Fast Hashing to $\mathbb{G}_2$ in Direct Anonymous Attestation

Yu Dai , Fangguo Zhang and Chang-An Zhao*

Abstract

To reduce the workload of the Trusted Platform Module (TPM) without affecting the security in pairing-based direct anonymous attestation (DAA) schemes, it is feasible to select pairing-friendly curves that provide fast group operations in the first pairing subgroup. In this scenario, the *BW13-P310* and *BW19-P286* curves become competitive. In order to improve the efficiency of the DAA schemes based on these curves, it is also necessary to design an efficient algorithm for hashing to $\mathbb{G}_2$. In this paper, we first generalize the previous work to address the bottlenecks involved in hashing to $\mathbb{G}_2$ on the two curves. On this basis, we further optimize the hashing algorithm, which would be nearly twice as fast as the previous one in theory. These techniques actually can be applied to a large class of curves. We also implement the proposed algorithms over the *BW13-P310* curve on a 64-bit computing platform.

Index Terms

Direct anonymous attestation, Pairing-friendly curves, Hashing to $\mathbb{G}_2$

I. Introduction

Pairings are a powerful mathematical tool to construct various cryptographic protocols with novel properties, such as identity-based encryption [6], direct anonymous attestation (DAA) [8], and zero-knowledge succinct non-interactive argument of knowledge (zk-SNARK) [25]. In implementation of pairing-based protocols, the selection of pairing-friendly curves is closely related to particular requirements given by specifications of the considered application [2], [10], [11].

Yu Dai is with School of Mathematics, Sun Yat-sen University, Guangzhou 510275, P.R.China.

Chang-An Zhao is with School of Mathematics, Sun Yat-sen University, Guangzhou 510275, P.R.China and with Guangdong Key Laboratory of Information Security, Guangzhou 510006, P.R. China.

Fangguo Zhang is with School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou 510006, P.R. China.

$*$ Corresponding author (E-mail: zhaochan3@mail.sysu.edu.cn)

For example, Yang $et\ al.$ [36] pointed out that the Trusted Platform Module (TPM) is a small chip equipped with constrained resources in DAA schemes. It is interesting to design a DAA scheme such that the workload of the TPM is as small as possible. In this scenario, curves with small size of prime p are particularly helpful since the TPM requires performing a number of scalar multiplications in the first pairing subgroup. Based on the above facts, Clarisse $et\ al.$ [10] suggested two curves suitable for the DAA schemes: *BW13-P310* with embedding degree 13 over a 310-bit field, and *BW19-P286* with embedding degree 19 over a 286-bit field. Consequently, it is meaningful to present efficient algorithms for building blocks in the DAA schemes based on the two curves.

Cryptographic pairings are built on elliptic curves over finite fields up till now. The following standard notations and settings are used throughout the whole paper. Let p be a large prime and E an elliptic curve over $\mathbb{F}_p$ with equation $y^2 = x^3 + ax + b$. Assume that r is a large prime with $r \parallel \#E(\mathbb{F}_p)$ (the notation $a \parallel b$ means $a \mid b$ but $a^2 \nmid b$). Let k be the smallest positive integer such that r divides $p^k - 1$. If $k > 1$, then the subgroup $E[r]$ is contained in $E(\mathbb{F}_{p^k})$ [3]. Denote by $\pi : (x, y) \to (x^p, y^p)$ the p-th power Frobenius endomorphism on E. A pairing $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$ is a bilinear and non-degenerate function, where $\mathbb{G}_1 = E(\mathbb{F}_p)[r]$, $\mathbb{G}_2 = E[r] \cap \text{Ker}(\pi - [p])$ and $\mathbb{G}_T$ is a subgroup of order r of $\mathbb{F}_{p^k}^*$.

In the DAA scheme proposed in [36], it is necessary to hash binary strings to $\mathbb{G}_1$ and $\mathbb{G}_2$. The standard approach of hashing to $\mathbb{G}_1$ is to hash to a random point of $E(\mathbb{F}_p)$ [16], [32], [34], followed by a scalar multiplication by the cofactor $h_1 = \#E(\mathbb{F}_p)/r$. The computational cost of the multiplication is not expensive since the size of h_1 is typically small on most of pairing-friendly curves. Recently, a more efficient method for clearing the cofactor is proposed in [12], which may further reduce the computational cost of hashing to $\mathbb{G}_1$.

By contrast, hashing to $\mathbb{G}_2$ is relatively complicated. Denote by $\text{Aut}(E)$ the automorphism group of E and let $d = \gcd(k, \#\text{Aut}(E))$. If $d > 1$ and $r^2 \parallel E(\mathbb{F}_{p^k})$, there exists a unique d-twist E' of E such that $r \parallel \#E'(\mathbb{F}_{p^e})$ [26, Section 4], where $e = k/d$. In this situation, the group $\mathbb{G}_2$ can be represented as $E'(\mathbb{F}_{p^e})[r]$. As such, the procedure of hashing to $\mathbb{G}_2$ consists of two phases: first hashing an arbitrary string to $E'(\mathbb{F}_{p^e})$, followed by a scalar multiplication by $h_2 = \#E'(\mathbb{F}_{p^e})/r$. In the case $e \neq 1$, the size of h_2 is large and thus the cofactor multiplication is expensive. In 2009, Scott $et\ al.$ [31] proposed an efficient method to reduce the computational cost of the cofactor multiplication. This method was further optimized by Fuentes $et\ al.$ [19]. Detailed comparisons of the two methods on different curves were given in [9], [15].

Unfortunately, we find that $\gcd(k, \#\text{Aut}(E)) = 1$ for the *BW13-P310* and *BW19-P286* curves. It implies that $\mathbb{G}_2$ can be only represented as $E[r] \cap \text{Ker}(\pi - [p])$. Following Enge and Milan [14], we refer to this type of curves as **curves with the lack of twists**. In fact, if $p \geq 5$, then $\#\text{Aut}(E) \in \{2, 4, 6\}$ [33, Theorem III.10.1]. So for $p \geq 5$, we can see that E is a curve with the lack of twists if and only if $\gcd(k, 6) = 1$. It seems that the extra effort is required for adapting the above two approaches of hashing to $\mathbb{G}_2$ to pairing-friendly curves with the lack of twists.

