



Almost Injective and Invertible Encodings for Jacobi Quartic Curves

Xiuxiu Li^{1,2,3}, Wei Yu^{1,2,3}(✉), Kunpeng Wang^{1,3}, and Luying Li^{1,3}

¹ State Key Laboratory of Information Security, Institute of Information Engineering, CAS, Beijing, China

yuwei@iie.ac.cn, yuwei_1_yw@163.com

² State Key Laboratory of Cryptology, P. O. Box 5159, Beijing 100878, China

³ School of Cyberspace Security, University of Chinese Academy of Sciences, Beijing, China

Abstract. This paper introduces a novel encoding scheme for hashing values from the finite field \mathbb{F}_p to points on Jacobi quartic curves. These curves possess efficient group law and are immune to timing attacks. The proposed encoding scheme achieves almost injective and invertible mappings of the input values into Jacobi quartic curves. When $p \equiv 3 \pmod{4}$, our encoding saves $2\mathbf{I} + \mathbf{D} - 8\mathbf{M} - 4\mathbf{S}$ compared to existing methods. This improvement amounts to approximately 50% on average when compared to existing methods. The encoding scheme can be used in a variety of cryptographic applications that rely on elliptic curves, such as identity-based encryption schemes and private set intersection protocols.

Keywords: Jacobi quartic curves · Injective and invertible encoding · Inverse map · B -well-distributed

1 Introduction

Since the introduction of elliptic curves into cryptography by Miller [25] and Koblitz [23], elliptic curve cryptography has become a major branch in the field of cryptography. The group structure of elliptic curves has become a focus of research under the impetus of cryptography. All elliptic curves are considered to have the Weierstrass form, which is parametrized by a cubic equation $y^2 = x^3 + ax + b$. In the real domain, the addition law on a Weierstrass curve can be described by three points where the line intersects the curve, with the unit element point being the infinity point.

To achieve better efficiency in various protocols, many different forms of elliptic curves have been studied in elliptic curve cryptography, including Edwards form, Montgomery form, and Jacobi model. The Jacobi quartic is one of the two Jacobi models. Compared with the Montgomery form and Edwards forms, the extended Jacobi quartic form includes more curves. Billet and Joye [4] showed

Supported by the National Natural Science Foundation of China (No. 62272453, U1936209, 61872442, and 61502487).

that every elliptic curve containing a point of order two could be written as a curve in Jacobi quartic form and provided the birational map between Weierstrass elliptic curves with a point of order two and the Jacobi quartic curves.

In [22], Hisil, Wong, Carter, and Dawson provided doubling formulae on Jacobi quartic curves that involves two field multiplications and five field squarings. According to Bernstein and Lange Explicit-Formulas Database [3], it is one of the fastest doubling formulae without loss of information. Meanwhile, [22] shows that the Jacobi quartic curves are competitive with twisted Edwards curves in variable-singlepoint-variable-single-scalar multiplication. Jacobi quartic curves can also be employed in pairing calculations [13, 14, 33].

Hashing into elliptic curves is an important procedure that encodes arbitrary values into points on elliptic curves over a finite field. This process is widely employed in elliptic curve cryptosystems, including password-authenticated key exchanges [7], identity based encryption [5], and Boneh-Lynn-Shacham signatures [6, 30].

The “try and increment” method, also referred to as “sample and reject” and “hint and pick”, is the first map in hashing into elliptic curves. This encoding involves repeatedly sampling the value of x and testing whether x could be the x -coordinates of a point on elliptic curves until a satisfactory x has been found.

Shallue and Woestijne employed Skalba’s equality [27] proposed Shallue-van de Woestijne map [26]. Three candidates of the x -coordinates were proposed, and at least one of them could be the x -coordinates of a point. Ulas [29] and Brier et al. [8] subsequently simplified the Shallue-van de Woestijne map. Their simplifications are referred to as SWU encoding and brief/simplified SWU encoding. Recently Wahby and Boneh further sped up this model and constructed the mapping for the BLS-12 381 curve in CHES2019 [30].

