Hashing-friendly elliptic curves

Dimitri Koshelev[1⋆]

University of Lleida, Department of Mathematics, Catalonia, Spain
dimitri.koshelev@gmail.com

Abstract. This article aims to consider batch hashing to elliptic curves. The given kind of hash functions found numerous applications in elliptic curve cryptography. In practice, a hash-to-curve function is often evaluated at a time by the same entity at many different inputs. It turns out that under certain mild conditions simultaneous evaluation can be carried out several times faster than separate ones. In this regard, the article introduces a new class of elliptic curves over finite fields, more appropriate for multiple hashing to them. Moreover, two explicit hashing-friendly Montgomery/twisted Edwards curves (of ≈ 128 security bits) have been generated: one of CM discriminant -7, i.e., a GLV-friendly curve and one of huge CM discriminant, i.e., a CM-secure curve. The new elliptic curves are intentionally covered by so-called Klein's and Bring's curves of geometric genera 3 and 4, respectively. The latter are well studied in various algebraic geometry contexts, although they have not yet been (reasonably) applied in cryptography to the author's knowledge. Such a mathematical complication is justified, since conventional curves (from existing standards or of j-invariants 0, 1728) are seemingly less efficient for batch hashing.

Keywords: Bring's curve · elliptic curve cryptography · hashing to elliptic curves · Klein's quartic · root extraction in finite fields

1 Introduction

According to [21,23], there is still no (essential) progress in number factorization on today's quantum computers. That is why pre-quantum cryptography, including *ECC (elliptic curve cryptography)* [18], does not lose its relevance. In particular, further research on acceleration of its various cryptographic primitives is encouraged. For example, many ECC schemes demand deterministic (a.k.a. constant-time) *hashing to elliptic curves*. Constructing such hash functions has a rich 20-year history starting with a breakthrough in the independent papers [41,43] and coming to some logical end in the recent one [31] (cf. [22]), which generalizes *SwiftEC* [15]. Besides them, one can benefit from the corresponding RFC [16], albeit it does not contain the latest results on the topic. Meanwhile, the present paper proposes an idea on how to bypass the encountered optimization barrier, reanimating the given research direction.

⋆ https://www.linkedin.com/in/dimitri-koshelev

Throughout the text, E will stand for an elliptic curve over a finite field $\mathbb{F}_q$ of large characteristic. A hash-to-curve function has the form $\mathcal{H}\colon \{0,1\}^* \to E(\mathbb{F}_q)$. As usual, the running time of $\mathcal{H}$ must depend not on a secret input but only on its length. This property will be assumed everywhere by default. In the real world, there are often a lot of strings $s_i \in \{0,1\}^*$ for which an entity computes the hash values $\mathcal{H}(s_i)$. The most practical (but not unique) use cases are perhaps the widespread *aggregate BLS (Boneh–Lynn–Shacham) signature* [9] (or its numerous modifications) and *multiset hashing* [37]. However, as far as the author knows, nobody has yet suggested (explicitly) batch techniques for the primitive under consideration. In comparison, the last source promotes elliptic curves in characteristic 2 (a.k.a. binary curves) as the most efficient for (multiple) hashing, not to mention the relative cheapness of binary finite fields for hardware. Nevertheless, elliptic curves in large characteristic (a.k.a. prime curves) remain much more preferable in practice for a variety of reasons related to functionality and safety (a good survey is [1]).

Recall that the bottleneck of almost any hash-to-curve function is a certain root $\sqrt[d]{\cdot} \in \mathbb{F}_q$ with $d \geqslant 2$, not taking into account the possible presence (when $2 \mid d$) of Legendre symbols $\left(\frac{\cdot}{q}\right)$. But the latter are inexpensive as is widely recognized nowadays [3]. Moreover, a degree-d root is often expressed via one exponentiation in $\mathbb{F}_q$. For instance, this is well known when $d = 2$ and $8 \nmid q-1$ or when $\gcd(d, q-1) = 1$, that is, d is invertible modulo $q-1$. Assume that $d = \prod_{i=0}^{k-1} d_i$, where $2 \leqslant d_i < d_{i+1}$ and $\gcd(d_i, d_j) = 1$ whenever $i \neq j$. For laconicity, it is worth putting $\overline{d} := (d_0, \cdots, d_{k-1})$. Borrowing the insight from [29, Section 2], we will prove that, for any tuple $\overline{p} = (p_0, \cdots, p_{k-1}) \in \mathbb{F}_q^k$, the roots $\sqrt[d_i]{p_i}$ can be rationally expressed via the elements of $\overline{p}$ and $\sqrt[d]{p}$, where p is an appropriate monomial in these elements. In addition, $\sqrt[d]{p} = \sqrt[d_{k-1}]{\cdots \sqrt[d_0]{p}}$, hence the d-th root is indistinguishable from a field exponentiation if all the d_i-th roots are. Thus, a single exponentiation is sufficient instead of k ones and so the optimum is only achieved when d_i are prime powers (as in the main part of this paper). To the best of the author's knowledge, this clever trick is not widespread in the ECC community, although it should be known among RSA (Rivest–Shamir–Adleman) fans due to Fiat [17].