Our contributions. The DAA scheme proposed in [36] consists of three algorithms: **Setup**, **Verify** and **Link**. Hashing to $\mathbb{G}_2$ is performed by the host in the **Verify** algorithm. In order to implement the DAA scheme based on the *BW13-P310* or *BW19-P286* curve efficiently, we investigate the problem of hashing to $\mathbb{G}_2$ on such type of curves in detail. To be precise, our contributions are mainly divided into the following two parts:

- We introduce the cyclotomic zero subgroup G_0 of elliptic curves. It is confirmed that $\mathbb{G}_2$ is the unique subgroup of order r contained in $\mathbb{G}_0$. Thus, given a random point of $E(\mathbb{F}_{p^k})$, we first map it to $\mathbb{G}_0$. Then, we generalize the Fuentes $et\ al.$ method [19] (denoted by Method I) to map a point of G_0 to $\mathbb{G}_2$ on the target curves. On the basis, a more efficient method is proposed and we denote it by Method II. It should be noted that Method II is only suitable for ordinary elliptic curves with j-invariant 0 or 1728.
- In order to explain the benefits resulting from Method II, we implement the two hashing algorithms over the *BW13-P310* curve on a 64-bit computing platform. Experimental results show that Method II leads to roughly 77.0% improvement for the whole procedure of hashing to $\mathbb{G}_2$. Hence, it is suitable for the DAA scheme given in [36].

Outline of the paper. The remainder of this paper is organized as follows. Section II introduces the standard approach of hashing to $\mathbb{G}_2$ on curves with the lack of twists, the endomorphism ring and the group structure of ordinary elliptic curves. In Section III, we define the cyclotomic zero subgroup of elliptic curves. The main results of hashing to $\mathbb{G}_2$ on curves with the lack of twists are presented in Section IV. The application of the proposed technique in DAA schemes is considered in Section V. Finally, we draw our conclusion in Section VI.

II. Background

In this section, we first recall the standard approach of hashing to $\mathbb{G}_2$ on pairing-friendly curves with the lack of twists. Then we introduce endomorphism ring and group structure of ordinary elliptic curves, which are exploited to speeding up the efficiency of hashing to $\mathbb{G}_2$ on the target curves.

A. The standard approach of hashing to $\mathbb{G}_2$

Let $E/\mathbb{F}_p$ be an ordinary elliptic curve with the lack of twists. Denote by $\mathcal{O}_E$ and $j(E)$ the identity element and j-invariant of E, respectively. For arbitrary $m \in \mathbb{Z}^+$, let t_m be the trace of the p^m-power Frobenius on E. Then the order of $E(\mathbb{F}_{p^m})$ is precisely $p^m + 1 - t_m$. Given $t_1 = t$, then the value t_m for any $m > 1$ can be obtained by performing the following recursion [35, Lemma 4.13]

$$t_0 \leftarrow 2, \quad t_1 \leftarrow t, \quad t_{i+1} \leftarrow t \cdot t_i - p \cdot t_{i-1}.$$

Given a random point $Q \in E(\mathbb{F}_{p^k})$, the standard approach of hashing to $\mathbb{G}_2$ on this type of curves is done as follows:

$$Q \rightarrow R = cQ \rightarrow \text{Tr}(R) - kR,$$

where the cofactor $c = \#E(\mathbb{F}_{p^k})/r^2$ and the mapping Tr acts as

$$\text{Tr}(S) = S + \pi(S) + \cdots + \pi^{k-1}(S)$$

for all $S \in E(\mathbb{F}_{p^k})$. The cofactor multiplication maps Q into $E[r]$, and the endomorphism $\text{Tr} - k$ forces R into $\mathbb{G}_2$ [5, Section IX]. It is obvious that the approach is extremely inefficient and shall not be admissible in practical applications.

B. Endomorphism ring of ordinary elliptic curves

Let $t^2 - 4p = -Df^2$ where D is square free and $f \in \mathbb{Z}$. The value D is referred to as the CM discriminant of E. Denote by K the imaginary quadratic field $Q(\sqrt{-D})$ and let O_K be the largest subring of K. An order in K is a subring $\mathscr{O}$ satisfying that $\mathbb{Z} \subsetneqq \mathscr{O} \subseteq O_K$. Any order $\mathscr{O}$ has the form $\mathbb{Z} \oplus \mathbb{Z}y\delta$, where $\delta = \sqrt{-D}$ or $(1 + \sqrt{-D})/2$ and $y \in \mathbb{Z}$. Let $D_{\mathscr{O}}$ denote the discriminant of order $\mathscr{O}$, which is defined as

$$D_{\mathscr{O}} = \begin{cases} -y^2 D, \text{if } D \equiv 3 \bmod 4, \\ -4y^2 D, \text{if } D \equiv 1, 2 \bmod 4. \end{cases}$$

Let q be a power of the prime p. Let $\text{End}_{\mathbb{F}_q}(E)$ and $\text{End}(E)$ be the ring of endomorphisms of E over $\mathbb{F}_q$ and $\overline{\mathbb{F}}_q$, respectively. Since E is ordinary, the ring $\text{End}(E)$ is isomorphic to an order in K [35, Theorem 10.6]. Thus, we conclude that $\text{End}_{\mathbb{F}_q}(E) \subseteq \text{End}(E) \subseteq O_K$. Generally speaking, it is not straightforward to determine $\text{End}_{\mathbb{F}_q}(E)$. However, for some special ordinary elliptic curves, the question becomes simple. To be precise,