In order to avoid censorship, Bernstein, Hamburg, Krasnova, and Lange proposed Elligator 1 and Elligator 2 encodings [2]. Both Elligator 1 and Elligator 2 are almost injective and invertible maps, with Elligator 2 can be employed on more curves. The injectivity and invertibility of Elligator 2 allow it to make the points indistinguishable from random strings at less cost. The IETF [18] prefers the Elligator 2 encodings over other encodings and has speed up the Elligator 2 encoding on Montgomery curves and Edwards curves.

Boneh and Franklin put forwarded a deterministic mapping for a specific type of supersingular curve over a finite field \mathbb{F}_p with $p \equiv 2 \pmod{3}$. Icart generalized Boneh and Franklin’s method for Weierstrass curves over a finite field \mathbb{F}_p with $p \equiv 2 \pmod{3}$. The SWU encodings, Elligator encodings, and Icart encoding have been extended and adapted into many other forms in the literature [12, 17, 20, 21, 31, 32]. Additionally, there has been significant research on the security of hashing into elliptic curves [1, 8, 11, 16, 19, 24, 28].

The current Jacobi quartic curves encoding and decoding process is inefficient due to the computational expense of the Elligator 2 algorithm used to deterministically encode arbitrary values into Jacobi quartic curves from \mathbb{F}_p with $p \equiv 3 \pmod{4}$. Additionally, the resulting curve points are not uniformly distributed. To address these issues, this paper proposes a new encoding method that constructs

almost-injective and invertible encodings for Jacobi quartic curves. The proposed encoding method is based on Elligator 2 and employs projective coordinates to reduce the number of inversions required in our mapping. An inverse map is provided to ensure that the resulting points are indistinguishable from uniform random strings. In theory, our new encoding method achieved a 50% reduction in time compared to previous square root encoding. This paper provides a solution to the inefficiency of the current Jacobi quartic curve encoding and decoding process, and demonstrates the effectiveness of our proposed method through experimentation.

The paper is organized as follows: Sect. 2 provides necessary background information for encoding, Sect. 3 presents the theorems and proofs about the map and inverse map, Sect. 4 introduces our injective and invertible encoding for Jacobi quartic curves, Sect. 5 compares our encoding’s time complexity to previous works, and Sect. 6 concludes this paper.

2 Background

Let K be a field of characteristic not equal to 2. The Jacobi quartic curves are elliptic curves of the form:

$$y^2 = (1 - x^2)(1 - k^2x^2), \quad k \neq \pm 1.$$

where k is a nonzero field element. Chudnovsky and Chudnovsky [10] introduced a variant of the Jacobi quartic curve in the form

$$y^2 = x^4 + ax^2 + b,$$

which they used to construct inversion-free addition formulas. Billet and Joye further extended the Jacobi quartic form to

$$y^2 = dx^4 + 2ax^2 + 1$$

where $a, d \in K$, $a, d \neq 0$, and $\Delta = 256d(a^2 - d)^2 \neq 0$.

Any elliptic curve in Weierstrass form $E : y^2 = x^3 + ax + b$, that has a point $(\theta, 0)$ of order two, is birational equivalent to the curve $E_{a',d'} : y^2 = d'x^4 + 2a'x^2 + 1$ in Jacobi quartic form, where $d = -(3\theta^2 + 4a)/16$ and $a' = 3\theta/4$. The birational map from E to $E_{a',d'}$ is given by

$$\begin{aligned} \phi : E &\rightarrow E_{a',d'} \\ (x, y) &\mapsto \left(\frac{x - \theta}{y}, \frac{(2x + \theta)(x - \theta)^2 - y^2}{y^2} \right), \end{aligned} \tag{1}$$

where $(x, y) \neq \mathcal{O}, (\theta, 0)$ and \mathcal{O} denotes the point at infinity. $\phi(\mathcal{O}) = (0, 1)$, $\phi(\theta, 0) = (0, -1)$.