In the literature, many hash-to-curve functions $\mathcal{H}$ are decomposed into two components $\eta\colon \{0,1\}^* \to \mathbb{F}_q$ and $h\colon \mathbb{F}_q \to E(\mathbb{F}_q)$. To be definite, we will deal exclusively with such $\mathcal{H}$. Here, η is an auxiliary hash function, which is universal and supposedly cheap, whereas the map h basically defines $\mathcal{H}$. Now, we are given k hash functions $\mathcal{H}_i = h_i \circ \eta$ to the same group $E(\mathbb{F}_q)$ such that $r_i = \sqrt[d_i]{p_i}$ is the unique root involved in h_i and the value p_i rationally depends on $x \in \mathbb{F}_q$. It is important to stress that p_i may vary (when $2 \mid d_i$) depending on Legendre-symbol outputs, but constantly $p_i \in \mathbb{F}_q(x)$. To put it another way, there is the identity $h_i(x) = \phi_i(x, r_i)$, where $\phi_i\colon C_{d_i} \to E$ is an $\mathbb{F}_q$-cover (varying also on $\left(\frac{\cdot}{q}\right)$) from the *superelliptic* $\mathbb{F}_q$*-curve* $C_{d_i}\colon y^{d_i} = p_i(x)$. Note that the rational maps $p_i(x)$, $\phi_i(x, y)$ may be undefined for some input $\mathbb{F}_q$-points. Fortunately, it is highly unlikely to meet such an input once η satisfies a good mixing property. Alternatively, one can extend h_i to the whole $\mathbb{F}_q$ in any way. Since the degenerate

field elements are very rare with respect to the size of q in ECC, any statistical indicators of $\mathcal{H}_i$ are not (significantly) affected.

It is tempting to introduce one more hash function $\mathcal{H}(s) := \mathcal{H}_{\mu(s)}(s)$, where the map $\mu\colon \{0,1\}^* \to \mathbb{Z}/k$ is of the trivial form $\mu(s) := s \bmod k$. Since the parameter k is little at least in the examples below and strings s are relatively long in practice, the map μ produces on $\mathbb{Z}/k$ a distribution close to uniform (cf. [12, Lemma 14], [16, Section 5]). In other words, μ is *regular* in the sense of [12, Definition 4]. Consequently, for a large number n of pairwise different strings s_i, one deals with $\approx n/k$ calls of the same atomic hash function $\mathcal{H}_j$. After renumbering, let's assume that $\mu(s_i) \equiv i \bmod k$ essentially without loss of generality. Then, the sequence $\mathcal{H}(s_1), \cdots, \mathcal{H}(s_n)$ can be computed at the cost of $\approx n/k$ roots of degree d by evaluating simultaneously $\mathcal{H}(s_i), \cdots, \mathcal{H}(s_{i+k-1})$. If we naturally suppose that every participating root $\sqrt[d]{\cdot}$ is nothing but one exponentiation in $\mathbb{F}_q$, then we obtain acceleration in k times, that is, $\approx n(k-1)/k$ field exponentiations are economized.

In the new notation, [29, Section 2.1] investigates the case $\overline{d} = (2,3)$, that is, $d = 6$. Whenever $3 \mid q-2$ and regardless of E, this case allows constructing the hash function $\mathcal{H}_0(s) + \mathcal{H}_1(s)$ *indifferentiable from a random oracle* [12, Section 2.2]. By the way, that is why the paper [31] (for which $d = \overline{d} = 2$) restricts itself on the opposite scenario $3 \mid q-1$, proving thereby the existence of one-root (up to two Legendre symbols) indifferentiable hashing for any pair $E/\mathbb{F}_q$. Many constructions with one $\sqrt{\cdot}$ (even without $\left(\frac{\cdot}{q}\right)$ for j-invariant $\neq 0$) exist for the role of $\mathcal{H}_0$, although the majority of them (with the varying parts p_0, ϕ_0) are grounded on *Shallue–van de Woestijne*'s [41] and *Skałba*'s [43] seminal works. In turn, $\mathcal{H}_1$ is instantiated by *Icart*'s hash function [24] with one $\sqrt[3]{\cdot}$. Besides, ordinary (a.k.a. non-supersingular) $\mathbb{F}_q$-curves of j-invariant 0 with an order-3 point also admit the setting $d = 6$ according to *Koshelev* [26], despite the condition $3 \mid q-1$ (and $27 \nmid q-1$ for finding $\sqrt[3]{\cdot}$ as one exponentiation). In particular, all *BLS (Barreto–Lynn–Scott) curves* [4] of arbitrary embedding degree are relevant (see, e.g., [29, Lemma 1]). On the other hand, BLS-12 curves are the most popular for deploying pairing-based protocols.

In a nutshell, a certain trigonal $\mathbb{F}_q$-curve $C_3\colon y^3 = p(x)$ of geometric genus 7 underlies Icart's hash function. Curiously, such a covering curve takes place for every elliptic curve E and its derivation is pretty easy. Denote by $\phi\colon C_3 \to E$ the corresponding $\mathbb{F}_q$-cover and by $pr_x\colon C_3 \to \mathbb{A}^1_x$ the projection to the x-coordinate. Once $3 \mid q-2$, there is always a single $\mathbb{F}_q$-root $y = \sqrt[3]{p(x)}$ given $x \in \mathbb{F}_q$. To put it another way, pr_x restricted on $C_3(\mathbb{F}_q)$ is a bijection onto $\mathbb{F}_q$. Therefore, the main hashing component is $h = \phi \circ pr_x^{-1}\colon \mathbb{F}_q \to E(\mathbb{F}_q)$. More broadly, nothing prevents from picking in the described method another superelliptic $\mathbb{F}_q$-curve $C_d\colon y^d = p(x)$ such that $\gcd(d, q-1) = 1$ (cf. [29, Question 1]). The derived hash function is said to be *Icart-like*. However, we cannot expect that the general E enjoys such a covering curve C_d provided that $\gcd(d,6) = 1$.