- if $j(E) = 0$, then $p \equiv 1 \bmod 3$ [35, Proposition 4.33]. There exists an endomorphism $\phi \in \text{End}_{\mathbb{F}_q}(E)$ acting as $\phi : (x, y) \to (\omega \cdot x, y)$, where ω is a primitive cube root of unity in $\mathbb{F}_p^*$. Since ϕ satisfies $\phi^2 + \phi + 1 = 0$, we have $\mathbb{Z}[\phi] = \mathbb{Z}[(1 + \sqrt{-3})/2] = O_K$. It implies that $\text{End}_{\mathbb{F}_q}(E) = O_K$ as $\mathbb{Z}[\phi] \subseteq \text{End}_{\mathbb{F}_q}(E) \subseteq O_K$;
- if $j(E) = 1728$, then $p \equiv 1 \bmod 4$ [35, Theorem 4.23]. There exists an endomorphism $\phi \in \text{End}_{\mathbb{F}_q}(E)$ acting as $\phi : (x, y) \to (-x, i \cdot y)$, where i is a primitive fourth root of unity in $\mathbb{F}_p^*$. Since $\phi^2 + 1 = 0$, it holds that $\mathbb{Z}[\phi] = \mathbb{Z}[\sqrt{-1}] = O_K$ and thus we conclude $\text{End}_{\mathbb{F}_q}(E) = O_K$.

We can see that the discriminant of $\text{End}_{\mathbb{F}_q}(E)$ for the above two classes of curves are -3 and -4, respectively. The following subsection will illustrate the connection between the group structure of ordinary curves and the associated endomorphism ring.

C. Group structure of ordinary elliptic curves over finite fields

By the basic theory of elliptic curves over finite fields [35, Thoerem 4.1], we know that $E(\mathbb{F}_q) \cong \mathbb{Z}_{m_1} \oplus \mathbb{Z}_{m_2}$ with $m_1 \mid m_2$. In particular, if $m_1 = 1$ then $E(\mathbb{F}_q)$ is cyclic. However, one can not determine the values m_1 and m_2 directly, even if $\#E(\mathbb{F}_q)$ is known. Indeed, let ℓ be an integer such that $\ell^2 \parallel \#E(\mathbb{F}_q)$. There are the following two possibilities: (a) $E[\ell] \subseteq E(\mathbb{F}_q)$; (b) $E(\mathbb{F}_q)$ has a cyclic subgroup of order ℓ^2. A sufficient and necessary condition of the question is given by Schoof, which is shown as follows.

Theorem 1 [30, Propostition 3.7] *Let E be an ordinary elliptic curve over $\mathbb{F}_q$, where q is a power of the prime p. Let $\tilde{t}$ denote the trace of the q-power Frobenius endomorphism $\pi_q : (x, y) \to (x^q, y^q)$. Assume $\ell \in \mathbb{Z}^+$ with $p \nmid \ell$. Then $E[\ell] \subseteq E(\mathbb{F}_q)$ if and only if $\ell^2 \mid \#E(\mathbb{F}_q)$, $\ell \mid q - 1$ and $\mathscr{O}(\frac{\tilde{t}^2-4q}{\ell^2}) \subseteq \text{End}_{\mathbb{F}_q}(E)$, where the notation $\mathscr{O}(\frac{\tilde{t}^2-4q}{\ell^2})$ represents the order with discriminant $\frac{\tilde{t}^2-4q}{\ell^2}$.*

The condition of Theorem 1 can be further simplified for curves with j-invariant 0 or 1728, which is given in Proposition 1. It should be noted that the conclusion is actually well known to the experts but we are incapable of finding the proof in the literature.

Proposition 1 *Let notation be as above. Assume $j(E) = 0$ or 1728. Then $E[\ell] \subseteq E(\mathbb{F}_q)$ if and only if $\ell^2 \mid \#E(\mathbb{F}_q)$ and $\ell \mid q - 1$.*

Proof *Since $j(E) = 0$ or 1728, it can be seen from Subsection II.B that $\mathrm{End}_{\mathbb{F}_q}(E) = O_K$ and thus the condition $\mathscr{O}(\frac{\tilde{t}^2-4q}{\ell^2}) \subseteq \mathrm{End}_{\mathbb{F}_q}(E)$ holds. By Theorem 1, the rest of the proof is immediate.* ■

Given the factorization of $\gcd(\#E(\mathbb{F}_q), q - 1)$, Proposition 1 induces a simple method to determine the group structure of $E(\mathbb{F}_q)$ if $j(E) = 0$ or 1728. For general cases, an alternative approach is given in [29, Algorithm 3], which is slightly complicated. In the following, we show how to use the conclusion to improve the efficiency of hashing to $\mathbb{G}_2$ on pairing-friendly curves with the lack of twists.

III. Cyclotomic zero subgroup of elliptic curves

We denote by G_0 the set

$$\{Q \in E(\mathbb{F}_{p^k}) \mid \Phi_k(\pi)(Q) = \mathcal{O}_E\},$$

where $\Phi_k(\pi)$ is the k-th cyclotomic polynomial with respect to π. It is obvious that G_0 forms a group. Recall that the trace zero subgroup T_k [18] of $E(\mathbb{F}_{p^k})$ is exactly the kernel of the mapping Tr. The subgroup T_k is particularly interesting in elliptic-curve cryptography [23], [24]. By the definition of G_0, we clearly have $G_0 \subseteq T_k$. In particular, the two subgroups are identical if and only if the embedding degree k is prime. In the following, we call G_0 the cyclotomic zero subgroup of $E(\mathbb{F}_{p^k})$. Several properties of G_0 are summarized in the following two propositions.

