\leq |N_X - N(\varphi)| + |N(\varphi) - \frac{1}{2}q| \\ &\leq 15 + (4q^{1/2} + 6q^{1/4} + 6) = 4q^{1/2} + 6q^{1/4} + 21. \end{aligned}$$

Analogously, $|N_Y - \frac{1}{2}q| \leq 4q^{1/2} + 6q^{1/4} + 21$.

Therefore, we have

$$\begin{aligned} |\#Im(f_S) - \frac{1}{2}q| &\leq |\#Im(f_S) - \frac{N_X + N_Y}{2}| + |\frac{N_X + N_Y}{2} - \frac{1}{2}q| \\ &\leq I_0 + |I_X - \frac{N_X}{2}| + |I_Y - \frac{N_Y}{2}| + (4q^{1/2} + 6q^{1/4} + 21) \\ &\leq 3 + \frac{3}{2} + \frac{3}{2} + (4q^{1/2} + 6q^{1/4} + 21) \\ &= 4q^{1/2} + 6q^{1/4} + 27. \end{aligned} \quad \blacksquare$$

4 Cube Root Encoding

4.1 Algorithm

When $q \equiv 2 \pmod{3}$ is a power of odd prime number, we give our deterministic construction $f_I : u \mapsto (x, y)$ in the following way:

Input: a, b, c, d , and $u \in \mathbb{F}_q$.

Output: A point $(x, y) \in E(\mathbb{F}_q)$.

1. $t = u^2 - a^2c - b^2d$.
2. $r = \frac{1}{2} \left(a^2b^2cd - \frac{1}{3}t^2 \right)$.
3. $s = \frac{ut}{3} + \sqrt[3]{ur^2 - \left(\frac{ut}{3}\right)^3}$.
4. $(x, y) = \left(\frac{bd(s + a^2cu)}{su + r}, \frac{ac(s + b^2du)}{su + r} \right)$.

In step 3, since $q \equiv 2 \pmod{3}$, we can efficiently calculate the the cube root by $\sqrt[3]{a} = a^{(2q-1)/3}$.

4.2 Theoretical Analysis of Time Cost

Let M , S , I and E_C represent the same as in Sect. 3.2. The cost of encoding function f_I can be estimated as follows:

1. Computing u^2 costs S . Then t can be calculated.
2. To compute r , we need S .
3. We use $S + M$ to calculate ur^2 , then use M to get ut and $S + M$ to calculate $\left(\frac{ut}{3}\right)^2$, take E_C to calculate s .
4. Finally, to calculate the inversion of $su + r$, we need $M + I$. Calculating $\frac{s}{us + r}$ and $\frac{u}{us + r}$ cost $2M$. Calculating $\frac{a^2bcd u}{su + r}$, $\frac{bds}{su + r}$, $\frac{b^2acdu}{su + r}$, $\frac{acs}{su + r}$ cost $4M$ with pre-computations.

Therefore, f_I requires $E_C + I + 4S + 10M = E + 24M$.

4.3 Properties of Cube Root Encodings

Lemma 2. Suppose $P(x, y)$ is a point on generalized Huff curve E , then equation $f_I(u) = P$ has solutions satisfying $H(u; x, y) = 0$.

When $a^4c^2 + b^4d^2 \neq a^2b^2cd$,

$$H(u; x, y) = (acx - bdy)u^4 + (2b^3d^2y - 2a^3c^2x + 4abcd(bx - ay))u^2 + 6abcd(a^2c - b^2d)u + (acx - bdy)(a^4c^2 + b^4d^2 - a^2b^2cd).$$

When $a^4c^2 + b^4d^2 = a^2b^2cd$,

$$H(u; x, y) = (acx - bdy)u^3 + (2b^3d^2y - 2a^3c^2x + 4abcd(bx - ay))u + 6abcd(a^2c - b^2d). \quad (2)$$

Proof. By the algorithm in Sect. 4.1, we have

$$\begin{aligned} \begin{cases} (xu - bd)s &= a^2bcd u - xr \\ (yu - ac)s &= ab^2cd u - yr \end{cases} \Rightarrow \frac{xu - bd}{yu - ac} = \frac{a^2bcd u - xr}{ab^2cd u - yr} \\ \Rightarrow (-bdy + acx)u^4 + (-2a^3c^2x + 4xab^2cd - 4bdya^2c + 2b^3d^2y)u^2 \\ + 6abcd(a^2c - b^2d)u + (-b^2da^2c + a^4c^2 + b^4d^2)(-bdy + acx) = 0. \end{aligned} \quad (3)$$

When $a^4c^2 + b^4d^2 = a^2b^2cd$, the constant coefficient of this equation is 0. Then eliminate u , we get (2).