The present paper concentrates on the instances $\overline{d} = (2,3,7)$, that is, $d = 42$ as well as $\overline{d} = (2,3,5)$, that is, $d = 30$. They are materialized due to higher-genus *Klein's K* [32] (cf. [14]) and *Bring's B curves* [11], famous very well in the alge-

braic geometry community but not yet in the cryptography one. These curves are characterized by the fact that they are unique (over the algebraic closure $\overline{\mathbb{F}_q}$) among curves of genera 3 and 4, respectively, that possess the maximal automorphism groups for these genera. More exactly, the groups $\mathrm{Aut}(K) \simeq \mathrm{PSL}_2(7)$ and $\mathrm{Aut}(B) \simeq \mathrm{S}_5$ are of orders $168 = 2^3 3 \cdot 7$ and $120 = 2^3 3 \cdot 5$, respectively (see, e.g., [13, Section 5.3]). Recall that elliptic curves of j-invariant 0 have over $\overline{\mathbb{F}_q}$ the largest automorphism group ($\simeq \mathbb{Z}/6$) in genus 1. These curves play a paramount role for ECC. Thus, the higher-genus analogues turn out to be also useful for this type of cryptography[1]. Unfortunately, the derived hash functions $\mathcal{H}_2$ are valid only for sporadic elliptic curves because of exceptional properties of the curves K, B. It seems very difficult (or even impossible) to obtain families of *hashing-friendly curves*, i.e., those having the parameter $k \geqslant 3$.

In [29, Section 2.2], the scenario $\overline{d} = (2, 7)$, i.e., $d = 14$ is dealt with for some elliptic $\mathbb{F}_q$-curve[2] of CM discriminant $D = -7$, i.e., of j-invariant $-3375 = -3^3 5^3$. It is about a curve E_7 covered over $\mathbb{F}_q$ by one C_7 of the superelliptic models of Klein's quartic K. One requires $q \equiv 2, 4 \pmod 7$ or, equivalently, $7 \nmid q - 1$ and $\sqrt{-7} \in \mathbb{F}_q$. The last condition is necessary for the ordinariness of E_7, otherwise the classical *MOV (Menezes–Okamoto–Vanstone) attack* [38] is applicable. Thus, the aforementioned instance $d = 42$ just consists in including Icart's hash function into consideration. For clarity, notable curves of $j = 0$, 1728 are exactly those of $D = -3$, -4, respectively, and -7 is the next CM discriminant. By analogy, the setting $d = 30$ is realized (in the main part of this paper) for an elliptic curve E_5 admitting an $\mathbb{F}_q$-cover by some superelliptic curve C_5 birationally isomorphic to Bring's one B. The curve E_5 is *CM-secure*, that is, its CM discriminant D is huge: $\log_2(-D) \approx \log_2(q)$. We thereby have two cardinally different hashing-friendly curves: E_7 is faster due to the *GLV (Gallant–Lambert–Vanstone) technique* [19], while E_5 is more reliable in the eyes of conservative stakeholders (such as the authors of [8]).

2 Main

Cornerstone formulas and claims of the given section can be verified with the help of the Magma code [30].

[1] In genus 2, the most symmetric curve is Bolza's one $y^2 = x^5 - x$ whose automorphism group is isomorphic to $\mathrm{GL}_2(3)$. Since its order $48 = 2^4 3$ is free of prime divisors > 3, this curve is beyond our interest.

[2] Recently, a pairing-friendly family with $D = -7$ and embedding degree 22 has been proposed in [20, Example 6] (under the name GG22D7) and sped up in [34] for the sake of the 192-bit security level. Nonetheless, it was analyzed in [2, Appendix A] that this family is not competitive with those of j-invariants 0, 1728 because of the absence of higher-degree twists. To the author's knowledge, this is the only situation before the current work when curves of $D = -7$ were tried to (justifiably) employ in ECC.

2.1 Batch hashing to elliptic curves

We start with a simple lemma, which should be valuable in its own right.

Lemma 1. *Let $d = \prod_{i=0}^{k-1} d_i$ be a factorization into pairwise relatively primes $d_i \geqslant 2$. Then, the roots $r_i := \sqrt[d_i]{p_i}$ of any field elements p_i are rationally expressed through p_j themselves and $r := \sqrt[d]{p}$, where $p := \prod_{i=0}^{k-1} p_i^{d'_i}$ and $d'_i := \prod_{j \neq i} d_j$.*

Proof. First, note that $r = \prod_{i=0}^{k-1} r_i$. As a consequence,

$$r^{d'_i} = r_i^{d'_i} \prod_{j \neq i} p_j^{\delta_{i,j}}, \qquad \text{that is,} \qquad p'_i := r_i^{d'_i} = \frac{r^{d'_i}}{\prod_{j \neq i} p_j^{\delta_{i,j}}},$$

where $\delta_{i,j} := \prod_{m \neq i,j} d_m$. Clearly, $\gcd(d_i, d'_i) = 1$ or, equivalently, there are c_i, $c'_i \in \mathbb{Z}$ such that $c_i d_i + c'_i d'_i = 1$ (schoolbook Bézout's identity). This implies that

$$r_i = r_i^{c_i d_i} r_i^{c'_i d'_i} = p_i^{c_i} (p'_i)^{c'_i}.$$

The lemma is proved. □

Bézout's lemma additionally says that the coefficients c_i, c'_i from the proof of the previous lemma can be picked, satisfying the inequalities $|c_i| < d'_i$ and $|c'_i| < d_i$. As a result, the exponents with respect to p_j, r for computing r_i are sufficiently modest once the initial values d_i, k are. Let's look at concrete examples important for us.