Proposition 2 *Let notation be as above. Then the order of the group G_0 is precisely equal to $\prod\limits_{d|k} \#E(\mathbb{F}_{p^d})^{\mu(k/d)}$, where $\mu(.)$ is the Moebius function. In addition, if $r \nmid \Phi_k(1)$, then we have the following equality $E[r] \cap G_0 = \mathbb{G}_2$.*

Proof *By* [28, Theorem 3.27]*, we have*

$$\Phi_k(\pi) = \prod_{d|k} (\pi^d - 1)^{\mu(k/d)}. \tag{1}$$

Since $\Phi_k(\pi)$ and $\pi^d - 1$ are separable, taking degrees of both sides of Equation (1), it yields that

$$\#G_0 = \#\mathrm{Ker}(\Phi_k(\pi)) = \prod_{d|k} \#\mathrm{Ker}(\pi^d - 1)^{\mu(k/d)}. \tag{2}$$

Furthermore, since $E(\mathbb{F}_{p^d}) = \mathrm{Ker}(\pi^d - 1)$ for any $d \in \mathbb{Z}^+$, Equation (2) implies that $\#G_0 = \prod_{d|k} \#E(\mathbb{F}_{p^d})^{\mu(k/d)}$.

Let R_1 and R_2 be generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. Then we get $\pi(R_1) = R_1$ and $\pi(R_2) = pR_2$. Since $r \nmid \Phi_k(1)$ and $r \mid \Phi_k(p)$, it indicates that

$$\Phi_k(\pi)(R_1) = \Phi_k(1)R_1 \neq \mathcal{O}_E, \tag{3}$$

$$\Phi_k(\pi)(R_2) = \Phi_k(p)R_2 = \mathcal{O}_E. \tag{4}$$

For any $R \in E[r]$, there exist $m_1, m_2 \in \mathbb{Z}/r\mathbb{Z}$ such that

$$R = m_1R_1 + m_2R_2.$$

By the relations (3) and (4) we can deduce that $R \in G_0$ if and only if $m_1 = 0$, which implies that $R \in \mathbb{G}_2$ and thus completes the proof. ■

It can be seen from Proposition 2 that given the order of $E(\mathbb{F}_{p^d})$ for each $d \mid k$, it is simple to calculate that of G_0. In addition, since r is a large prime in pairing-based cryptographic schemes, the condition $r \nmid \Phi_k(1)$ clearly holds. Thus, we confirm that $\mathbb{G}_2$ is the unique subgroup of order r contained in G_0. We now consider how to determine the group structure of G_0 for curves with j-invariant 0 or 1728.

Proposition 3 *Let notation be as above. Let $E/\mathbb{F}_p$ be an ordinary elliptic curve with $j(E) = 0$ or 1728. Assume $\ell \in \mathbb{Z}^+$ with $p \nmid \ell$ and $\ell^3 \nmid \#E(\mathbb{F}_{p^k})$. Then $E[\ell] \subseteq G_0$ if and only if $\ell^2 \mid \#G_0$ and $\ell \mid p^k - 1$.*

Proof *By Proposition 1, the necessity is obvious, and the hypothesis that $\ell^3 \nmid \#E(\mathbb{F}_{p^k})$ is not necessary. Conversely, since $G_0 \subseteq E(\mathbb{F}_{p^k})$ and $\ell^2 \mid \#G_0$, we have $\ell^2 \mid E(\mathbb{F}_{p^k})$. Furthermore, by the condition $\ell \mid p^k - 1$, it is obvious from Proposition 1 that $E[\ell] \subseteq E(\mathbb{F}_{p^k})$. On the other hand, since $\ell^3 \nmid \#E(\mathbb{F}_{p^k})$ and $G_0 \subseteq E(\mathbb{F}_{p^k})$, there is no cyclic subgroup of order ℓ^2 contained in G_0. By the condition that $\ell^2 \mid \#G_0$, we get $E[\ell] \subseteq G_0$, which completes the proof.* ■

Proposition 3 induces an efficient way to determine the group structure of $\mathbb{G}_0$ for ordinary elliptic curves with j-invariant 0 or 1728 under a weak condition.

IV. Main results

Based on the analysis in Section III, the group G_0 is isomorphic to $\mathbb{Z}_m \oplus \mathbb{Z}_{mnr}$ for some $m, n \in \mathbb{Z}$. We use H to denote mG_0. Then H is a cyclic group of order nr. Let the mapping τ act as

$$\tau(Q) = \big(\pi^k - 1)/\Phi_k(\pi)\big)(Q)$$

for all $Q \in E(\mathbb{F}_{p^k})$. We can see that $\tau(Q) \in G_0$. To summarize, mapping a random point of $E(\mathbb{F}_{p^k})$ to $\mathbb{G}_2$ can be performed as follows:

$$E(\mathbb{F}_{p^k}) \xrightarrow{\tau} G_0 \xrightarrow{m} H \xrightarrow{n} \mathbb{G}_2.$$

Since the action of τ on a random point of $E(\mathbb{F}_{p^k})$ only requires a few point additions and applications of the Frobenius endomorphism π, and the size of m is typically small, the most significant cost of hashing to $\mathbb{G}_2$ is the scalar multiplication by n. In this section, we show how to perform the cofactor multiplication efficiently.

A. Method I: Generalized Fuentes $et\ al.$ method

It is well-known that efficiently computable endomorphisms are a powerful tool to accelerate elliptic curve scalar multiplication [20], [22]. This technique was further used by Fuentes $et\ al.$ to reduce the overhead of hashing to $\mathbb{G}_2$ for curves admitting a twist [19]. However, few research studied the application of the technique for hashing to $\mathbb{G}_2$ on curves with the lack of twists in existing literature. Since the endomorphism π on an original curve plays a similar role as the "untwist-Frobenius-twist" endomorphism [21] on its twist, it seems that the Fuentes $et\ al.$ method also can be applied into pairing-friendly curves with the lack of twists. But there is a few details left to sort out in practice. First, it is necessary to confirm that $\pi(P) \in H$ for all $P \in H$. Moreover, one also needs to determine the value a satisfying that $\pi(P) = aP$ if $\pi(P) \in H$. In this subsection, we solve the above two questions and thus generalize the Fuentes $et\ al.$ method on pairing-friendly curves with the lack of twists.