Meanwhile, if $H(u; x, y) = 0$ and $(x, y) \in E$, we have

$$\begin{cases} ax(y^2 - c) = by(x^2 - d) \\ (acx - bdy) \left(b^2da^2c - \frac{(a^2c + b^2d - u^2)^2}{3} \right) = 2u((xu - bd)ab^2cd \\ - a^2bcdt(yu - ac)) \end{cases}$$

which leads to

$$(xu - bd)ab^2cd - (yu - ac)a^2bcd = (acx - bdy)(a^2b^2cd - \frac{1}{3}(a^2c + b^2d - u^2)^2)/2u,$$

from this equation and the definition of s, r , we get

$$\begin{cases} x = \frac{bd(s + a^2cu)}{ac(s + b^2du)} \\ y = \frac{su + r}{su + r} \end{cases} \Rightarrow (x, y) = f_I(u). \quad \blacksquare$$

4.4 The Genus of Curve C

Denote F by the algebraic closure of \mathbb{F}_q . We consider the graph of f_I :

$$\begin{aligned} C &= \{(x, y, u) \in E \times \mathbb{P}^1(F) \mid f_I(u) = (x, y)\} \\ &= \{(x, y, u) \in E \times \mathbb{P}^1(F) \mid H(u; x, y) = 0\}, \end{aligned}$$

which is the subscheme of $E \times \mathbb{P}^1(F)$.

Now we calculate the genus of C . In the case $a^4c^2 + b^4d^2 \neq a^2b^2cd$, the projection $g : C \rightarrow E$ is a morphism of degree 4, hence the fiber at each point of E contains 4 points. The branch points of E are points $(x, y) \in E$ where $H(u; x, y)$ has multiple roots, which means the discriminant $D = \text{disc}(H)$ vanishes at (x, y) .

By substituting $x^2 = -\frac{-axy^2 + axc - byd}{by}$ into D , it can be represented as

$$D = -\frac{16a^3(P_1(y)x + P_2(y))}{b^5y^5} \Rightarrow x = -\frac{P_2(y)}{P_1(y)},$$

where $P_1(y)$ is a polynomial of degree 10, $P_2(y)$ is a polynomial of degree 11. Substituting $x = -\frac{P_2(y)}{P_1(y)}$ into $E(x, y) = 0$, we find that y satisfies $y^{11} \cdot Q(y) = 0$, where $Q(y)$ is a polynomial of degree 12. Hence there are at most 12 branch points on E other than $(0, 0)$. It is easy to check that $(x, y) = (0, 0)$ is a branch point, since the multiplicity of $u = \infty$ is 3. If $H(u; x, y)$ has triple roots at (x, y) , we have:

$$\begin{cases} E(x, y) = 0 \\ H(u; x, y) = 0 \\ \frac{d}{dy} H(u; x, y) = 0 \\ \frac{d^2}{du^2} H(u; x, y) = 0. \end{cases} \quad (4)$$

In general cases, when $(x, y) \neq (0, 0)$, (4) has no solution, thus all 12 branch points have ramification index 2. By Riemann-Hurwitz formula, $2g_C - 2 \leq 4 \cdot (2 \cdot 1 - 2) + 12 \cdot (2 - 1) + 1 \cdot (3 - 1)$, we get $g_C \leq 8$.

In the case that $a^4c^2 + b^4d^2 = a^2b^2cd$, analogous to previous proof, we can show that g is a morphism of degree 3, D is a cubic function of u and hence has 3 different roots unless $\text{disc}(D) = 0$. By similar calculation, we find that only when y satisfies some sextic function, the point $(x, y) \in E$ is a branch point. Hence there are at most 6 branch points on E , with ramification index 2. By Riemann-Hurwitz formula, $2g_C - 2 \leq 3 \cdot (2 \cdot 1 - 2) + 6 + (3 - 1)$, we get $g_C \leq 5$.

Hence we have

Theorem 5. *If $a^4c^2 + b^4d^2 \neq a^2b^2cd$, the genus of curve C is at most 8; if $a^4c^2 + b^4d^2 = a^2b^2cd$, the genus of curve C is at most 5.*

Next, we will utilize this theorem to estimate the upper bound of the character sum for an arbitrary nontrivial character of $E(\mathbb{F}_q)$.

4.5 Estimating Character Sums on the Curve

Theorem 6. *Let f_I be the cube root encoding from \mathbb{F}_q to generalized Huff curve E , $q \equiv 3 \pmod{4}$. For any nontrivial character χ of $E(\mathbb{F}_q)$, the character sum $S_{f_I}(\chi)$ satisfies:*

$$|S_{f_I}(\chi)| \leq \begin{cases} 14\sqrt{q} + 3, & a^4c^2 + b^4d^2 \neq a^2b^2cd, \\ 8\sqrt{q} + 3, & a^4c^2 + b^4d^2 = a^2b^2cd. \end{cases} \quad (5)$$

Proof. Let $K = \mathbb{F}_q(x, y)$ be the function field of E . Recall that a point $(x, y) \in E$ is the image of u if and only if

$$H(u; x, y) = 0.$$

Then a smooth projective curve $C = \{(x, y, u) | (x, y) \in E, H(u; x, y) = 0\}$ is introduced, whose function field is the extension $L = K[u]/(H)$. By field inclusions $\mathbb{F}_q(u) \subset L$ and $K \subset L$ we can construct birational maps $g : C \rightarrow \mathbb{P}^1(\mathbb{F}_q)$ and $h : C \rightarrow E$. Then g is a bijection and $f_I(u) = H \circ g^{-1}(u)$.