Group Law. On Jacobi quartic curve, the $(0, 1)$ is the identity point, and $(0, -1)$ is a point of order two. The negative of a point (x, y) is $(-x, y)$. Given two points (x_1, y_1) and (x_2, y_2) on the curve $E_{a,d}$, their sum is the point (x_3, y_3) with

$$x_3 = \frac{x_1y_2 + y_1x_2}{1 - dx_1^2x_2^2},$$

$$y_3 = \frac{(y_1y_2 + 2ax_1x_2)(1 + dx_1^2x_2^2) + 2dx_1x_2(x_1^2 + x_2^2)}{1 - dx_1^2x_2^2},$$

where a and d are parameters of the curve. Hisil et al. have also shown that y_3 can be alternatively represented as followings:

$$y_3 = \frac{(y_1y_2 - 2ax_1x_2)(x_1^2 + x_2^2) - 2x_1x_2(1 + dx_1^2x_2^2)}{(x_1y_2 - y_1x_2)^2},$$

$$y_3 = \frac{(y_1y_2 + 2ax_1x_2)(x_1y_2 - y_1x_2) + 2(x_2y_2 - x_1y_1)}{(x_1y_2 - y_1x_2)(1 - dx_1^2x_2^2)},$$

and

$$y_3 = \frac{x_1y_1(2 + 2ax_1^2 - y_1^2) - x_2y_2(2 + 2ax_2^2 - y_2^2)}{(x_1y_2 - y_1x_2)(1 - dx_1^2x_2^2)}.$$

3 The Map and the Inverse Map

First we employ the Elligator 2 to construct the deterministic encoding from \mathbb{F}_p with $p \equiv 3 \pmod{4}$ to Jacobi quartic curves. Inspired by [30] and [18], we adopt the projective form to improve the efficiency of the encoding. The encoding process first maps the values to the curve $Y'^2Z' = X'(X'^2 - 4aX'Z' + (4a^2 - 4d)Z'^2)$ over \mathbb{F}_p , and then maps them to the Jacobi quartic curve $Y^2Z^2 = dX^4 + 2aX^2Z^2 + Z^4$.

Let u be an element in \mathbb{F}_p for the encoding. Generally, we select u from set

$$S = \{u \in \mathbb{F}_p \mid u \neq 0, \pm 1, 16a^2u^2 \neq 4(a^2 - 4d)(1 + u^2)^2\}.$$

Here S corresponds to the set R given by Theorem 5 in [2]. With this selection, we can derive the following theorem.

Theorem 1. *Let p be a odd prime power satisfies $p \equiv 3 \pmod{4}$. Let $u \in S$, where S is defined above. Let*

$$D = 1 - u^2$$

$$U = 64a^3 - 64a^3D + 16a(a^2 - 4d)D^2$$

$$V = D^3$$

$$R = (UV)(UV^3)^{\frac{p-3}{4}}$$

If $VR^2 = U$, let

$$(X', Y', Z') = (-4a, R'D, D),$$

where R' is the even one in $\{R, -R\}$. Else let

$$(X', Y', Z') = (-4a(1 - D), R'D, D),$$

where R' is the odd one in $\{uR, -uR\}$. Then the followings can be obtained:

- (1) $DUVRX'Y'Z' \neq 0, D \neq 0, R \neq 0, V \neq 0, U \neq 0, Y' \neq 0, Z' \neq 0, X' \neq 0,$
- (2) $(X' : Y' : Z')$ is a point on curve $Y'^2Z' = X'(X'^2 - 4aX'Z' + (4a^2 - 4d)Z'^2)$.
- (3) $(X : Y : Z) = (2Y'Z' : X'^2 - 4(a^2 - d)Z'^2 : X'^2 - 4aX'Z' + (4a^2 - 4d)Z'^2)$ is a point on Jacobi quartic curve $Y^2Z^2 = dX^4 + 2aX^2Z^2 + Z^4$.
- (4) Let $g(x) = x(x^2 - 4ax + 4(a^2 - d))$ and denote $\sqrt{\cdot}$ as the principle square root, i.e., the even one in the two square roots, then if $X' = 4a, R' = \sqrt{g(X'/Z')}$. If $X' = -4a(1 - D), R' = -\sqrt{g(X'/Z')}$.

Proof. Let $A = -4a, B = 4(a^2 - 4d)$. Then

$$4a/D = -A/(1 - u^2) \triangleq v,$$

$$U/V = v^3 + Av^2 + Bv,$$

and

$$\sqrt{U/V} = R.$$

Inserting these expressions in Theorem 5 in [2], choose the principle square root as the even one, then (1) and (2) can be obtained. (3) is derived from (2) and the birational map in [22] §2.3.2. (4) is obvious.