Lemma 1 *Let notation be as above. Let $g(\pi) = \pi^2 - t\pi + p$ be the characteristic polynomial of the Frobenius endomorphism π. For all $P \in H$, there exists an integer a such that $\pi(P) = aP$. Furthermore, the integer a is one of solutions of the linear congruence equation*

$$a_0 + a_1 x \equiv 0 \bmod nr, \tag{5}$$

where the integers a_0 and a_1 are determined by the following congruence equation

$$\Phi_k(\pi) \equiv a_0 + a_1\pi \bmod g(\pi).$$

Proof *Without loss of generality, we regard P as a generator of H. Apparently, the order of $\pi(P)$ divides the order of P. On the other hand, since $P = \pi^{k-1}(\pi(P))$ the order of P also divides the order of $\pi(P)$. Therefore, both P and $\pi(P)$ are points of order nr. Since*

$$P \in H \subseteq G_0 \cong \mathbb{Z}_m \oplus \mathbb{Z}_{mnr},$$

then we have $mR = P$ for some $R \in G_0$. It is easy to deduce that $m\pi(R) = \pi(P)$. Furthermore,

$$\Phi_k(\pi)\big(\pi(R)\big) = \pi\big(\Phi_k(\pi)(R)\big) = \mathcal{O}_E,$$

which indicates that $\pi(R) \in G_0$. In total, there exists a point $\pi(R) \in G_0$ such that $m\pi(R) = \pi(P)$. By the definition of H, we can see that $\pi(P) \in H$ and thus $\pi(P)$ is a generator of H. It means that the endomorphism π acting on H corresponds to a scalar multiplication. In other words, there exists an integer a such that $\pi(P) = aP$ for all $P \in H$.

By the Euclidean algorithm, there exists a polynomial $u(\pi) \in \mathbb{Z}[\pi]$ such that

$$\Phi_k(\pi) = u(\pi) \cdot g(\pi) + r(\pi), \tag{6}$$

where $r(\pi) = a_0 + a_1\pi$. Moreover, the Frobenius endomorphism π on H satisfies the relations

$$\Phi_k(\pi) = 0, \quad g(\pi) = 0. \tag{7}$$

Putting Equations (6) and (7) together, we deduce that

$$(a_0 + a_1 \cdot a)P = r(\pi)(P) = \Phi_k(\pi)(P) - u(\pi)\big(g(\pi)(P)\big) = \mathcal{O}_E.$$

Since the order of the point P is nr, we conclude that $a_0 + a_1 \cdot a \equiv 0 \bmod nr$, which completes *the proof.* ■

Lemma 1 explains the effect of the endomorphism π on the group H. Let x_0 be a particular solution of the linear congruence equation (5) and $d = \gcd(a_1, nr)$. By Lemma 1, the integer a would be one of

$$\{x_0, x_0 + \frac{nr}{d}, \cdots, x_0 + (d-1)\frac{nr}{d}\}.$$

In fact, it is easy to determine which one is the solution. We can search the integer i between 0 and $d-1$ which makes $a = x_0 + i\frac{nr}{d}$, and then $\pi(P) = (x_0 + i\frac{nr}{d})P$. Now we are in a position to generalize the Fuentes $et\ al.$ method.

Theorem 2 *Let $E/\mathbb{F}_p$ be an ordinary elliptic curve with the lack of twists. Let G_0 be the cyclotomic zero subgroup of $E(\mathbb{F}_{p^k})$, and H the cyclic subgroup of G_0 of order nr. Then there exists a polynomial*

$$h(z) = h_0 + h_1 z + \cdots + h_{\varphi(k)-1} z^{\varphi(k)-1} \in \mathbb{Z}[z]$$

and an efficiently computable endomorphism ψ such that $h(\psi)(P) \in \mathbb{G}_2$ for all $P \in H$, where $|h_i| < |n|^{1/\varphi(k)}$ for $i = 0, \cdots, \varphi(k) - 1$.

Proof *Taking $\psi = \pi$, it can be obtained from Lemma 1 that $\psi(P) = aP$ for all $P \in H$. Since the order of ψ is precisely k restricted in the group H, we conclude that*

$$\Phi_k(a) \equiv 0 \bmod nr.$$

Similar to the proof in [19, Theorem 1], *there exists a polynomial in*

$$h(z) = h_0 + h_1 z + \cdots + h_{\varphi(k)-1} z^{\varphi(k)-1} \in \mathbb{Z}[z]$$

such that $h(a)$ is a multiple of n, where $|h_i| < |n|^{1/\varphi(k)}$. Therefore, we have $h(\psi)P \in \mathbb{G}_2$ for all $P \in H$, which completes the proof. ■

Applying the LLL algorithm [27] one can obtain a short coefficient vector $(h_0, \cdots, h_{\varphi(k)-1})$ in the following $\varphi(k)$-dimensional lattice:

$$\begin{bmatrix} n & 0 & 0 & \cdots & 0 \\ -a & 1 & 0 & \cdots & 0 \\ -a^2 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & & \ddots & \\ -a^{\varphi(k)-1} & 0 & \cdots & 0 & 1 \end{bmatrix}.$$

As mentioned in Section IV, the computational cost of hashing to $\mathbb{G}_2$ on pairing-friendly curves with the lack of twists is dominated by the scalar multiplication by n. Thus, the time complexity of Method I should be in $O\big(\log n/\varphi(k)\big)$. Apparently, this method is significantly faster than the standard one.

B. Method II

If $j(E) = 0$ or 1728, there exists another efficiently computable endomorphism ϕ that is given in Section II. Similarly, the endomorphism ϕ also corresponds to a scalar multiplication restricted in the group H. We summarize this observation as follows.