Since the genus of curve C is at most 8, by Theorem 1, we have

$$|S_{f_I}(\chi) + \sum_{P \in C(\mathbb{F}_q), u(P)=\infty} \chi \circ h(P)| = \left| \sum_{P \in C(\mathbb{F}_q)} \chi \circ h(P) \right| \leq (2 \cdot 8 - 2)\sqrt{q} = 14\sqrt{q}.$$

For $(x, y) = (0, 0)$, function $H(u; x, y) = 0$ has only one finite solution, hence there exist 3 infinite solutions of u ; for other points on \bar{E} , it can be check that all solutions of $H(u; x, y) = 0$ are finite. Therefore $|\sum_{P \in C(\mathbb{F}_q), u(P)=\infty} \chi \circ h(P)| \leq 3$. Hence $|S_{f_I}(\chi)| \leq 14\sqrt{q} + 3$.

In the case that $a^4c^2 + b^4d^2 = a^2b^2cd$, it is proved that the genus of C is at most 5. Analogous to previous discussion, we have $|S_{f_I}(\chi)| \leq 8\sqrt{q} + 3$. ■

4.6 Galois Group of Field Extension

Let $K = F(x, y)$ be the function field of generalized Huff curve E , L be the function field of C . To estimate the character sum of any character of Jacobian group of E , or to estimate the size of image of f_I , we need know the structure of $Gal(L/K)$. By [31], when L/K is a quartic extension, then $Gal(L/K) = S_4$ if and only if

1. $H(u)$ is irreducible over $F(x, y)$.
2. Let $R(u)$ be the resolvent cubic of $H(u)$, then $R(u)$ is irreducible over $F(x, y)$.
3. The discriminant of $R(u)$ is not a square in $F(x, y)$.

if L/K is a cubic extension, then $Gal(L/K) = S_3$ if and only if

1. $H(u)$ is irreducible over $F(x, y)$.
2. The discriminant of $H(u)$ is not a square in $F(x, y)$.

When L/K is a quartic extension, we have to prove 3 following lemmas:

Lemma 3. *The polynomial $H(u)$ is irreducible over $F(x, y)$.*

Proof. Substitute $x = \frac{bd(s + a^2c)}{t}$ and $y = \frac{ac(s + b^2d)}{t}$ into $H(u; x, y)$, we only need to show

$$\tilde{H}(u; s, t) = \begin{cases} u^4 - (2a^2c + 2b^2d - 6s)u^2 + 6tu + (a^4c^2 + b^4d^2 - a^2b^2cd), \\ \text{when } a^4c^2 + b^4d^2 \neq a^2b^2cd, \\ u^3 + (-2a^2c - 6s - 2b^2d)u + 6v, \\ \text{when } a^4c^2 + b^4d^2 = a^2b^2cd \end{cases}$$

is irreducible over $F(s, t) = F(x, y) = K$. Let σ be the non trivial Galois automorphism in $Gal(F(s, t)/F(t))$, which maps t to $-t$, it remains to show $\tilde{H}_0(u; s, t) = \tilde{H}(u; s, t)\tilde{H}(u; s, t)^\sigma$ is irreducible over $F(t)$. Let $v = u^2$, Note that $\tilde{H}_0(u)$ can be represented as polynomial of v :

$$\begin{aligned}
J_0(v) = & v^4 + (-4ca^2 - 12s - 4db^2)v^3 + (6b^4d^2 + 6a^2b^2cd + 6a^4c^2 + 24sca^2 + \\
& 36s^2 + 24sdb^2)v^2 + (-12b^4d^2s - 4b^6d^3 - 24a^2b^2cds - 4a^6c^3 - 12a^4c^2s \\
& - 36s^3 - 36db^2s^2 - 36ca^2s^2)t + (b^4d^2 - a^2b^2cd + a^4c^2)^2, \\
& \text{if } a^4c^2 + b^4d^2 \neq a^2b^2cd,
\end{aligned} \tag{6}$$

or

$$\begin{aligned}
J_0(v) = & v^3 + (-4a^2c - 12s - 4b^2d)v^2 + 4(a^2c + 3s + b^2d)^2v \\
& - 36s(s^2 + a^2c + b^2d + b^2da^2c),
\end{aligned} \tag{7}$$

$$\text{if } a^4c^2 + b^4d^2 = a^2b^2cd.$$

From (6), by Theorem 1.2.3 in [31], if $J_0(v)$ is reducible over $F(s)$, then either it can be decomposed as

$$\begin{aligned}
J_0(v) &= (v + A)(v^3 + Bv^2 + Cv + D) \\
&= v^4 + (A + B)v^3 + (AB + C)v^2 + (AC + D)v + AD,
\end{aligned}$$

or it can be decomposed as

$$\begin{aligned}
J_0(v) &= (v^2 + Av + B)(v^2 + Cv + D) \\
&= v^4 + (A + C)v^3 + (B + AC + D)v^2 + (BC + AD)v + BD,
\end{aligned}$$

where $A, B, C, D \in F[s]$.

In the first case, note that $AD = (b^4d^2 - a^2b^2cd + a^4c^2)^2$, A and D are both constant. Since $A + B = -4ca^2 - 12s - 4db^2$, B is of degree 1. Since the coefficient of v^2 is 2, degree of C is 2, which can lead to the inference that the degree of v is also 2, a contradiction to the fact it is 3.

In the second case, B and D are constants. Hence summation of the degree of A and the degree of C equals to 2, which shows that the coefficient of v is at most 2, also a contradiction.