Theorem 2. Let $\varphi = \psi \circ \tau$ be map provided in Theorem 1, where τ is the map from S to points on curve $Y'^2Z' = X'(X'^2 - 4aX'Z' + (4a^2 - 4d)Z'^2)$ and ψ is the map from curve $Y'^2Z' = X'(X'^2 - 4aX'Z' + (4a^2 - 4d)Z'^2)$ to Jacobi quartic curve $Y^2Z^2 = dX^4 + 2aX^2Z^2 + Z^4$. Then

- (1) For any $u \in S$, if $\varphi(u)$ and $\varphi(u')$ denote the same projective point, then $u = \pm u'$.
- (2) If $(X : Y : Z) \in \varphi(S)$ then the following element \bar{u} of S is defined and $\varphi(\bar{u}) = (X : Y : Z)$:

$$\bar{u} = \begin{cases} \sqrt{\frac{Z^2 - aX^2 + YZ}{Z^2 + aX^2 + YZ}}, & \text{if } \frac{4YZ^2 + 4Z^3 + 4aX^2Z}{X^3} \text{ is even,} \\ \sqrt{\frac{Z^2 + aX^2 + YZ}{Z^2 - aX^2 + YZ}}, & \text{if } \frac{4YZ^2 + 4Z^3 + 4aX^2Z}{X^3} \text{ is odd.} \end{cases} \quad (2)$$

Proof. Let $u, X', Y', Z', X, Y,$ and Z be defined as in Theorem 1. (1) By the birational map ψ and its inverse map ψ' given in [22] §2.3.2, $\varphi(u) = \varphi(u')$ follows that $\tau(u) = \tau(u')$. According to Theorem 7 in [2], $u' = \pm u$. (Note that when $u \in S, X', Y', Z', X, Z$ are not zero.) (2) can be derived by the birational map ψ' and Theorem 7 in [2].

4 Hash into Jacobi Quartic Curves

4.1 B-Well-Distributed Property

Recall the definition of B -well-distributed in [15].

Definition 1 ([15]). *Let X be a smooth projective curve over a finite field \mathbb{F}_p , J its Jacobian, f a function $\mathbb{F}_p \rightarrow X(\mathbb{F}_p)$ and B a positive constant. We say that f is B -well-distributed if for any nontrivial character χ of $J(\mathbb{F}_p)$, the character sum $S_f(\chi)$ satisfies the following equation:*

$$|S_f(\chi)| \leq B\sqrt{p}.$$

In the following, we introduce the basic theorem for the B -well-distributed property.

Theorem 3 (Theorem 7 in [15]). *Let $h : \tilde{X} \rightarrow X$ be a nonconstant morphism of curves, and χ be any nontrivial character of $J(\mathbb{F}_p)$, where J is the Jacobian of X . Assume that h does not factor through a nontrivial unramified morphism $Z \rightarrow X$. Then*

$$\left| \sum_{P \in \tilde{X}(\mathbb{F}_p)} \chi(X(P)) \right| \leq (2\tilde{g} - 2)\sqrt{p}$$

where \tilde{g} is the genus of \tilde{X} . Furthermore, if p is odd and φ is a nonconstant rational function on \tilde{X} , then

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

where (\cdot) denotes the Legendre symbol.

Theorem 4. *Let φ be the encoding defined in Theorem 1, $p \equiv 3 \pmod{4}$. For any nontrivial character χ of $E(\mathbb{F}_p)$, the character sum $S_\varphi(\chi)$ satisfies:*

$$|S_\varphi(\chi)| \leq 16\sqrt{p} + 43.$$

Proof. Let S, R', D be defined as in Sect. 3. Let $\bar{S} = \mathbb{F}_p \setminus S$. Then for any $u \in S$, the following equivalents are established:

$$\begin{aligned} X' &= -4a \Leftrightarrow u^2 - \omega = 0 \\ X' &= -4a(1 - D) \Leftrightarrow u^2 - \frac{1}{\omega} = 0 \end{aligned}$$

where $\omega = (1 + ax^2 - y)/(1 - ax^2 + y)$. The coordinates $x = X/Z$ and $y = Y/Z$ are from equation (2). Let two coverings $h_j : C_j \rightarrow E$, $j = 1, 2$ be the smooth projective curves whose function field are the extensions of $\mathbb{F}_p(x, y)$ defined by $u^2 - \omega = 0$ and $u^2 - 1/\omega = 0$. Then the parameter u is a rational function on

each of the C_j giving rise to morphisms $g_j : C_j \rightarrow \mathbb{P}^1$, such that any point in $\mathbb{A}^1(S)$ has exactly two preimages in $C_j(\mathbb{F}_p)$ for one of $j = 1, 2$, and none in the other. It follows that h_j is ramified if and only if $u = 0$ or $u = \infty$. Hence by Riemann-Hurwitz formula,