Lemma 2 *Let notation be as above. Let $t^2 - 4p = -Df^2$, where $D, f \in \mathbb{Z}$ and D is square free. If $j(E) = 0$ or 1728, then $\phi(P) = bP$ for all $P \in H$, where*

$$b = \begin{cases} \dfrac{-f \pm (2a-t)}{2f} \bmod nr & \text{if } j(E) = 0, \\ \dfrac{\pm(2a-t)}{2f} \bmod nr & \text{if } j(E) = 1728. \end{cases}$$

Proof *We only give the proof for the case j(E) = 0 (The proof of the remaining case is similar). Since E is ordinary and $j(E) = 0$, we have $D = 3$. Furthermore, since the value a in Lemma 1 is one of solutions of the quadratic congruence equation*

$$x^2 - tx + p \equiv 0 \bmod nr,$$

we get

$$a \equiv \frac{1}{2}(t \pm \sqrt{t^2 - 4p}) \equiv \frac{1}{2}(t \pm f\sqrt{-3}) \bmod nr,$$

which implies

$$\sqrt{-3} \equiv \pm(2a-t)/f \bmod nr.$$

On the other hand, since ϕ satisfies the quadratic relation

$$\phi^2 + \phi + 1 = 0,$$

we have $b^2 + b + 1 \equiv 0 \bmod nr$. It yields that

$$b = \frac{-1 \pm \sqrt{-3}}{2} = \frac{-f \pm (2a - t)}{2f} \bmod nr,$$

which completes the proof. ■

Putting Lemmas 1 and 2 together, we get the following theorem.

Theorem 3 *Let $E/\mathbb{F}_p$ be an ordinary elliptic curve with the lack of twists. Let G_0 be the cyclotomic zero subgroup of $E(\mathbb{F}_{p^k})$, and H the cyclic subgroup of G_0 of order nr. If $j(E) = 0$ or 1728, then there exists a polynomial*

$$h(z) = h_0 + h_1 z + \cdots + h_{2\varphi(k)-1} z^{2\varphi(k)-1} \in \mathbb{Z}[z]$$

and an efficiently computable endomorphism ψ such that $h(\psi)(P) \in \mathbb{G}_2$ for all $P \in H$, where $|h_i| < |n|^{1/(2\varphi(k))}$ for $i = 0, \cdots, 2\varphi(k) - 1$.

Proof *If $j(E) = 0$ (resp.1728), we take $\psi = \pi \circ \phi$, where ϕ is given in Section II. Combining Lemmas 1 and 2, we have $\psi(P) = \lambda P$ for all $P \in H$, where $\lambda = a \cdot b \bmod nr$. Moreover, we can find that $\gcd(k, 3) = 1$ (resp. $\gcd(k, 4) = 1$). Otherwise, the curve E admits a twist of degree 3 (resp. 2). Hence, the order of ψ is precisely $3k$ (resp. $4k$) restricted in H. It means that the integer λ satisfies that*

$$\Phi_{3k}(\lambda) \equiv 0 \bmod nr (resp.\ \Phi_{4k}(\lambda) \equiv 0 \bmod nr).$$

Since the degree of the cyclotomic polynomial is $2\varphi(k)$, there exists a polynomial

$$h(z) = h_0 + h_1 z + \cdots + h_{2\varphi(k)-1} z^{2\varphi(k)-1} \in \mathbb{Z}[z]$$

such that $h(\lambda)$ is a multiple of n, where $|h_i| < |n|^{1/(2\varphi(k))}$. From this result, we can conclude $h(\psi)P = h(\lambda)P \in \mathbb{G}_2$ for all $P \in H$, which completes the proof. ■

Likewise, applying the LLL algorithm one can obtain a $2\varphi(k)$-dimensional coefficient vector $(h_0, \cdots, h_{2\varphi(k)-1})$ for curves with j-invariant 0 or 1728. In this situation, the time complexity

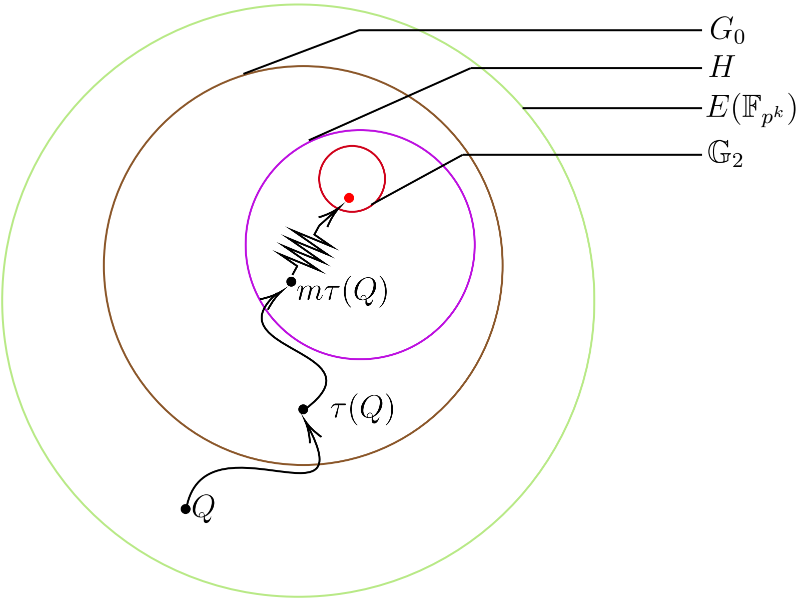

Fig. 1. Mapping a random point Q of $E(\mathbb{F}_{p^k})$ to $\mathbb{G}_2$ on pairing-friendly curves with the lack of twists

of hashing to $\mathbb{G}_2$ can be further reduced to $O\big(\log n/(2\varphi(k))\big)$. In order to give readers a clear understanding of the proposed approaches, the process of mapping a random point of $E(\mathbb{F}_{p^k})$ into $\mathbb{G}_2$ is illustrated in Fig.1.

V. Applications

In this section, we give a detailed description for hashing to $\mathbb{G}_2$ on the *BW13-P310* and *BW19-P286* curves. As a by-product, we also apply Method II into *BW25-P663*, which was studied by Enge and Milan [14]. All of the three curves are defined by the equation $y^2 = x^3 + b$ and parameterized by u as follows [17, Construction 6.6]:

$$p = \frac{1}{3}(u+1)^2(u^{2k} - u^k + 1) - u^{2k+1},$$

$$r = \Phi_{6k}(u),$$

$$t = -u^{k+1} + u + 1.$$

In Table I, we summarize the important parameters for the three curves.