Then we have shown that $J_0(v)$ is irreducible over $F(s)$. Let z be a root of $H_0(u)$. Then

$$[F(s, z) : F(s)] = [F(s, z) : F(s, z^2)] \cdot [F(s, z^2) : F(s)] = 4[F(s, z) : F(s, z^2)].$$

Since $\tau \in \text{Gal}(F(s, z)/F(s, z^2))$ which maps z to $-z$ is not an identity, hence $\text{Gal}(F(s, z)/F(s, z^2)) \neq \{\iota\}$, then $[F(s, z) : F(s, z^2)] \geq 2$. Hence $[F(s, z) : F(s)] \geq 8$, which shows that $H_0(u)$ is irreducible over $F(s)$.

From (7), $J_0(v)$ is cubic, then if it is reducible, it should have a root in $F(s)$, which is factor of the constant coefficient of $J_0(v)$. However, we can confirm that such root does not exist by enumerating all the possibilities. The remaining step is similar to previous case. ■

Lemma 4. *The resolvent polynomial $R(u; x, y)$ is irreducible over $F(x, y)$.*

Proof. In the case that $H(u; x, y)$ is quartic, the resolvent cubic of $H(u)$ is

$$\begin{aligned} R(u; x, y) = & (acx - bdy)^2 u^3 + 2(acx - bdy)(-2cxb^2 ad + a^3 c^2 x + 2a^2 bcdy \\ & - b^3 d^2 y) u^2 - 4(a^4 c^2 + b^4 d^2 - a^2 b^2 cd)(acx - bdy)^2 u - 36 a^6 b^2 c^4 d^2 \\ & + 72 a^4 b^4 c^3 d^3 - 36 a^2 b^6 c^2 d^4 + 16 b^6 d^3 a^2 c^2 x^2 + 24 b^6 d^4 a^2 c y^2 \\ & - 24 b^4 d^2 x^2 a^4 c^3 - 24 a^4 b^4 c^2 d^3 y^2 + 24 a^6 b^2 c^4 d x^2 + 16 a^6 c^3 b^2 d^2 y^2 \\ & - 8 b^7 d^4 y a c x - 8 a^7 c^4 b d y x - 8 b^8 d^5 y^2 - 8 a^8 c^5 x^2 \end{aligned} \quad (8)$$

Similar to previous lemma, we only need to show $\tilde{R}(u; s, t)$, the transformation of $R(u; x, y)$ such that it is defined on $\psi^{-1}(E)$, is irreducible over $F(s, t)$. Represent x, y with variable s, t , we have

$$\begin{aligned} \tilde{R}(u; s) = & u^3 + (2ca^2 + 6s + 2db^2)u^2 + (-4b^4d^2 + 4a^2b^2cd - 4a^4c^2)u \\ & - 24a^4c^2s - 12b^2da^2cs - 24b^4d^2s - 8a^6c^3 - 36s^2a^2c - 36s^2b^2d \quad (9) \\ & - 8b^6d^3 - 36s^3 \end{aligned}$$

If $\tilde{R}(u; s)$ is reducible, it must have a degree 1 factor $u + A$, where $A \in F[s, t]$. If $A \notin F[s]$, then $(u + A)^\sigma$ is a factor of $\tilde{R}(u; s)^\sigma = \tilde{R}(u; s)$. Hence $\frac{\tilde{R}(u; s)}{(u + A)(u + A)^\sigma} \in F[s]$. Without loss of generality, we suppose $A \in F[s]$. Hence $\tilde{R}(u; s) = (u + A)(u^2 + Bu + C)$, $A, B, C \in F[s]$. In this case, $\tilde{R}(u; s)$ has a solution in $F[s]$ whose degree is 1, since when the value of u is a polynomial with degree $\neq 1$, $\tilde{R}(u; s)$ will be equal to a polynomial whose degree greater than 0. Suppose $A = Ps + Q$, $P, Q \in F$, then

$$\begin{cases} B &= 6s + 2b^2d + 2a^2c - (Ps + Q) \\ C &= -4b^4d^2 + 4a^2b^2cd - 4a^4c^2 - AB \\ AC &= -24a^4c^2s - 12b^2da^2cs - 24b^4d^2s - 8a^6c^3 - 36s^2a^2c - 36s^2b^2d \\ &\quad - 8b^6d^3 - 36s^3. \end{cases}$$

Then P and Q satisfies

$$\begin{aligned} & P^2(P - 6)s^3 + P(3QP - 12Q - 2Pb^2d - 2Pa^2c)s^2 + \\ & (3Q^2P - 6Q^2 - 4QPb^2d - 4QPa^2c - 4Pa^4c^2 - 4Pb^4d^2 + 4Pa^2b^2cd)s + \\ & Q(Q^2 - 4b^4d^2 + 4a^2b^2cd - 4a^4c^2 - 2Qb^2d - 2Qa^2c) = 0 \end{aligned} \quad (10)$$

where s is the variable. When $\text{char}(F) \geq 3$, it can be checked that solutions of P and Q do not exist. \blacksquare

Lemma 5. Let $D(x, y)$ be the discriminant of $R(u; x, y)$, then $D(x, y)$ is not a square in $F(x, y)$.