The case $\overline{d} = (2, 3)$. For

$$p = p_0^3 p_1^2, \qquad r = \sqrt[6]{p}, \qquad r_0 = \sqrt{p_0}, \qquad r_1 = \sqrt[3]{p_1},$$

we get:

$$r_0 = \frac{r^3}{p_0 p_1}, \qquad r_1 = \frac{r}{r_0};$$

The case $\overline{d} = (2, 3, 7)$. For

$$p = p_0^{21} p_1^{14} p_2^6, \qquad r = \sqrt[42]{p}, \qquad r_0 = \sqrt{p_0}, \qquad r_1 = \sqrt[3]{p_1}, \qquad r_2 = \sqrt[7]{p_2},$$

we get:

$$r_0 = \frac{r^{21}}{p_0^{10} p_1^7 p_2^3}, \qquad r_1 = \frac{p_0^7 p_1^5 p_2^2}{r^{14}}, \qquad r_2 = \frac{r}{r_0 r_1};$$

The case $\overline{d} = (2, 3, 5)$. For

$$p = p_0^{15} p_1^{10} p_2^6, \qquad r = \sqrt[30]{p}, \qquad r_0 = \sqrt{p_0}, \qquad r_1 = \sqrt[3]{p_1}, \qquad r_2 = \sqrt[5]{p_2},$$

we get:

$$r_0 = \frac{r^{15}}{p_0^7 p_1^5 p_2^3}, \qquad r_1 = \frac{r^{10}}{p_0^5 p_1^3 p_2^2}, \qquad r_2 = \frac{r}{r_0 r_1}.$$

Sticking to the notation from the introduction, the new approach of batch hashing to $E(\mathbb{F}_q)$ is formalized in Algorithm 1. It is prone to further acceleration (especially for considerable d_i, k) by means of *Pippenger's multi-exponentiation* celebrated, e.g., in [6].

Algorithm 1: Batch hashing to elliptic curves

Data: strings $s_0, \cdots, s_{k-1} \in \{0,1\}^*$ such that $\mu(s_i) \equiv i \bmod k$;
Result: the hash values $\mathcal{H}(s_i) = \mathcal{H}_i(s_i) \in E(\mathbb{F}_q)$;
begin
  $p := 1$;
  for $i := 0$ **to** $k-1$ **do**
    $x_i := \eta(s_i)$;
    $p_i := p_i(x_i)$;
    $p := p \cdot p_i^{d_i'}$;
  end
  $r := \sqrt[d]{p}$;
  for $i := 0$ **to** $k-1$ **do**
    $r_i := r_i(p_0, \cdots, p_{k-1}, r)$;
    $P_i := \phi_i(x_i, r_i)$;
  end
  return $P_0, \cdots, P_{k-1}$.
end

2.2 Algebraic geometry formulas

First, Klein's quartic is defined by the equation

$$K\colon v_0^3 v_1 + v_1^3 v_2 + v_2^3 v_0 = 0 \quad \subset \quad \mathbb{P}^2_{(v_0:v_1:v_2)}.$$

However, we are primarily interested in its affine model

$$C_7\colon y^7 = x^2(x+1)$$

covering over $\mathbb{F}_q$ the elliptic curve

$$E_7\colon y^2 + xy = x^3 - x^2 - 2x - 1$$

from the web page [35]. The corresponding cover has the form

$$\phi\colon C_7 \to E_7 \qquad (x,y) \mapsto \Big(\frac{\mathrm{num}_x}{\mathrm{den}}, \frac{\mathrm{num}_y}{\mathrm{den}}\Big),$$

where

$$\begin{aligned}
\mathrm{num}_x &:= \big(x^2(-y^7 - y^5 + y^2 + y + 1) + x(y^2 + y + 1)y^7 + (y^5 - y^2 - y - 1)y^7\big)y,\\
\mathrm{num}_y &:= x^2(2y^8 + y^7 + 2y^6 + y^5 + y^4 - y^3 - 2y^2 - 2y - 1) +\\
&\qquad x(y^4 - y^3 - 2y^2 - 2y - 1)y^7 + (-2y^6 - y^5 - y^4 + y^3 + 2y^2 + 2y + 1)y^7,\\
\mathrm{den} &:= (x^2 + xy^7 - y^7)y^3.
\end{aligned}$$

These formulas are almost as in [29, Section 2.2]. Once $\sqrt{7} \in \mathbb{F}_q$, the curve E_7 also possesses the Montgomery form

$$M_7\colon \sqrt{7}\cdot y^2 = x^3 + \frac{3\sqrt{7}}{4}x^2 + x$$

up to the isomorphism

$$\tau\colon E_7 \to M_7 \qquad (x,y) \mapsto \Big(\frac{x-2}{\sqrt{7}},\ \frac{x+2y}{14}\Big),$$
$$\tau^{-1}\colon M_7 \to E_7 \qquad (x,y) \mapsto \Big(\sqrt{7}\cdot x + 2,\ -\frac{\sqrt{7}}{2}x + 7y - 1\Big).$$