The hash function $\mathcal{H}_{\mathbb{G}_2} : \{0,1\}^* \to \mathbb{G}_2$ is one of building blocks of the DAA scheme proposed in [36]. In fact, it consists of the following three phases:

(1) hashing an arbitrary string to $\mathbb{F}_p^*$ using a standard cryptographic hash function;

TABLE I
Parameters for the *BW13-P310*, *BW19-P286* and *BW25-P663* curves.

Curve	k	seed u	b
BW13-P310	13	-2224	-17
BW19-P286	19	-145	31
BW25-P663	25	6995	31

(2) mapping a random element of $\mathbb{F}_p^*$ to $E(\mathbb{F}_{p^k})$ in constant-time;

(3) mapping a random element of $E(\mathbb{F}_{p^k})$ to $\mathbb{G}_2$ in constant-time.

The mapping involved in the phase (2) is also referred to as encoding function in [13], which can be constructed by the Shallue-van de Woestijne (SVW) method [32]. A specialization of the SVW method on BN curves [4] were presented by Fouque and Tibouchi [16]. In fact, these techniques also can be applied to the above three curves. See Algorithm 1 for details. Note that $\chi_p()$ and $\chi_{p^k}()$ represent the quadratic residuosity testing functions in $\mathbb{F}_p$ and $\mathbb{F}_{p^k}$, respectively. The computation cost of Algorithm 1 is of about one quadratic residuosity testing in $\mathbb{F}_p$, two quadratic residuosity testings, one square root, and a few multiplications in $\mathbb{F}_{p^k}$.

To map a random point of $E(\mathbb{F}_{p^k})$ to $\mathbb{G}_2$ using the proposed techniques, we first need to determine the order of the group H that is defined in Section IV. By Proposition 2, we find that

$$\#G_0 = \begin{cases} \#E(\mathbb{F}_{p^k})/(\#E(\mathbb{F}_p)) & \text{if } k = 13 \text{ or } 19, \\ \#E(\mathbb{F}_{p^k})/(\#E(\mathbb{F}_{p^5})) & \text{if } k = 25. \end{cases}$$

Algorithm 1 Indifferentiable mapping to $E(\mathbb{F}_{p^k})$ for the *BW13-P310*, *BW19-P286* and *BW25-P663* curves

Input: $t \in \mathbb{F}_p^*$, the curve parameter $b \in \mathbb{F}_p^*$

Output: a point $P \in E(\mathbb{F}_{p^k})$

```
$sr3 \leftarrow \sqrt{-3}$; //Precomputation
$j \leftarrow (sr3 - 1)/2$;
$t_0 \leftarrow 0 + t \cdot \alpha + 0 \cdot \alpha^2 + \cdots + 0 \cdot \alpha^{k-1} \in \mathbb{F}_{p^k}$
$w \leftarrow sr3 \cdot \frac{t_0}{1+b+t_0^2}$;
$x_0 \leftarrow j - t_0 \cdot w$;
$x_1 \leftarrow -1 - x_0$;
$x_2 \leftarrow 1 + 1/w^2$;
```

```
$r_0, r_1, r_2 \xleftarrow{\$} \mathbb{F}_p^*$;
$a_0 \leftarrow \chi_{p^{13}}(r_0^2 \cdot (x_0^3 + b))$;
$a_1 \leftarrow \chi_{p^{13}}(r_1^2 \cdot (x_1^3 + b))$;
$i \leftarrow (a_0 - 1) \cdot a_1 \bmod 3 + 1$;
return $P \leftarrow (x_i, \chi_p(r_2^2 \cdot t) \cdot \sqrt{x_i^3 + b}, )$;
```

It is easy to check that $\gcd(p^k - 1, \#G_0) = r$ and $r^2 \nmid \#G_0$ for each curve. Thus, it can be deduced from Proposition 3 that

$$H = G_0 \cong \mathbb{Z}_{nr}.$$

Given a random point $Q \in E(\mathbb{F}_{p^k})$, then $P = \tau(Q) \in H$ and thus $nP \in \mathbb{G}_2$, where $\tau = (\pi^k - 1)/\Phi_k(\pi)$. In the following, we discuss how to map P into $\mathbb{G}_2$ using the two methods reported in Section IV.

Method I:

We first determine the integer a such that $\pi(Q) = aQ$ for all $Q \in H$ using Lemma 1. Constructing the lattice, we then obtain the vector $(h_0, h_1, \cdots, h_{\varphi(k)-1})$ such that

$$h(a) = h_0 + h_1 a + \cdots + h_{\varphi(k)-1} a^{\varphi(k)-1}$$

is a multiple of n. For the *BW13-P310* or *BW19-P286* curve, each term of h_i for $i = 0, 1, \cdots \varphi(k) - 1$ is given as follows

$$h_i = \begin{cases} (-1)^m \left(u^{2k-4-i} - u^{2k-5-i} + u^{2k-6-i} + 2u^{k-4-i} - 2u^{k-5-i} - u^{k-6-i}\right) - \ell_{k-1} & \text{if } i = 3m, 0 \le m < s-1, \\ (-1)^m \left(2u^{k-4-i} + u^{k-5-i} - u^{k-6-i}\right) - \ell_{k-1} & \text{if } i = 3m+1, 0 \le m < s-1, \\ (-1)^{m+1} \left(u^{2k-4-i} - u^{2k-5-i} + u^{2k-6-i} - 3u^{k-5-i}\right) - \ell_{k-1} & \text{if } i = 3m+2, 0 \le i < s-1, \\ -u^{2k} + u^{2k-1} - u^{2k-2} + 3u^{k-1} - 3 - \ell_{k-1} & \text{if } i = 3s-3, \\ -u^{2k-1} + u^{2k-2} - u^{2k-3} - 2u^{k-1} + 2u^{k-2} + u^{k-3} - \ell_{k-1} & \text{if } i = 3s-2, \\ -2u^{k-2} - u^{k-3} + u^{k-4} - \ell_{k-1} & \text{if } i = 3s-1. \end{cases}$$

where $s = (k-1)/3$ and $\ell_{k-1} = u^{2k-3} - u^{2k-4} + u^{2k-5} - 3u^{k-4}$. Since $P \in G_0 \subseteq T_k$, we have $Tr(P) = \mathcal{O}_E$. Thus,

$$h(a)P = h(\pi)P = \sum_{i=0}^{k-2} \pi^i (h_i P) = \sum_{i=0}^{k-1} \pi^i (\ell_i P),$$

where $\ell_i = h_i + \ell_{k-1}$ for $i = 0, 1, \cdots, k-2$.