Proof. Similar to previous proof, we only need to show that

$$\tilde{D}(s, t) = D(x(s, t), y(s, t))$$

is not a square in $F(s, t)$. After simplification,

$$\begin{aligned} \tilde{D}(s, t) = & -\frac{2^7 \cdot 3^5 \cdot (abcd(a^2c - b^2d))^8}{t^8} \cdot (27s^6 - (-54a^2c - 54b^2d)s^5 - (-27a^4c^2 \\ & - 108a^2b^2cd - 27b^4d^2)s^4 + 2(a^2c + b^2d)(8b^4d^2 + 7a^2b^2cd + 8a^4c^2)s^3 \\ & + 3a^2b^2cd(8a^4c^2 - 23a^2b^2cd + 8b^4d^2)s^2 - 24a^4b^4c^2d^2(a^2c + b^2d)s \\ & - 16b^4d^2a^4c^2(a^4c^2 + b^4d^2 - a^2b^2cd)), \end{aligned} \quad (11)$$

In fact, we only need to show that $\tilde{G}(s, t) = -\frac{t^8}{2^7 \cdot 3^5 \cdot (abcd(a^2c - b^2d))^8} \tilde{D}(s, t)$ is irreducible over $F(s, t)$.

Suppose \tilde{G} is a square in $F(s, t)$, then $F(s, t) \supseteq F(s, \sqrt{\tilde{G}}) \supseteq F(s)$. Note that $[F(s, t) : F(s)] = 2$, either $F(s, \sqrt{\tilde{G}}) = F(s, t)$ or $F(s, \sqrt{\tilde{G}}) = F(s)$.

In the first case, \tilde{G} is $s(s + a^2c)(s + b^2d) = t^2$ times a square in $F(s)$. But divide \tilde{G} by $s(s + a^2c)(s + b^2d)$, the remainder vanishes if and only if $a^4c^2 + b^4d^2 - a^2b^2cd = 0$.

In the second case, \tilde{G} is a square over $F(s)$. Suppose

$$\tilde{G}(s) = \left(\sqrt{27}s^3 + Bs^2 + Cs \pm 4a^2b^2cd\sqrt{a^2b^2cd - a^4c^2 - b^4d^2} \right)^2,$$

expand the right hand side of this equation and compare its coefficients of $s^i, i = 1$ to 5 with the left hand side, and it is checked there are no $B, C \in F$ s.t the equality holds. \blacksquare

Remark: by similar method, we can also prove that when L/K is a cubic extension, $H(u; x, y)$ is irreducible over $F(x, y)$ and its discriminant is not a square in $F(x, y)$.

Summarize these lemmas, we directly deduce:

Theorem 7. *Let $K = \mathbb{F}_q(x, y)$ be the function field of E . The polynomial $H(u; x, y)$ introduced in (3) is irreducible over K , then when $a^4c^2 + b^4d^2 \neq a^2b^2cd$, its Galois group is S_4 ; when $a^4c^2 + b^4d^2 = a^2b^2cd$, its Galois group is S_3 .*

In Sect. 5.2, we will use this theorem to construct a hash function indistinguishable from random oracle.

4.7 Calculating the Density

Similar to Sect. 3.4, we apply Chebotarev density theorem to estimate the size of image of f_I .

Theorem 8. *Let E be the generalized Huff curve over \mathbb{F}_q defined by equation $ax(y^2 - c) = by(x^2 - d)$, $abcd(a^2c - b^2d) \neq 0$, f_I is the corresponding hash function defined in Sect. 4.1. Then if $a^4c^2 + b^4d^2 \neq a^2b^2cd$, we have*

$$|\#Im(f_I) - \frac{5}{8}q| \leq \frac{5}{4}(31q^{1/2} + 72q^{1/4} + 67),$$

and if $a^4c^2 + b^4d^2 = a^2b^2cd$, we have

$$|\#Im(f_I) - \frac{2}{3}q| \leq \frac{4}{3}(10q^{1/2} + 18q^{1/4} + 30).$$

Proof. K is the function field of E which is the quadratic extension of $\mathbb{F}_q(x)$, hence $d = 2$, and by the property of elliptic curve, $g_K = 1$.

In the case that $a^4c^2 + b^4d^2 \neq a^2b^2cd$, $Gal(L/K) = S_4$, hence $m = \#S_4 = 24$. φ is the subset of $Gal(L/K)$ consisting at least 1 fixed point, which are conjugates of $(1)(2)(3)(4)$, $(12)(3)(4)$ and $(123)(4)$, then $s = 1 + 6 + 8 = 15$. Since the place v of K of degree 1 correspond to the projective unramified points on $E(\mathbb{F}_q)$, hence $|\#Im(f_I) - N(\varphi)| \leq 12 + 3 = 15$, where 3 represents the number of infinite points, 12 represents the number of ramified points. Then we have

$$\begin{aligned} |\#Im(f_I) - \frac{5}{8}q| &\leq |\#Im(f_I) - N(\varphi)| + |N(\varphi) - \frac{5}{8}q| \\ &\leq 15 + \frac{5}{4}(31q^{1/2} + 72q^{1/4} + 55) \\ &= \frac{5}{4}(31q^{1/2} + 72q^{1/4} + 67). \end{aligned}$$

