Efficient constant-time hashing to elliptic curves $y^2 = x^3 + b$ provided that b is a quadratic residue

Dmitrii Koshelev [1]

Versailles Laboratory of Mathematics, Versailles Saint-Quentin-en-Yvelines University

Abstract. Let $\mathbb{F}_q$ be a finite field and $E_b\colon y^2 = x^3 + b$ be an ordinary elliptic $\mathbb{F}_q$-curve of j-invariant 0 such that $\sqrt{b} \in \mathbb{F}_q$. In particular, this condition is fulfilled for the curve BLS12-381 and for one of sextic twists of the curve BW6-761 (in both cases $b = 4$). These curves are very popular in pairing-based cryptography. The article provides an efficient constant-time hashing $h\colon \mathbb{F}_q \to E_b(\mathbb{F}_q)$ of an absolutely new type for which $q/6 \leqslant \#\mathrm{Im}(h)$. The main idea of our hashing consists in extracting in $\mathbb{F}_q$ a cubic root instead of a square root as in the well known (universal) SWU hashing and in its simplified analogue. Besides, the new hashing can be implemented without quadratic and cubic residuosity tests (as well as without inversions) in $\mathbb{F}_q$. Thus in addition to the protection against timing attacks, h is much more efficient than the SWU hashing, which generally requires to perform two quadratic residuosity tests in $\mathbb{F}_q$. For instance, in the case of BW6-761 this allows to avoid at least approximately $2 \cdot 761 \approx 1500$ field multiplications.

Key words: constant-time implementation, cubic residue symbol and cubic roots, hashing to elliptic curves, pairing-based cryptography.

Introduction

Many protocols of *pairing-based cryptography* [1] use a mapping $h\colon \mathbb{F}_q \to E_b(\mathbb{F}_q)$ called *hashing* [1, §8] such that $\#\mathrm{Im}(h) = \Theta(q)$, where $q \approx \#E_b(\mathbb{F}_q)$ according to the Hasse inequality [2, Theorem V.1.1]. In other words, h should cover most $\mathbb{F}_q$-points of E_b. In addition, the hashing h is called *constant-time* if the computation time of its value is independent of an input argument. Almost all hashings used in practice have this property in order to be protected against *timing attacks* [1, §8.2.2, §12.1.1].

There is the so-called *SWU hashing* [1, §8.3.4], which is applicable to any elliptic $\mathbb{F}_q$-curve (i.e., not necessarily of $j = 0$). However it generally requires the computation of two Legendre symbols (i.e., quadratic residuosity tests) in $\mathbb{F}_q$. Unfortunately, this operation (as well as the inversion one in $\mathbb{F}_q$) is vulnerable to timing attacks.

There is also the *simplified SWU hashing* (see, e.g., [3, §2]), which, on the contrary, does not contain Legendre symbols at all. However, at the moment it cannot be applied to some curves E_b, including the sextic twist (with $b = 4$) of the curve BW6-761 from [4]. The simplified SWU hashing sometimes can be constructed by means of a vertical $\mathbb{F}_q$-isogeny (the Wahby–Boneh approach [5]) or $\mathbb{F}_{q^2}$-isogeny (the Koshelev approach [3]) $\psi\colon E \to E_b$ of

[1] web page: https://www.researchgate.net/profile/Dimitri_Koshelev
email: dishport@yandex.ru

This work was supported by a public grant as part of the FMJH project

small degree d, where $j(E) \neq 0$. For example, the curve BLS12-381 (also with $b = 4$) [5, §2.1] benefits from a vertical $\mathbb{F}_q$-isogeny of degree $d = 11$.

In our opinion, the main disadvantage of using such isogenies is decreasing the cardinality $\#\mathrm{Im}(h)$ (not to mention increasing the computation time of h) with increasing degree d, even though this correlation is linear. More precisely, if $\ker(\psi) \subset E(\mathbb{F}_q)$ (what is true in the case of BLS12-381), then the image cardinality of the Wahby–Boneh hashing equals $\approx 3q/(8d)$. Indeed, that of the simplified SWU hashing equals $\approx 3q/8$ according to [6, Propositon 4]. At the same time, the isogeny ψ (by definition of its degree) maps exactly d points of $E(\mathbb{F}_q)$ to only one point of $E_b(\mathbb{F}_q)$. In particular, for $d = 11$ we have $8d/3 = 88/3 \approx 29.3$.