Meanwhile, Bring's curve is by definition the complete intersection

$$B := \begin{cases} v_0 + v_1 + v_2 + v_3 + v_4 = 0, \\ v_0^2 + v_1^2 + v_2^2 + v_3^2 + v_4^2 = 0, \\ v_0^3 + v_1^3 + v_2^3 + v_3^3 + v_4^3 = 0 \end{cases} \subset \mathbb{P}^4_{(v_0:\cdots:v_4)}.$$

As well as K, the curve B itself is not required to us in contrast to its plane affine model

$$C_5\colon (x-1)y^5 = x^2(x+1).$$

The following formulas are a slight modification of those from [45, Section 6]:

$$\phi\colon C_5 \to E_5 \qquad (x,y) \mapsto \Big(\frac{\mathrm{num}_x}{\mathrm{den}}, \frac{\mathrm{num}_y}{\mathrm{den}}\Big),$$

where

$$\begin{aligned} \mathrm{num}_x &:= \big(-x^2 + x(y^2-1)y + y^4\big)y, \\ \mathrm{num}_y &:= x^2(-y^2 - y + 1) + x(y^2 - 2y - 1)y^2 + (y^2 + y - 1)y^4, \\ \mathrm{den} &:= xy^3 \end{aligned}$$

and

$$E_5\colon y^2 + xy + y = x^3 - x - 2.$$

More information about this elliptic curve is contained on the web page [36]. Among other things, its j-invariant is equal to $-25/2$. Thanks to [42, Theorem II.6.1], its non-integrality implies that $E_5/\overline{\mathbb{Q}}$ is without complex multiplication, that is, the endomorphism ring $\mathrm{End}(E_5) \simeq \mathbb{Z}$ in characteristic 0. This means that the good reduction $E_5/\mathbb{F}_q$ at a prime characteristic has huge CM discriminant with overwhelming probability.

2.3 Cryptographic instances of the curves E_7, E_5

It is readily checked that $E_7(\mathbb{Q})_{\mathrm{tors}} \simeq \mathbb{Z}/2$ and $E_5(\mathbb{Q})_{\mathrm{tors}} \simeq \mathbb{Z}/3$. On the other hand, we want ordinary reductions $E_7/\mathbb{F}_q$ and $E_5/\mathbb{F}_q$ representable in the fast *Montgomery/twisted Edwards forms* [18, Section 9.12]. Up to a quadratic $\mathbb{F}_q$-isogeny, this happens if and only if 4 divides the curve $\mathbb{F}_q$-order N. Moreover, a (birational) $\mathbb{F}_q$-isomorphism is sufficient if there is an $\mathbb{F}_q$-point of order 4. For E_7 (ordinary only if $\sqrt{-7} \in \mathbb{F}_q$), it is enough to impose one more restriction $4 \mid q-1$, i.e., $\sqrt{-1} \in \mathbb{F}_q$ yielding the root $\sqrt{7} \in \mathbb{F}_q$ and hence the Montgomery curve M_7.

In fact, elliptic curves satisfying $3 \mid N$ enjoy the so-called *twisted Hessian form* [7], but it is not popular among implementers (like *double-odd curves* [39] when $2 \mid\mid N$). So, it was decided to increase the cofactor of E_5 to 12, obtaining an isomorphic Montgomery $\mathbb{F}_q$-curve M_5 at the price of loosing just two security bits. Unfortunately, some coefficients of M_7, M_5 (and of the elliptic curve from [29, Section 2.2]) are general field elements unlike those of E_7, E_5. Hence, in our context the advantage of the Montgomery arithmetic (as well as that of Edwards) is questionable without benchmarks of a low-level implementation. The point is that $E_7/\mathbb{Q}$ and $E_5/\mathbb{Q}$ are the (unique) *reduced minimal models* [5] for their $\mathbb{Q}$-isomorphism classes.

Both curves have been instantiated with ≈ 128-bit security level by changing only the base prime field $\mathbb{F}_q$ and keeping the curve coefficients in line with "the global method" of [25, Section 5.2]. Thus, the new curves are promptly encountered if one iterates q among primes, starting with $2^{255}+1$ as long as the following conditions are not met:

The case of E_7. The number $N/4 \in \mathbb{N}$ is prime, $8,\ 3 \nmid q-1$, but $4 \mid q-1$ and $q \equiv 2, 4 \pmod 7$;
The case of E_5. The number $N/12 \in \mathbb{N}$ is prime and $8,\ 3,\ 5 \nmid q-1$.

The generated $\mathbb{F}_q$-curves E_7, E_5, namely their subgroups of prime order $r > 12$ are actually safe, that is, they respect the requirements of [25, Section 5.2]. Inter alia, the curves are not anomalous (i.e., their Frobenius trace $\neq 1$), because they have non-trivial cofactors. Besides, it is easily checked in Magma that the embedding degree (i.e., the order of q modulo r) is of magnitude close to r for both curves as expected. Thereby, (the additive version [18, Section 26.4.1] of) the MOV attack is impossible in the case of E_7, E_5.