In the case that $a^4c^2 + b^4d^2 = a^2b^2cd$, $Gal(L/K) = S_3$, hence $m = \#S_3 = 6$. The corresponding s has the value of 4. $|\#Im(f_I) - N(\varphi)| \leq 6 + 3 = 9$, where 3 represents the number of infinite points, 6 represents the number of ramified points. Hence

$$\begin{aligned} |\#Im(f_I) - \frac{2}{3}q| &\leq |\#Im(f_I) - N(\varphi)| + |N(\varphi) - \frac{2}{3}q| \\ &\leq 9 + \frac{2}{3}(10q^{1/2} + 18q^{1/4} + 16) \\ &= \frac{2}{3}(10q^{1/2} + 18q^{1/4} + 30). \end{aligned} \quad \blacksquare$$

5 Construction of Hash Function Indifferentiable from Random Oracle

Let h be a classical hash function from messages to finite field \mathbb{F}_q , we can show that both $f_S \circ h$ and $f_I \circ h$ are one-way and collision-resistance according to the fact that each point on E has finite preimage through f_S and f_I [16]. Hence $f_S \circ h$ and $f_I \circ h$ are both hash functions mapping messages to $E(\mathbb{F}_q)$. However, since f_S and f_I are not surjective, $f_S \circ h$ and $f_I \circ h$ are easy to be distinguished from a random oracle even when h is modeled as a random oracle to \mathbb{F}_q [33]. Therefore, we introduce 2 new constructions of hash functions which are indifferentiable from a random oracle.

5.1 First Construction

Suppose $f : \mathbb{S} \rightarrow \mathbb{G}$ is a weak encoding [26] to a cyclic group \mathbb{G} , where \mathbb{S} denotes prime field \mathbb{F}_q , \mathbb{G} denotes $E(\mathbb{F}_q)$ which is of order N with generator G , $+$ denotes

elliptic curve addition. According to the proof of random oracle, we can construct a hash function $H_R : \{0, 1\}^* \rightarrow \mathbb{G}$:

$$H_R(m) = f(h_1(m)) + h_2(m)G,$$

where $h_1 : \{0, 1\}^* \rightarrow \mathbb{F}_q$ and $h_2 : \{0, 1\}^* \rightarrow \mathbb{Z}/N\mathbb{Z}$ are both classical hash functions. $H_R(m)$ is indifferentiable from a random oracle in the random oracle model for h_1 and h_2 .

We only need to show f_S, f_I are both weak encodings to prove that $H_S(m) = f_S(h_1(m)) + h_2(m)G$ and $H_I(m) = f_I(h_1(m)) + h_2(m)G$ are indifferentiable from a random oracle in the random oracle model for h_1 and h_2 . By the definition of weak encoding [26], f_S is a $\frac{2N}{q}$ -weak encoding and f_I is a $\frac{4N}{q}$ -weak encoding, both $\frac{2N}{q}$ and $\frac{4N}{q}$ are polynomial functions of the security parameter.

5.2 Second Construction

Another construction is as follows:

$$\begin{cases} H_{S'} = f_S(h_1(m)) + f_S(h_2(m)) \\ H_{I'} = f_I(h_1(m)) + f_I(h_2(m)). \end{cases}$$

We have proved that f_S, f_I are both well distributed encodings in Sects. 3.3 and 4.5. According to corollary 2 of [29], $H_{I'}$ and $H_{S'}$ are both indifferentiable from a random oracle, where h_1 and h_2 are regarded as independant random oracles with values in \mathbb{F}_q .

6 Time Comparison

When $q \equiv 3 \pmod{4}$, the key step of an encoding function is calculating square root for given element of \mathbb{F}_q . For convenience to make comparisons, we first introduce a birational map between generalized Huff curve E and short Weierstrass curve

$$E_W : t^2 = s^3 + \frac{a^2 b^2 c d - a^4 c^2 - b^4 d^2}{3} s + \frac{1}{27} (2 a^6 c^3 - 3 a^4 c^2 b^2 d - 3 a^2 c b^4 d^2 + 2 b^6 d^3),$$

via maps

$$\vartheta : E \rightarrow E_W :$$

$$(x, y) \mapsto (s, t) = \left(\frac{1}{3} \frac{2 a^2 b c d y - 2 a b^2 c d x + x a^3 c^2 - b^3 d^2 y}{a x c - b y d}, \frac{b d a c (a^2 c - b^2 d)}{a x c - b y d} \right),$$

$$\varsigma : E_W \rightarrow E :$$

$$(s, t) \mapsto (x, y) = \left(\frac{b d \left(s + \frac{2}{3} a^2 c - \frac{1}{3} b^2 d \right)}{t}, \frac{a c \left(s + \frac{2}{3} b^2 d - \frac{1}{3} a^2 c \right)}{t} \right). \quad (12)$$

Table 1. Theoretic time cost of different deterministic encodings

Encoding	Cost	Converted cost
f_S	$E_S + 2I + D + S + 10M$	$E + D + 31M$
f_U	$E_S + 2I + D + 4S + 15M$	$E + D + 39M$
f_E	$E_S + 2I + D + 4S + 10M$	$E + D + 34M$
f_I	$E_C + I + 4S + 10M$	$E + 24M$
f_Y	$E_C + 2I + 3S + 7M$	$E + 30M$
f_A	$E_C + 2I + 4S + 9M$	$E + 33M$

Table 2. NIST primes

Prime	Value	Residue (mod 3)	Residue (mod 4)
P_{192}	$2^{192} - 2^{64} - 1$	2	3
P_{384}	$2^{384} - 2^{128} - 2^{96} + 2^{32} - 1$	2	3

Table 3. Time cost (ms) of different square root methods on NIST

Prime	P_{192}	P_{384}
f_S	0.053	0.235
f_E	0.057	0.248
f_U	0.058	0.250

Therefore, we compare our encoding f_S with 2 encodings: birational equivalence ς in (12) composed with Ulas' encoding function [15], denoted by f_U ; ς composed with simplified Ulas map given by Eric Brier et al., denoted by f_E .