The given work continues the previous ones [3], [7], [8] of the author. Therefore let us not repeat a detailed overview of the given scientific field for the sake of brevity. In this article it is represented a new efficient constant-time hashing $h\colon \mathbb{F}_q \to E_b(\mathbb{F}_q)$ provided that $\sqrt{b} \in \mathbb{F}_q$. We establish that $q/6 \leqslant \#\mathrm{Im}(h)$, which is at least $44/9 \approx 4.9$ times more than the image cardinality of the Wahby–Boneh hashing (for $d = 11$).

Our approach is based on using the elliptic $\mathbb{F}_q(t)$-curve $\mathcal{E}$ (1) and its $\mathbb{F}_q(t)$-point φ (2), where $\mathbb{F}_q(t)$ denotes the rational function field in one variable t over the constant field $\mathbb{F}_q$. Moreover, φ has very simple formulas, hence h can be implemented quite efficiently. In order not to complicate the text we do not explain why the particular surface $\mathcal{E}$ was taken and how the formulas of φ were derived. We can just add that it was used a certain cubic $\mathbb{F}_q$-twist of the generalized Kummer threefold of Calabi–Yau type from [9, §1.3] and the theory of the Mordell–Weil lattices of elliptic $\mathbb{F}_q(t)$-curves.

Obtained results

Consider the $\mathbb{F}_q(t)$-curve given as the intersection of two quadratic $\mathbb{F}_q(t)$-surfaces

$$\mathcal{E}\colon \begin{cases} y_1^2 - b = bt^3(y_0^2 - b), \\ y_2^2 - b = b^2t^3(y_0^2 - b) \end{cases} \quad \subset \quad \mathbb{A}^3_{(y_0,y_1,y_2)}. \tag{1}$$

Lemma 1 ([10]). *$\mathcal{E}$ is an elliptic $\mathbb{F}_q(t)$-curve of j-invariant*

$$256 \cdot \frac{\left(b^4t^6 - b^2(b+1)t^3 + b^2 - b + 1\right)^3}{\left(b(b-1)(b^2t^3 - 1)(bt^3 - 1)\right)^2}.$$

In other words, $\mathcal{E} \subset \mathbb{A}^4_{(y_0,y_1,y_2,t)}$ is an *elliptic $\mathbb{F}_q$-surface* (see, e.g., [11, Chapter III]), whose the elliptic fibration is the projection to t. In [12, §2.5.4] it is described how to transform $\mathcal{E}$ into Weierstrass form.

Theorem 1 ([10]). *$\mathcal{E}$ has the $\mathbb{F}_q(t)$-point (i.e., $\mathbb{F}_q$-section)*

$$\varphi := \begin{cases} y_0(t) := \sqrt{b} \cdot \dfrac{-b^2(b-1)^2 \cdot t^6 - 2b(b+1) \cdot t^3 + 3}{den}, \\[2ex] y_1(t) := \sqrt{b} \cdot \dfrac{b^2(b+3)(b-1) \cdot t^6 - 2b(b-1) \cdot t^3 + 1}{den}, \\[2ex] y_2(t) := \sqrt{b} \cdot \dfrac{b^2(3b+1)(b-1) \cdot t^6 - 2b(b-1) \cdot t^3 - 1}{den}, \end{cases} \tag{2}$$

where

$$den := b^2(b-1)^2\cdot t^6 - 2b(b+1)\cdot t^3 + 1.$$

Moreover,

$$y_0(t) - y_1(t) + y_2(t) = \sqrt{b}, \qquad by_1^2(t) - y_2^2(t) = b(b-1).$$

For the frequent case $b = 4$ we obtain

$$\varphi = \begin{cases} y_0(t) := 2\cdot\dfrac{-2^4 3^2\cdot t^6 - 2^3 5\cdot t^3 + 3}{den}, \\ y_1(t) := 2\cdot\dfrac{2^4 3\cdot 7\cdot t^6 - 2^3 3\cdot t^3 + 1}{den}, \\ y_2(t) := 2\cdot\dfrac{2^4 3\cdot 13\cdot t^6 - 2^3 3\cdot t^3 - 1}{den}, \end{cases} \qquad \text{where} \qquad den = 2^4 3^2\cdot t^6 - 2^3 5\cdot t^3 + 1.$$

We everywhere assume that $q \equiv 1 \pmod 3$, i.e., $\omega := \sqrt[3]{1} \in \mathbb{F}_q^*$, where $\omega \neq 1$. In particular, by virtue of [2, Example V.4.4] this is true if E_b is an ordinary (i.e., non-supersingular) curve. As is well known, only such curves are applied in pairing-based cryptography. For $a \in \mathbb{F}_q^*$ denote by $\left(\frac{a}{q}\right)_3 := a^{(q-1)/3}$ the *cubic residue symbol*, which is a group homomorphism $\mathbb{F}_q^* \to \{\omega^i\}_{i=0}^2$.