Finally, the new curves are appropriate for state-of-the-art *Koshelev's subgroup membership testing* [27,28] against the *small-subgroup attack* [33]. This countermeasure has just quadratic bit complexity in $\log_2(q) \approx 256$. In comparison, the naive multiplication by r (or its GLV-type modification [10,40] for E_7) has the cubic one. Curiously, Koshelev's test is grounded on the Tate pairing, despite the non-pairing-friendliness of E_7, E_5. In our situation, the given technique principally consists in computing two Legendre symbols $\left(\frac{\cdot}{q}\right)$ for E_7 as well as two *power residue symbols* $\left(\frac{\cdot}{q^2}\right)_4$ for E_5 and two ones $\left(\frac{\cdot}{q^2}\right)_3$ for E_5. This follows from the successfully formed group structures (unchangeable when transitioning to M_7, M_5). Indeed, $E_7(\mathbb{F}_q)[2] \simeq (\mathbb{Z}/2)^2$ and

$$E_5(\mathbb{F}_{q^2})[12] = E_5(\mathbb{F}_{q^2})[6 \cdot 12] \simeq (\mathbb{Z}/12)^2, \qquad 12 \mid q^2-1,$$

respectively. Note that the test does not work over $\mathbb{F}_q$ for the curve E_5, since

$$E_5(\mathbb{F}_q)[12] \simeq \mathbb{Z}/12, \qquad 4, 3 \nmid q-1.$$

It may be feasible to find another field $\mathbb{F}_q$ for which either $E_5(\mathbb{F}_q)[2] \simeq (\mathbb{Z}/2)^2$ or $4 \mid q-1$ similarly to the curve E_7. However, it makes little sense, as the second restriction $3 \nmid q-1$ cannot be eliminated in principle.

3 Conclusion

To sum up, Table 1 exhibits basically all known tuples $\overline{d}$ from the literature, suitable for batch hashing to elliptic curves (including the conventional case $k = 1$). A more subtle up-to-date classification of non-batch hash-to-curve functions is represented in [29, Section 3]. It is classical Shallue–van de Woestijne–Skałba's fact that the parameter k is positive for every elliptic $\mathbb{F}_q$-curve. Again, there are variations for square-root hash function $\mathcal{H}_0$, targeting Legendre-symbol elimination (primarily due to [12, Section 7], [44, Section 4]) or indifferentiability property (due to [15,31]), but the mentioned references are omitted in the table for conciseness.

k	$\overline{d}$	d	D	Conditions	Reference
1	2		any		[41,43]
2	$(2,3)$	6	any	$3 \nmid q-1$	[29, Section 2.1]
			-3	$27 \nmid q-1$ and $3 \mid q-1$, N	
3	$(2,3,7)$	42	-7	$3 \nmid q-1$ and $q \equiv 2,4 \pmod 7$ and $2 \mid N$	this work
	$(2,3,5)$	30	huge	$3, 5 \nmid q-1$ and $3 \mid N$	

Table 1. Existing instances of batch hashing to elliptic curves (for all of them, $8 \nmid q-1$)

At the end, it is worth stating explicitly a few challenging problems.

Problem 1. Is the case $k \geqslant 2$ realizable for any pair $E/\mathbb{F}_q$?

If so, it will follow that any elliptic $\mathbb{F}_q$-curve (at least of j-invariant $\neq 0$) admits an indifferentiable hash function with the cost of only one $\mathbb{F}_q$-root (cf. [29, Conjecture 1]) and without Legendre symbols as opposed to [31, Corollary 3]. Of course, this by-product result would be of theoretical significance rather than of practical one.

Problem 2. Are there elliptic curves of j-invariants 0, 1728 that are simultaneously hashing-friendly (i.e., $k \geqslant 3$) and pairing-friendly?

Problem 3. Are there elliptic curves with $k \geqslant 4$?

Finally, even though in this article we have not concerned a lot the behavior of hash functions as random oracles, it is also worthwhile to wonder about the following.

Problem 4. Are there elliptic curves that are *indifferentiably hashing-friendly*, i.e., those having at least 3 indifferentiable hash functions $\mathcal{H}_i$ with pairwise coprime degrees d_i?

Counting starts from 3 in the last problem, because the curves E_7, E_5 clearly admit the random oracle $\mathcal{H}_1$ with $d_1 = 21$ and 15, respectively, in addition to the SwiftEC(-like) construction $\mathcal{H}_0$ with $d_0 = 2$ as before. Moreover, [26] allows the value $d_1 = 3$ for numerous (pairing-friendly) curves of j-invariant 0.

Acknowledgements. The author expresses his gratitude to Emanuele Cesena for attracting attention to the problem of batch hashing to elliptic curves. This research is a result of the strategic project "Advances in post-quantum cryptography applied to the development of a coupon system" (C039/24), resulting from an agreement between the Spanish National Cybersecurity Institute (INCIBE) and University of Lleida. This initiative is carried out in the scope of the funds from the Recovery, Transformation and Resilience Plan, funded by the European Union (Next Generation). The paper is also part of the R&D+i project PID2021-124613OB-I00 funded by MICIU/AEI/10.13039/501100011033 and FEDER, EU. Besides, the author was supported by Ethereum Foundation through the grant FY25-1896.