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

Hence curves C_j are of genus 2. Denote the map from C_j to \mathbb{F}_p that maps $P = (u, x, y)$ to R' by $\bar{\varphi}$. We have $\deg \bar{\varphi} = 6$. Let $S_j = h_j^{-1}(S \cup \{\infty\})$

$$\begin{aligned} \left| \sum_{u \in S} \chi(\varphi(u)) \right| &= \left| \sum_{\substack{P \in C_0(\mathbb{F}_p) \setminus S_0, \\ \left(\frac{R'}{p}\right)=1}} \chi(h_0(P)) + \sum_{\substack{P \in C_1(\mathbb{F}_p) \setminus S_1, \\ \left(\frac{R'}{p}\right)=-1}} \chi(h_1(P)) \right| \\ &\leq \left| \sum_{\substack{P \in C_0(\mathbb{F}_p), \\ \left(\frac{R'}{p}\right)=1}} (h_0^* \chi)(P) \right| + \left| \sum_{\substack{P \in C_1(\mathbb{F}_p), \\ \left(\frac{R'}{p}\right)=-1}} (h_1^* \chi)(P) \right| + \#S_0 + \#S_1 \end{aligned}$$

And we have

$$\begin{aligned} 2 \left| \sum_{\substack{P \in C_0(\mathbb{F}_p), \\ \left(\frac{R'}{p}\right)=1}} (h_0^* \chi)(P) \right| &= \left| \sum_{P \in C_0(\mathbb{F}_p)} (h_0^* \chi)(P) + \sum_{P \in C_0(\mathbb{F}_p)} (h_0^* \chi)(P) \cdot \left(\frac{R'}{p}\right) \right. \\ &\quad \left. - \sum_{\substack{P \in C_0(\mathbb{F}_p), \\ \left(\frac{R'}{p}\right)=0}} (h_2^* \chi)(P) \right| \\ &\leq \left| \sum_{P \in C_0(\mathbb{F}_p)} (h_0^* \chi)(P) \right| + \left| \sum_{P \in C_0(\mathbb{F}_p)} (h_0^* \chi)(P) \cdot \left(\frac{R'}{p}\right) \right| \\ &\quad + \#\{u \mid R' = 0\} \end{aligned}$$

By Theorem 3, we have

$$\left| \sum_{P \in C_j(\mathbb{F}_p)} (h_j^* \chi)(P) \right| \leq (2g_{C_j} - 2)\sqrt{p} = 2\sqrt{p}$$

and

$$\left| \sum_{P \in C_j(\mathbb{F}_p)} (h_j^* \chi)(P) \cdot \left(\frac{R'}{p}\right) \right| \leq (2g_{C_j} - 2 + 2 \deg \bar{\varphi})\sqrt{p} = 14\sqrt{p}$$

Since for all $u \in S$, $R' \neq 0$, and $\#\bar{S} \leq 1+7 = 8$. We have $\#S_j \leq 2(\#\bar{S}+1) \leq 18$. It follows that

$$|S_\varphi(\chi)| = \left| \sum_{u \in S} \chi(\varphi(u)) \right| \leq 16\sqrt{p} + 43$$

4.2 Indifferentiable from Random Oracle

According to Theorem 3 in [15], our encoding φ described in the previous paragraph is a well-distributed encoding. Furthermore, Corollary 2 in [15] states that if h_1 and h_2 are two independent random oracle hash functions, then the following construction:

$$H(m) = \varphi(h_1(m)) + \varphi(h_2(m))$$

is indifferentiable from a random oracle.