When $q \equiv 2 \pmod{3}$, the essential of an encoding function is calculating the cube root for elements of \mathbb{F}_q . We compare our encoding f_I with Alasha's work [19] denoted by f_A and Yu's encoding function [32] denoted by f_Y . In comparison with f_A , we let $c = \frac{1}{a}, d = \frac{1}{b}$ since Alasha only treats this special case; in comparison with f_Y , we let $c = d = 1$, since Yu's work can only be applied on classical Huff curves.

We have shown that f_S costs $E + D + 31M$, f_I costs $E + 24M$. For comparison, f_U costs $(E_S + I + 4S + 11M + D) + (I + 4M) = E + D + 39M$ by Theorem 2.3(2), [15] and the map ς in (12), while f_E costs $(E_S + I + 4S + 6M + D) + (I + 4M) = E + D + 34M$ by [14]. Yu's encoding f_Y costs $E_C + 2I + 3S + 7M = E + 30M$, Alasha's encoding f_A costs $E_C + 9M + 4S + 2I = E + 33M$ (Table 1).

We do experiments on prime field $\mathbb{F}_{P_{192}}$ and $\mathbb{F}_{P_{384}}$ (see Table 2). General Multiprecision PYthon project (GMPY2) [34], which supports the GNU Multiple Precision Arithmetic Library (GMP) [35] is used for big number arithmetic. The experiments are operated on an Intel(R) Core(TM) i5-4570, 3.20 GHz processor. We ran f_S, f_U, f_E, f_I, f_Y and f_A 1,000,000 times each, where u is randomly chosen on $\mathbb{F}_{P_{192}}$ and $\mathbb{F}_{P_{384}}$.

Table 4. Time cost (*ms*) comparison between f_I and f_A

Prime	P_{192}	P_{384}
f_I	0.053	0.233
f_A	0.061	0.252

Table 5. Time cost (*ms*) comparison between f_I and f_Y

Prime	P_{192}	P_{384}
f_I	0.052	0.233
f_Y	0.058	0.244

From the average running times listed in Table 3, f_S is the fastest among encodings which need calculate square roots. On $\mathbb{F}_{P_{192}}$, it saves 9.19 % running time compared with f_U , 7.69 % running time compared with f_E . On $\mathbb{F}_{P_{384}}$, f_S saves 5.92 % running time compared with f_U and 5.17 % running time compared with f_E . f_I is also the fastest among encodings which need to calculate cube roots. On $\mathbb{F}_{P_{192}}$, it saves 13.20 % of running time compared with f_A and 8.97 % compared with f_Y . On $\mathbb{F}_{P_{384}}$, the relevant percentages are 7.51 % and 4.40 % (see Tables 4 and 5).

7 Conclusion

We provide two constructions of deterministic encoding into generalized Huff curves over finite fields, namely, brief SWU encoding and cube root encoding. We do theoretical analysis and practical implementations to show that when $q \equiv 3 \pmod{4}$, SWU encoding is the most efficient among existed methods mapping \mathbb{F}_q into generalized Huff curve E , while cube root encoding is the most efficient one when $q \equiv 2 \pmod{3}$. For any nontrivial character χ of $E(\mathbb{F}_q)$, we estimate the upper bound of the character sums of both encodings. As a corollary, hash functions indiffereniable from random oracle are constructed. We also estimate image sizes of our encodings by applying Chebotarev density theorem.

References

1. Baek, J., Zheng, Y.: Identity-based threshold decryption. In: Bao, F., Deng, R., Zhou, J. (eds.) PKC 2004. LNCS, vol. 2947, pp. 262–276. Springer, Heidelberg (2004)
2. 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)