References

1. Aranha, D.F., El Housni, Y., Guillevic, A.: A survey of elliptic curves for proof systems. Designs, Codes and Cryptography **91**(11), 3333–3378 (2023), https://doi.org/10.1007/s10623-022-01135-y
2. Aranha, D.F., Fotiadis, G., Guillevic, A.: A short-list of pairing-friendly curves resistant to the special TNFS algorithm at the 192-bit security level. IACR Communications in Cryptology **1**(3), article 3 (2024), https://doi.org/10.62056/angyl86bm
3. Aranha, D.F., Salling Hvass, B., Spitters, B., Tibouchi, M.: Faster constant-time evaluation of the Kronecker symbol with application to elliptic curve hashing. In: CCS 2023: ACM SIGSAC Conference on Computer and Communications Security. pp. 3228–3238. Association for Computing Machinery, New York (2023), https://doi.org/10.1145/3576915.3616597
4. Barreto, P.S.L.M., Lynn, B., Scott, M.: Constructing elliptic curves with prescribed embedding degrees. In: Cimato, S., Persiano, G., Galdi, C. (eds.) Security in Communication Networks. SCN 2002. Lecture Notes in Computer Science, vol. 2576, pp. 257–267. Springer, Berlin, Heidelberg (2003), https://doi.org/10.1007/3-540-36413-7_19
5. Barrios, A.J.: Reduced minimal models and torsion (2023), https://arxiv.org/abs/2301.09488
6. Bernstein, D.J.: Pippenger's exponentiation algorithm (2002), https://cr.yp.to/papers/pippenger-20020118-retypeset20220327.pdf
7. Bernstein, D.J., Chuengsatiansup, C., Kohel, D., Lange, T.: Twisted Hessian curves. In: Lauter, K., Rodríguez-Henríquez, F. (eds.) Progress in Cryptology – LATINCRYPT 2015. Lecture Notes in Computer Science, vol. 9230, pp. 269–294. Springer, Cham (2015), https://doi.org/10.1007/978-3-319-22174-8_15
8. Bernstein, D.J., Lange, T.: CM field discriminants (2013), https://safecurves.cr.yp.to/disc.html

9. Boneh, D., Gentry, C., Lynn, B., Shacham, H.: Aggregate and verifiably encrypted signatures from bilinear maps. In: Biham, E. (ed.) Advances in Cryptology – EUROCRYPT 2003. Lecture Notes in Computer Science, vol. 2656, pp. 416–432. Springer, Berlin, Heidelberg (2003), https://doi.org/10.1007/3-540-39200-9_26
10. Bowe, S.: Faster subgroup checks for BLS12-381 (2019), https://eprint.iacr.org/2019/814
11. Braden, H.W., Disney-Hogg, L.: Bring's curve: old and new. European Journal of Mathematics **10**(1), article 3 (2024), https://doi.org/10.1007/s40879-023-00706-0
12. Brier, E., Coron, J.S., Icart, T., Madore, D., Randriam, H., Tibouchi, M.: Efficient indifferentiable hashing into ordinary elliptic curves. In: Rabin, T. (ed.) Advances in Cryptology – CRYPTO 2010. Lecture Notes in Computer Science, vol. 6223, pp. 237–254. Springer, Berlin, Heidelberg (2010), https://doi.org/10.1007/978-3-642-14623-7_13
13. Broughton, S.A., Jones, G.A., Singerman, D.: The engaging symmetry of Riemann surfaces: a historical perspective. In: Wootton, A., Broughton, S.A., Paulhus, J. (eds.) Automorphisms of Riemann surfaces, subgroups of mapping class groups and related topics. Contemporary Mathematics, vol. 776, pp. 1–35. American Mathematical Society, Providence (2022), https://doi.org/10.1090/conm/776
14. Chevignard, C., Tibouchi, M.: New Icart-type encodings from elliptic curve automorphisms (2022), https://www.youtube.com/watch?v=krvPUUFQ79E
15. Chávez-Saab, J., Rodríguez-Henríquez, F., Tibouchi, M.: SwiftEC: Shallue-van de Woestijne indifferentiable function to elliptic curves. In: Agrawal, S., Lin, D. (eds.) Advances in Cryptology – ASIACRYPT 2022. Lecture Notes in Computer Science, vol. 13791, pp. 63–92. Springer, Cham (2022), https://doi.org/10.1007/978-3-031-22963-3_3
16. Faz-Hernandez, A., Scott, S., Sullivan, N., Wahby, R.S., Wood, C.A.: Hashing to elliptic curves (RFC 9380) (2023), https://datatracker.ietf.org/doc/draft-irtf-cfrg-hash-to-curve
17. Fiat, A.: Batch RSA. In: Brassard, G. (ed.) Advances in Cryptology – CRYPTO 1989. Lecture Notes in Computer Science, vol. 435, pp. 175–185. Springer, New York (1990), https://doi.org/10.1007/0-387-34805-0_17
18. Galbraith, S.D.: Mathematics of public key cryptography. Cambridge University Press, New York (2012), https://doi.org/10.1017/CBO9781139012843
19. Gallant, R.P., Lambert, R.J., Vanstone, S.A.: Faster point multiplication on elliptic curves with efficient endomorphisms. In: Kilian, J. (ed.) Advances in Cryptology – CRYPTO 2001. Lecture Notes in Computer Science, vol. 2139, pp. 190–200. Springer, Berlin, Heidelberg (2001), https://doi.org/10.1007/3-540-44647-8_11
20. Gasnier, J., Guillevic, A.: An algebraic point of view on the generation of pairing-friendly curves. SIAM Journal on Applied Algebra and Geometry **9**(2), 456–480 (2025), https://doi.org/10.1137/23M1601961
21. Gidney, C.: Why haven't quantum computers factored 21 yet? (2025), https://algassert.com/post/2500
22. Groth, J., Malvai, H., Miller, A., Zhang, Y.N.: Constraint-friendly map-to-elliptic-curve-group relations and their applications. In: Hanaoka, G., Yang, B.Y. (eds.) Advances in Cryptology – ASIACRYPT 2025. Lecture Notes in Computer Science, vol. 16246, pp. 511–543. Springer, Singapore (2026), https://doi.org/10.1007/978-981-95-5096-8_16