4.3 Points Indistinguishable from Uniform Random Strings

Since our encoding is almost injective and invertible, points on Jacobi quartic curves can be encoded as strings to avoid censorship by the inverse map given in Theorem 2. Based on the B -well-bounded property of our encoding, it is easy to prove that our encoding is (d, B) -well-bounded. Therefore, the Elligator Square method can be applied to our encoding to make the resulting points indistinguishable from uniform random strings. However, it should be noted that the Elligator Square method is time-consuming. For further details on the (d, B) -well-bounded property and the Elligator Square method, please refer to [28].

5 Time Complexity

In the rest of this work, we use **I** denotes field inversion, **E** denotes field exponentiation, **M** denotes field multiplication, and **S** denotes field squarings for simplification. Then the cost of our almost-injective and invertible encoding can be summarized as follows:

1. Compute u^2 require one **S**, and it is enough for D .
2. Compute D^2 and $V = D^3$ require **M** + **S**.
3. $2\mathbf{M}$ are required in the computation of U since $64a^3$ and $16a(a^2 - 4d)$ can be pre-computed.
4. Computing R as $R = (U \cdot V) \cdot ((UV) \cdot V^2)^{(q-3)/4}$ costs **E** + $3\mathbf{M}$ + **S**
5. Checking whether $VR^2 = U$ costs **M** + **S**
6. (X', Y', Z') can be computed within $3\mathbf{M}$.
7. Computing $X'^2, Z'^2, 2X'Z' = (X' + Z')^2 - X'^2 - Z'^2$. And then compute (X, Y, Z) by $Y', Z', X'^2, Z'^2, 2X'Z'$ and pre-computed values $4a$ and $4(a^2 - d)$. This procedure costs $3\mathbf{M}$ + $3\mathbf{S}$ in total to obtain X, Y and Z .

To sum up, $\overline{\varphi}$ costs $\mathbf{E} + \mathbf{13M} + \mathbf{7S}$. And the inverse map φ^{-1} can be computed as follows:

1. Computing Z^2, X^2, X^3, aX^2 and YZ in $\mathbf{3M} + \mathbf{2S}$.
2. Employing Montgomery’s technique compute the inversion $s = 1/(X^3(Z^2 + aX^2 + YZ)(Z^2 - aX^2 + YZ))$ by $\mathbf{I} + \mathbf{2M}$.
3. Using $\mathbf{3M} + \mathbf{S}$ to check the parity of $4YZ^2 + 4aX^2Z + 4Z^3/X^3 = 4Z(Z^2 + aX^2 + YZ)^2(Z^2 - aX^2 + YZ)s$.
4. If the parity is even, let $(U, V) = (Z^2 + aX^2 + YZ, Z^2 - aX^2 + YZ)$, and else let $(U, V) = (Z^2 - aX^2 + YZ, Z^2 + aX^2 + YZ)$. This step needs no cost.
5. $\bar{u} = \sqrt{U/V} = (UV)(UV^3)^{(p-3)/4}$ is obtained in $\mathbf{E} + \mathbf{3M} + \mathbf{S}$.

Let f_A denote the encoding proposed by Alasha [1], f_{YS} and f_{YI} denotes the encoding proposed by Yu et al. [31], which are based on brief SWU encoding and Icart encoding respectively. Table 1 shows the theoretical time complexity of these encodings compared with ours. Specifically, when the finite field \mathbb{F}_p satisfying $p \equiv 3 \pmod 4$, our encoding φ saves $\mathbf{2I} + \mathbf{D} - \mathbf{8M} - \mathbf{4S}$ compared to f_{YS} . According to Bernstein and Lange Explicit-Formulas Database [3], if the ratio $\mathbf{I/M} = \mathbf{100}$, our encoding on Jacobi quartic curves is more than 50% faster than f_{YS} when $p \equiv 3 \pmod 4$.

Table 1. Theoretical time cost of different encodings on Jacobi quartic curves

Encodings	Field condition	Costs
f_A	$p \equiv 2 \pmod 3$	$\mathbf{E} + \mathbf{2I} + \mathbf{8M} + \mathbf{3S}$
f_{YI}	$p \equiv 2 \pmod 3$	$\mathbf{E} + \mathbf{I} + \mathbf{9M} + \mathbf{5S}$
f_{YS}	$p \equiv 3 \pmod 4$	$\mathbf{E} + \mathbf{2I} + \mathbf{5M} + \mathbf{3S} + \mathbf{D}$
φ	$p \equiv 3 \pmod 4$	$\mathbf{E} + \mathbf{13M} + \mathbf{7S}$