Lemma 2 ([13, Remark 2.3])**.** *An element $a \in \mathbb{F}_q^*$ is a cubic residue if and only if $\left(\frac{a}{q}\right)_3 = 1$. Moreover, in this case*

$$\sqrt[3]{a} = \begin{cases} \text{[14, Proposition 1]} & \text{if} \quad q \equiv 1 \pmod 9 \text{ and } q \not\equiv 1 \pmod{27}, \\ a^{-(q-4)/9} = a^{(8q-5)/9} & \text{if} \quad q \equiv 4 \pmod 9, \\ a^{(q+2)/9} & \text{if} \quad q \equiv 7 \pmod 9. \end{cases}$$

It is well known (see, e.g., [8, Remark 1]) that in the case $\left(\frac{b}{q}\right)_3 = 1$ the simplified SWU hashing can be used. Therefore without loss of generality we will assume that $\left(\frac{b}{q}\right)_3 = \omega$.

We would like to explain how $\varphi\colon \mathbb{A}_t^1 \dashrightarrow \mathcal{E} \subset \mathbb{A}^4_{(y_0,y_1,y_2,t)}$ gives a constant-time hashing $h\colon \mathbb{F}_q \to E_b(\mathbb{F}_q)$. It will be considered the cases $q \equiv 4 \pmod 9$ (occurs for BW6-761) and $q \equiv 10 \pmod{27}$ (occurs for BLS12-381). The cases $q \equiv 7 \pmod 9$ and $q \equiv 19 \pmod{27}$ are processed in a similar way.

Letting $g_i := y_i^2 - b$ for $i \in \{0,1,2\}$, we get $\mathcal{E}\colon \{g_j = b^j t^3 g_0\}_{j=1}^2$. It is obvious that $\{\left(\frac{g_i}{q}\right)_3\}_{i=0}^2 = \{\omega^i\}_{i=0}^2$ whenever $g_i, t \in \mathbb{F}_q^*$. We denote by U and V respectively the domain of definition and the image for φ. Besides, $n \in \{0,1,2\}$ will be the position number of an element $t \in \mathbb{F}_q^*$ in the set $\left\{\omega^i t\right\}_{i=0}^2$ ordered with respect to some order in $\mathbb{F}_q^*$. For example, if q is a prime, then this can be the usual numerical one.

The case $q \equiv 4 \pmod 9$**.** Under this assumption

$$\left(\frac{\omega}{q}\right)_3 = \omega^{(q-1)/3} = \omega^{(q-4)/3}\cdot\omega = \omega^{3(q-4)/9}\cdot\omega = \omega.$$

Let $\theta := g_0^{(8q-5)/9}$ and $c_j := \sqrt[3]{(b/\omega)^j} \in \mathbb{F}_q^*$ for $j \in \{1,2\}$. We obtain

$$g_j = b^j t^3 g_0 = (c_j \theta t)^3 \qquad \text{if} \qquad \theta^3 = \omega^j g_0, \text{ i.e., } \left(\frac{g_0}{q}\right)_3 = \omega^{3-j}.$$

Consider the auxiliary map

$$h'\colon V(\mathbb{F}_q) \to E_b(\mathbb{F}_q), \qquad (y_0, y_1, y_2, t) \mapsto \begin{cases} \big(\omega^n \cdot \theta,\ y_0\big) & \text{if} \quad \theta^3 = g_0, \\ \big(\omega^n \cdot c_1 \theta t,\ y_1\big) & \text{if} \quad \theta^3 = \omega g_0, \\ \big(\omega^n \cdot c_2 \theta t,\ y_2\big) & \text{if} \quad \theta^3 = \omega^2 g_0. \end{cases}$$

The element θ can be computed with the cost of one exponentiation in $\mathbb{F}_q$. Indeed,

$$(u/v)^{(8q-5)/9} = u^{(8q-5)/9} \cdot v^{(q-4)/9} = u^3 (u^8 v)^{(q-4)/9} \tag{3}$$

for any $u, v \in \mathbb{F}_q^*$. Since

$$\theta^3 = g_0^{-(q-4)/3} = g_0^{q-1-(q-4)/3} = g_0^{(2q+1)/3} = g_0^{2(q-1)/3} \cdot g_0,$$

the map h' is well defined everywhere on $V(\mathbb{F}_q)$.