3. Boneh, D., Gentry, C., Lynn, B., Shacham, H.: Aggregate and verifiably encrypted signatures from bilinear maps. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 416–432. Springer, Heidelberg (2003)
4. Zhang, F., Kim, K.: ID-based blind signature and ring signature from pairings. In: Zheng, Y. (ed.) ASIACRYPT 2002. LNCS, vol. 2501, pp. 533–547. Springer, Heidelberg (2002)
5. Boyen, X.: Multipurpose identity-based signcryption. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 383–399. Springer, Heidelberg (2003)
6. Libert, B., Quisquater, J.-J.: Efficient signcryption with key privacy from gap Diffie-Hellman groups. In: Bao, F., Deng, R., Zhou, J. (eds.) PKC 2004. LNCS, vol. 2947, pp. 187–200. Springer, Heidelberg (2004)
7. Lindell, Y.: Highly-efficient universally-composable commitments based on the DDH assumption. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 446–466. Springer, Heidelberg (2011)
8. 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)
9. Boyd, C., Montague, P., Nguyen, K.: 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)
10. Jablon, D.P.: Strong password-only authenticated key exchange. SIGCOMM Comput. Commun. Rev. **26**(5), 5–26 (1996)
11. Boyko, V., MacKenzie, P.D., Patel, S.: Provably secure password-authenticated key exchange using Diffie-Hellman. In: Preneel, B. (ed.) EUROCRYPT 2000. LNCS, vol. 1807, pp. 156–171. Springer, Heidelberg (2000)
12. Shallue, A., van de Woestijne, C.E.: Construction of rational points on elliptic curves over finite fields. In: Hess, F., Pauli, S., Pohst, M. (eds.) ANTS 2006. LNCS, vol. 4076, pp. 510–524. Springer, Heidelberg (2006)
13. Skalba, M.: Points on elliptic curves over finite fields. Acta Arith. **117**, 293–301 (2005)
14. Fouque, P.-A., Tibouchi, M.: Estimating the size of the image of deterministic hash functions to elliptic curves. In: Abdalla, M., Barreto, P.S.L.M. (eds.) LATINCRYPT 2010. LNCS, vol. 6212, pp. 81–91. Springer, Heidelberg (2010)
15. Ulas, M.: Rational points on certain hyperelliptic curves over finite fields. Bull. Polish Acad. Sci. Math. **55**, 97–104 (2007)
16. Icart, T.: How to hash into elliptic curves. In: Halevi, S. (ed.) CRYPTO 2009. LNCS, vol. 5677, pp. 303–316. Springer, Heidelberg (2009)
17. Farashahi, R.R.: Hashing into hessian curves. In: Nitaj, A., Pointcheval, D. (eds.) AFRICACRYPT 2011. LNCS, vol. 6737, pp. 278–289. Springer, Heidelberg (2011)
18. Yu, W., Wang, K., Li, B., Tian, S.: About hash into montgomery form elliptic curves. In: Deng, R.H., Feng, T. (eds.) ISPEC 2013. LNCS, vol. 7863, pp. 147–159. Springer, Heidelberg (2013)
19. Alasha, T.: Constant-time encoding points on elliptic curve of different forms over finite fields (2012). http://iml.univ-mrs.fr/editions/preprint2012/files/tammam_alasha-IML_paper_2012.pdf
20. Yu, W., Wang, K., Li, B., Tian, S.: Construct hash function from plaintext to C_{34} curves. Chin. J. Comput. **35**(9), 1868–1873 (2012)
21. Huff, G.B.: Diophantine problems in geometry and elliptic ternary forms. Duke Math. J. **15**(2), 443–453 (1948)
22. Joye, M., Tibouchi, M., Vergnaud, D.: Huff’s model for elliptic curves. In: Hanrot, G., Morain, F., Thomé, E. (eds.) ANTS-IX. LNCS, vol. 6197, pp. 234–250. Springer, Heidelberg (2010)

23. Wu, H., Feng, R.: Elliptic curves in Huff model. *Wuhan Univ. J. Nat. Sci.* **17**(6), 473–480 (2011)
24. Elmegaard-Fessel, L.: Efficient Scalar Multiplication and Security against Power Analysis in Cryptosystems based on the NIST Elliptic Curves Over Prime Fields. Eprint, 2006/313. <http://eprint.iacr.org/2006/313>
25. Standards for Efficient Cryptography: Elliptic Curve Cryptography Ver. 5 (1999). <http://www.secg.org/drafts.html>
26. 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)
27. Ciss, A.A., Sow, D.: On a new generalization of Huff curves. *Cryptology ePrint Archive: Report 2011/580* (2011). <http://eprint.iacr.org/2011/580.pdf>
28. Devigne, J., Joye, M.: Binary Huff curves. In: Kiayias, A. (ed.) *CT-RSA 2011*. LNCS, vol. 6558, pp. 340–355. Springer, Heidelberg (2011)
29. Farashahi, R.R., Fouque, P.-A., Shparlinski, I.E., Tibouchi, M., Voloch, J.F.: Indifferentiable deterministic hashing to elliptic and hyperelliptic curves. *Math. Comp.* **82**, 491–512 (2013)
30. Farashahi, R.R., Shparlinski, I.E., Voloch, J.F.: On hashing into elliptic curves. *J. Math. Cryptol.* **3**(4), 353–360 (2009)
31. Roman, S.: *Field Theory*. Graduate Texts in Mathematics, vol. 158, 2nd edn. Springer, New York (2011)
32. Wei, Y., Wang, K., Li, B.: Constructing hash function from plaintext to Huff curves. *J. Univ. Sci. Tech. China* (10), 835–838 (2014)
33. Tibouchi, M.: Impossibility of surjective icart-like encodings. In: Chow, S.S.M., Liu, J.K., Hui, L.C.K., Yiu, S.M. (eds.) *ProvSec 2014*. LNCS, vol. 8782, pp. 29–39. Springer, Heidelberg (2014)
34. GMPY2, General Multiprecision Python (Version 2.2.0.1). <https://gmpy2.readthedocs.org>
35. GMP: GNU Multiple Precision Arithmetic Library. <https://gmplib.org/>

Information Security and Cryptology

11th International Conference, Inscrypt 2015, Beijing,
China, November 1-3, 2015, Revised Selected Papers

Lin, D.; Wang, X.; Yung, M. (Eds.)

2016, XIV, 490 p. 102 illus., Softcover

ISBN: 978-3-319-38897-7