To compare the efficiency of our encoding and f_{YS} , both running on the finite field \mathbb{F}_p with $p \equiv 3 \pmod 4$, we conducted experiments using SageMath for big number arithmetic. The experiments were performed on a 12th Gen Intel(R) Core(TM) i7-12700H 2.30 GHz processor, with each encoding running 1,000,000 times, where u was randomly chosen on \mathbb{F}_{P256} and \mathbb{F}_{P384} . The primes $P256$ and $P384$ were selected as the NIST primes [9]. The experiments results are presented in Table 2.

Table 2. Time cost (μs) comparison on \mathbb{F}_{P256} and \mathbb{F}_{P384}

Encodings	P256	P384
f_{YS}	87	152
φ (ours)	45	75

According to above experimental results, our encoding is 48.3% faster than f_{YS} on field $\mathbb{F}_{P_{256}}$ and 50.7% faster on field $\mathbb{F}_{P_{384}}$. The experimental results are consistent with the previous theoretical results.

6 Conclusion

This paper presents an almost-injective and invertible encoding scheme for Jacobi quartic curves using Elligator 2 encoding and projective coordinates. The proposed encoding reduces the number of inversions required for mapping, resulting in a faster algorithm compared to previous square root encoding techniques. The inverse map is also provided to ensure that the encoded points are indistinguishable from uniform random strings. Our results show that the proposed encoding technique outperforms previous methods by reducing computation time by approximately 50%. Additionally, the decoding of points on elliptic curves into finite fields is also addressed in this paper.

References

1. Alasha, T.: Constant-time encoding points on elliptic curve of different forms over finite fields (2012)
2. Bernstein, D., Hamburg, M., Krasnova, A., Lange, T.: Elligator: Elliptic-curve points indistinguishable from uniform random strings, pp. 967–980 (2013). <https://doi.org/10.1145/2508859.2516734>
3. Bernstein, D., Lange, T.: Explicit-formulas database (2020). <http://hyperelliptic.org/EFD/>
4. Billet, O., Joye, M.: The Jacobi model of an elliptic curve and side-channel analysis. In: Fossorier, M., Høholdt, T., Poli, A. (eds.) AAEECC 2003. LNCS, vol. 2643, pp. 34–42. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-44828-4_5
5. 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). https://doi.org/10.1007/3-540-44647-8_13
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). https://doi.org/10.1007/3-540-45682-1_30
7. Boyko, V., MacKenzie, P., 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). https://doi.org/10.1007/3-540-45539-6_12
8. 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). https://doi.org/10.1007/978-3-642-14623-7_13
9. Chen, L., Moody, D., Regenscheid, A., Randall, K.: Draft nist special publication 800-186 recommendations for discrete logarithm-based cryptography: elliptic curve domain parameters. Technical report, National Institute of Standards and Technology (2019)