The case $q \equiv 10 \pmod{27}$**.** Take any $\zeta := \sqrt[9]{1} \in \mathbb{F}_q^*$ such that $\zeta^3 = \omega$. In this case

$$\left(\frac{\zeta}{q}\right)_3 = \zeta^{(q-1)/3} = \omega^{(q-1)/9} = \omega^{(q-10)/9} \cdot \omega = \omega^{3(q-10)/27} \cdot \omega = \omega.$$

Let $\theta := g_0^{(2q+7)/27}$ and $c_j := \sqrt[3]{(b/\zeta)^j} \in \mathbb{F}_q^*$ for $j \in \{1,2\}$. Given $i \in \{0,1,2\}$ we obtain

$$g_j = b^j t^3 g_0 = (c_j \theta t)^3 / \omega^i \qquad \text{if} \qquad \theta^3 = \omega^i \zeta^j g_0, \text{ i.e., } \left(\frac{g_0}{q}\right)_3 = \omega^{3-j}.$$

Consider the auxiliary map

$$h'\colon V(\mathbb{F}_q) \to E_b(\mathbb{F}_q), \qquad (y_0, y_1, y_2, t) \mapsto \begin{cases} \big(\omega^n \cdot \theta / \zeta^i,\ y_0\big) & \text{if} \quad \exists i\colon \theta^3 = \omega^i g_0, \\ \big(\omega^n \cdot c_1 \theta t / \zeta^i,\ y_1\big) & \text{if} \quad \exists i\colon \theta^3 = \omega^i \zeta g_0, \\ \big(\omega^n \cdot c_2 \theta t / \zeta^i,\ y_2\big) & \text{if} \quad \exists i\colon \theta^3 = \omega^i \zeta^2 g_0. \end{cases}$$

The element θ can be computed with the cost of one exponentiation in $\mathbb{F}_q$. Indeed,

$$\begin{aligned} (u/v)^{(2q+7)/27} = u^{(2q+7)/27} \cdot v^{q-1-(2q+7)/27} = u^{(2q+7)/27} \cdot v^{(25q-34)/27} = \\ = u \cdot u^{2(q-10)/27} \cdot v^3 v^{5(5q-23)/27} = uv^8 (u^2 v^{25})^{(q-10)/27}. \end{aligned} \tag{4}$$

for any $u, v \in \mathbb{F}_q^*$. Since

$$\theta^3 = g_0^{(2q+7)/9} = g_0^{2(q-1)/9} \cdot g_0,$$

the map h' is well defined everywhere on $V(\mathbb{F}_q)$.

In both cases, for any $t \in \mathbb{F}_q$ we can put

$$h(t) := \begin{cases} (h' \circ \varphi)(t) & \text{if} \quad t \in U(\mathbb{F}_q), \\ (0:1:0) & \text{if} \quad t \notin U(\mathbb{F}_q). \end{cases}$$

We emphasize that in the definition of h' (a fortiori, φ) the cubic residue symbol (in other words, cubic residuosity test) does not appear. In turn, by returning the value of h in (weighted) projective coordinates, we entirely avoid inversions in the field. It is also worth noting that the constants c_j are found once, using precalculations. Finally, by virtue of the formulas (3), (4) we obtain

Remark 1. *On the set $U(\mathbb{F}_q)$ the new hashing h is computed in constant time, namely in that of one exponentiation in $\mathbb{F}_q$.*

Theorem 2. *We have $q/6 \leqslant \#\mathrm{Im}(h)$.*

Proof. First, suppose that $h(t) = \pm P_0$, where $P_0 := (0, \sqrt{b})$. Then $\theta(t) = g_0(t) = 0$ or $t = 0$. In the first case, $y_0(t) = \pm\sqrt{b}$. More precisely,

$$y_0(t) = \sqrt{b} \;\Leftrightarrow\; t^3 = \frac{\pm 1}{b(b-1)}, \qquad y_0(t) = -\sqrt{b} \;\Leftrightarrow\; t^3 = \frac{1}{b(b+1)}.$$

In the second case, $y_0(0) = 3\sqrt{b}$, $g_0(0) = 8b$, and hence $\left(\frac{g_0(0)}{q}\right)_3 = \omega$. Since $y_2(0) = -\sqrt{b}$, we have $h(0) = -P_0$. As a result, $\#h^{-1}(P_0) \leqslant 6$ and $\#h^{-1}(-P_0) \leqslant 4$.