23. Gutmann, P., Neuhaus, S.: Replication of quantum factorisation records with an 8-bit home computer, an abacus, and a dog (2025), https://eprint.iacr.org/2025/1237
24. Icart, T.: How to hash into elliptic curves. In: Halevi, S. (ed.) Advances in Cryptology – CRYPTO 2009. Lecture Notes in Computer Science, vol. 5677, pp. 303–316. Springer, Berlin, Heidelberg (2009), https://doi.org/10.1007/978-3-642-03356-8_18
25. Koblitz, N.I., Menezes, A.J., Vanstone, S.A.: The state of elliptic curve cryptography. Designs, Codes and Cryptography **19**(2-3), 173–193 (2000), https://doi.org/10.1023/A:1008354106356
26. Koshelev, D.: Indifferentiable hashing to ordinary elliptic $\mathbb{F}_q$-curves of $j = 0$ with the cost of one exponentiation in $\mathbb{F}_q$. Designs, Codes and Cryptography **90**(3), 801–812 (2022), https://doi.org/10.1007/s10623-022-01012-8
27. Koshelev, D.: Subgroup membership testing on elliptic curves via the Tate pairing. Journal of Cryptographic Engineering **13**(1), 125–128 (2023), https://doi.org/10.1007/s13389-022-00296-9
28. Koshelev, D.: Correction to: Subgroup membership testing on elliptic curves via the Tate pairing. Journal of Cryptographic Engineering **14**(1), 127–128 (2024), https://doi.org/10.1007/s13389-023-00331-3
29. Koshelev, D.: Some remarks on how to hash faster onto elliptic curves. Journal of Computer Virology and Hacking Techniques **20**(4), 593–605 (2024), https://doi.org/10.1007/s11416-024-00514-4
30. Koshelev, D.: Magma code (2025), https://github.com/Dimitri-Koshelev/Hashing-friendly-elliptic-curves
31. Koshelev, D.: Simultaneously simple universal and indifferentiable hashing to elliptic curves. In: Nitaj, A., Petkova-Nikova, S., Rijmen, V. (eds.) Progress in Cryptology – AFRICACRYPT 2025. Lecture Notes in Computer Science, vol. 15651, pp. 395–413. Springer, Cham (2026), https://doi.org/10.1007/978-3-031-97260-7_18
32. Levy, S. (ed.): The eightfold way: the beauty of Klein's quartic curve, Mathematical Sciences Research Institute Publications, vol. 35. Cambridge University Press, Cambridge (1999), https://doi.org/10.1017/9781009701730
33. Lim, C.H., Lee, P.J.: A key recovery attack on discrete log-based schemes using a prime order subgroup. In: Kaliski, B.S. (ed.) Advances in Cryptology – CRYPTO 1997. Lecture Notes in Computer Science, vol. 1294, pp. 249–263. Springer, Berlin, Heidelberg (1997), https://doi.org/10.1007/BFb0052240
34. Lin, J., Zhao, C.A., Zheng, Y.: Constructing more super-optimal pairings via small degree endomorphisms (2024), https://eprint.iacr.org/2024/1195
35. LMFDB collaboration: Elliptic curve with LMFDB label 49.a4 (Cremona label 49a1), https://www.lmfdb.org/EllipticCurve/Q/49/a/4
36. LMFDB collaboration: Elliptic curve with LMFDB label 50.a3 (Cremona label 50a1), https://www.lmfdb.org/EllipticCurve/Q/50/a/3
37. Maitin-Shepard, J., Tibouchi, M., Aranha, D.F.: Elliptic curve multiset hash. The Computer Journal. Section D: Security in Computer Systems and Networks **60**(4), 476–490 (2017), https://doi.org/10.1093/comjnl/bxw053
38. Menezes, A.J., Okamoto, T., Vanstone, S.A.: Reducing elliptic curve logarithms to logarithms in a finite field. IEEE Transactions on Information Theory **39**(5), 1639–1646 (1993), https://doi.org/10.1109/18.259647
39. Pornin, T.: A prime-order group with complete formulas from even-order elliptic curves. IACR Communications in Cryptology **1**(1), article 10 (2024), https://doi.org/10.62056/akmp-4c2h

40. Scott, M.: A note on group membership tests for $\mathbb{G}_1$, $\mathbb{G}_2$ and $\mathbb{G}_T$ on BLS pairing-friendly curves (2021), https://eprint.iacr.org/2021/1130
41. 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.) Algorithmic Number Theory Symposium. ANTS 2006. Lecture Notes in Computer Science, vol. 4076, pp. 510–524. Springer, Berlin, Heidelberg (2006), https://doi.org/10.1007/11792086_36
42. Silverman, J.H.: Advanced topics in the arithmetic of elliptic curves, Graduate Texts in Mathematics, vol. 151. Springer, New York (1994), https://doi.org/10.1007/978-1-4612-0851-8
43. Skałba, M.: Points on elliptic curves over finite fields. Acta Arithmetica **117**(3), 293–301 (2005), https://doi.org/10.4064/aa117-3-7
44. Wahby, R.S., Boneh, D.: Fast and simple constant-time hashing to the BLS12-381 elliptic curve. IACR Transactions on Cryptographic Hardware and Embedded Systems (CHES) **2019**(4), 154–179 (2019), https://doi.org/10.13154/tches.v2019.i4.154-179
45. Weber, M.: Kepler's small stellated dodecahedron as a Riemann surface. Pacific Journal of Mathematics **220**(1), 167–182 (2005), http://doi.org/10.2140/pjm.2005.220.167