10. Chudnovsky, D., Chudnovsky, G.: Sequences of numbers generated by addition in formal groups and new primality and factorization tests. *Adv. Appl. Math.* **7**(4), 385–434 (1986). [https://doi.org/10.1016/0196-8858\(86\)90023-0](https://doi.org/10.1016/0196-8858(86)90023-0)
11. Chávez-Saab, J., Rodríguez-Henrquez, F., Tibouchi, M.: SwiftEC: Shallue-van de Woestijne indifferentiable function to elliptic curves (2022). <https://eprint.iacr.org/2022/759>
12. Diarra, N., Sow, D., Khilil, A.Y.O.C.: On indifferentiable deterministic hashing into elliptic curves. *Eur. J. Pure Appl. Math.* **10**, 363–391 (2017)
13. Doss, S., Kaondera-Shava, R.: An optimal Tate pairing computation using Jacobi quartic elliptic curves. *J. Comb. Optim.* **35**(4), 1086–1103 (2018). <https://doi.org/10.1007/s10878-018-0257-y>
14. Duquesne, S., Fouotsa, E.: Tate pairing computation on Jacobi’s elliptic curves. In: Abdalla, M., Lange, T. (eds.) *Pairing 2012*. LNCS, vol. 7708, pp. 254–269. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36334-4_17
15. Farashahi, R., Fouque, P.A., Shparlinski, I., Tibouchi, M., Voloch, J.: Indifferentiable deterministic hashing to elliptic and hyperelliptic curves. *IACR Cryptol. ePrint Arch.* **2010**, 539 (2010). <https://doi.org/10.1090/S0025-5718-2012-02606-8>
16. Farashahi, R.R., Shparlinski, I.E., Voloch, J.F.: On hashing into elliptic curves. *J. Math. Cryptol.* **3**(4), 353–360 (2009)
17. Farashahi, R.R.: Hashing into hessian curves. In: Nitaĵ, A., Pointcheval, D. (eds.) *AFRICACRYPT 2011*. LNCS, vol. 6737, pp. 278–289. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-21969-6_17
18. Faz-Hernández, A., Scott, S., Sullivan, N., Wahby, R.S., Wood, C.A.: Hashing to elliptic curves. Internet-Draft draft-irtf-cfrg-hash-to-curve-13, Internet Engineering Task Force (2021). <https://datatracker.ietf.org/doc/html/draft-irtf-cfrg-hash-to-curve-13>
19. 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). https://doi.org/10.1007/978-3-642-14712-8_5
20. Fouque, P.-A., Tibouchi, M.: Indifferentiable hashing to Barreto–Naehrig curves. In: Hevia, A., Neven, G. (eds.) *LATINCRYPT 2012*. LNCS, vol. 7533, pp. 1–17. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-33481-8_1
21. He, X., Yu, W., Wang, K.: Hashing into generalized huff curves. In: Lin, D., Wang, X.F., Yung, M. (eds.) *Inscrypt 2015*. LNCS, vol. 9589, pp. 22–44. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-38898-4_2
22. Hisil, H., Wong, K.K.-H., Carter, G., Dawson, E.: Jacobi quartic curves revisited. In: Boyd, C., González Nieto, J. (eds.) *ACISP 2009*. LNCS, vol. 5594, pp. 452–468. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-02620-1_31
23. Koblitz, N.: Elliptic curve cryptosystems. *Math. Comput.* **48**, 203–209 (1987)
24. Koshelev, D.: Indifferentiable hashing to ordinary elliptic F_q -curves of $j = 0$ with the cost of one exponentiation in F_q . *Designs Codes Cryptogr.* **90** (2022). <https://doi.org/10.1007/s10623-022-01012-8>
25. Miller, V.S.: Use of elliptic curves in cryptography. In: Williams, H.C. (ed.) *CRYPTO 1985*. LNCS, vol. 218, pp. 417–426. Springer, Heidelberg (1986). https://doi.org/10.1007/3-540-39799-X_31
26. 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). https://doi.org/10.1007/11792086_36

27. Skalba, M.: Points on elliptic curves over finite fields. *Acta Arith.* **117**(3), 293–301 (2005)
28. Tibouchi, M.: Elligator squared: uniform points on elliptic curves of prime order as uniform random strings. In: Christin, N., Safavi-Naini, R. (eds.) *FC 2014*. LNCS, vol. 8437, pp. 139–156. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45472-5_10
29. Ulas, M.: Rational points on certain hyperelliptic curves over finite fields. *arXiv Number Theory* (2007)
30. Wahby, R.S., Boneh, D.: Fast and simple constant-time hashing to the BLS12-381 elliptic curve. *IACR Trans. Cryptogr. Hardw. Embed. Syst.* **2019**, 154–179 (2019)
31. Yu, W., Wang, K., Li, B., He, X., Tian, S.: Hashing into Jacobi quartic curves. In: Lopez, J., Mitchell, C.J. (eds.) *ISC 2015*. LNCS, vol. 9290, pp. 355–375. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23318-5_20
32. Yu, W., Wang, K., Li, B., He, X., Tian, S.: Deterministic encoding into twisted Edwards curves. In: Liu, J.K., Steinfeld, R. (eds.) *ACISP 2016*. LNCS, vol. 9723, pp. 285–297. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-40367-0_18
33. Zhang, F., Li, L., Wu, H.: Faster pairing computation on Jacobi quartic curves with high-degree twists. In: Yung, M., Zhu, L., Yang, Y. (eds.) *INTRUST 2014*. LNCS, vol. 9473, pp. 310–327. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-27998-5_20