Now take $t \in U(\mathbb{F}_q)$ such that $h(t) \neq \pm P_0$. For definiteness let the value $g_0(t)$ is a cubic residue in $\mathbb{F}_q$. Then for $t' \in \mathbb{F}_q$ from the collision $h(t) = h(t')$ it follows that exists $i \in \{0,1,2\}$ such that $y_0(t) = y_i(t')$. Every given equation has at most 6 solutions in $\mathbb{F}_q$ with respect to t'. However, in the definition of h' the (non-zero) x-coordinate is multiplied by ω^n, hence we can take into account only 2 solutions (with the different cubic powers).

In turn, the set $\mathbb{F}_q \setminus U(\mathbb{F}_q)$ contains only $\mathbb{F}_q$-roots of the polynomial *den*, that is

$$t \notin U(\mathbb{F}_q) \;\Leftrightarrow\; t^3 = \frac{(\sqrt{b} \pm 1)^2}{b(b-1)^2}.$$

Thus for every point from $E_b(\mathbb{F}_q)$ its inverse image under h contains at most 6 elements of the field $\mathbb{F}_q$ and our theorem is proved. □

Acknowledgements. The author expresses his deep gratitude to his scientific advisor M. Tsfasman.

References

[1] N. El Mrabet, M. Joye, *Guide to Pairing-Based Cryptography*, Cryptography and Network Security Series, Chapman and Hall/CRC, New York, 2016.

[2] J. Silverman, *The arithmetic of elliptic curves*, Graduate Texts in Mathematics, **106**, Springer, New York, 2009.

[3] D. Koshelev, *Hashing to elliptic curves of $j = 0$ and quadratic imaginary orders of class number* 2, https://www.researchgate.net/profile/Dimitri_Koshelev, 2020.

[4] Y. El Housni, A. Guillevic, *Optimized and secure pairing-friendly elliptic curves suitable for one layer proof composition*, ePrint IACR 2020/351.

[5] R. Wahby, D. Boneh, "Fast and simple constant-time hashing to the BLS12-381 elliptic curve", *IACR Transactions on Cryptographic Hardware and Embedded Systems*, **2019(4)**, 154–179.

[6] P. Fouque, M. Tibouchi, "Estimating the size of the image of deterministic hash functions to elliptic curves", *Security and Cryptology*, Latincrypt 2010, **6212**, ed. M. Abdalla, P. Barreto, Springer, Berlin, 2010, 81–91.

[7] D. Koshelev, *Hashing to elliptic curves of j-invariant* 1728, ePrint IACR 2019/1294.

[8] D. Koshelev, *Hashing to elliptic curves of $j = 0$ and Mordell–Weil groups*, arXiv:2005.08336, 2020.

[9] K. Oguiso, T. Truong, "Explicit examples of rational and Calabi–Yau threefolds with primitive automorphisms of positive entropy", *Journal of Mathematical Sciences, the University of Tokyo*, **22** (2015), 361–385.

[10] D. Koshelev, *Magma code*, https://github.com/dishport/Efficient-constant-time-hashing-to-elliptic-curves-of-j-0-provided-that-b-is-a-quadratic-residue, 2020.

[11] J. Silverman, *Advanced topics in the arithmetic of elliptic curves*, Graduate Texts in Mathematics, **151**, Springer, New York, 1994.

[12] L. Washington, *Elliptic curves: number theory and cryptography*, Discrete Mathematics and Its Applications, Chapman & Hall, London, 2008.

[13] A. Dudeanu, G.-R. Oancea, S. Iftene, "An x-coordinate point compression method for elliptic curves over $\mathbb{F}_p$", Inter. Symp. on Symb. and Num. Algor. for Scientific Comp., 2010, 65–71.

[14] G. Cho et al., "New cube root algorithm based on the third order linear recurrence relations in finite fields", *Designs, Codes and Cryptography*, **75(3)** (2015), 483–495.