In order to compute $\ell_i P$ for each i, the following scalar multiplications are performed:

$$P \to uP \to u^2P \cdots \to u^{k+1}P \to (2u^2+u)P \to (2u^5+u^4)P \cdots \to (2u^{k-2}+u^{k-3})P$$

On this basis, we then calculate R_i and H_i for $i = 0, 1, \cdots s-1$, where

$$R_i = (2u^{3i+2}+u^{3i+1})P - u^{3i}P, H_i = 2u^{3i+3}P - (2u^{3i+2}P + u^{3i+1})P.$$

The above calculation is done at a cost of $k+1$ scalar multiplications by u, $2s$ point doublings and $3s$ point additions. We denote L_i by $u^i(u^{k+1}-u^k+u^{k-1})P$ for $i = 0, 1, \cdots k-1$, which can be obtained as follows

$$(u^{k+1}-u^k+u^{k-1})P \to u(u^{k+1}-u^k+u^{k-1})P \cdots \to u^{k-1}(u^{k+1}-u^k+u^{k-1})P.$$

Afterwards, we can calculate $\ell_i P$ for $i = 0, 1, \cdots, k-1$, where

$$\ell_i P = \begin{cases} (-1)^m \big(L_{k-5-i} + H_{s-2-m}\big) & \text{if } i{=}3m, 0 \le m < s-1, \\ (-1)^m R_{s-2-m} & \text{if } i{=}3m{+}1, 0 \le i < s-1, \\ (-1)^{m+1}\big(L_{k-5-i} - 3u^{k-5-i}P\big) & \text{if } i{=}3m{+}2, 0 \le i < s-1, \\ -L_{k-1} + 3(u^{k-1}P - P) & \text{if } i{=}3s-3, \\ -L_{k-2} - H_{s-1} & \text{if } i{=}3s-2, \\ -R_{s-1} & \text{if } i{=}3s-1, \\ L_{k-4} - 3u^{k-4}P & \text{if } i{=}3s. \end{cases}$$

The calculations of L_i and $\ell_i P$ for $i = 0, 1, \cdots, k-1$ require $k-1$ scalar multiplications by u, $s+1$ point doublings and $3s+5$ point additions. Finally, the operation

$$h(\pi)P = \sum_{i=0}^{k-1} \pi^i(\ell_i P)$$

includes the computation of $k-1$ point additions and $k-1$ applications of the endomorphism π.

Since $s = (k-1)/3$, Method I totally requires $2k$ scalar multiplications by u, k point doublings and $3k+2$ point additions, one application of the endomorphism τ and $k-1$ applications of the endomorphism π for the *BW13-P310* or *BW19-P286* curve. Clearly, Method I is also suitable for the *BW25-P663* curve.

Method II:

Putting Lemmas 1 and 2 together, it is easy to calculate the integer λ such that $\psi(P) = \lambda P$ for all $P \in H$, where $\psi = \pi \circ \phi$. Applying LLL algorithm, we then obtain a $2\varphi(k)$-dimensional vector $(h_0, \cdots, h_{2\varphi(k)-1})$ such that

$$h(\lambda) = h_0 + h_1\lambda + \cdots + h_{2\varphi(k)-1}\lambda^{2\varphi(k)-1}$$

is a multiple of n, which implies that $h(\psi)(P) = h(\lambda)P \in \mathbb{G}_2$. Specifically, for the *BW13-P310*, *BW19-P286* or *BW25-P663* curve, each term of h_i for $i = 0, \cdots, 2\varphi(k) - 1$ can be expressed as follows:

$$h_i = \begin{cases} 0 & \text{if } k+2 \leq i \leq 2\varphi(k)-1, \\ 2 & \text{if } i = k+1, \\ u^2 - u + 1 & \text{if } i = k-1, \\ -\, uh_{i+1} & \text{if } 2 \leq i \leq k-2, \\ -\, uh_2 + 1 & \text{if } i = 1, \\ \sum_{j=0}^{(k-4)/3} ifh_{3j+1} - \sum_{j=1}^{(k-1)/3} h_{3j} - u & \text{if } i = 0, \\ \sum_{j=0}^{(k-4)/3} h_{3j+1} - \sum_{j=1}^{(k-1)/3} h_{3j-1} - 2u + 1 & \text{if } i = k. \end{cases}$$

In order to compute $h(\psi)P$, we first perform the following sequence of calculations:

$$P \to uP \to (u-1)P \to u^2P \to (u^2 - u + 1)P.$$

Then we have

$$h(\psi)P = \sum_{i=0}^{k+1} \psi^i(R_i),$$

TABLE II
Comparison between the operation count of Method I and Method II.

Curve	Method I	Method II
BW13-P310	$26Z + 13D + 41A + 1\tau + 12\pi$	$14Z + 1D + 30A + 1\tau + 14\psi$
BW19-P286	$38Z + 19D + 59A + 1\tau + 18\pi$	$20Z + 1D + 42A + 1\tau + 20\psi$
BW25-P663	$-$	$26Z + 1D + 54A + 1\tau + 26\psi$

where R_i for $i = 0, \cdots, k+1$ satisfies

$$\begin{aligned}
&R_{k+1} = 2P,\\
&R_{k-1} = (u^2 - u + 1)P,\\
&R_i = -uR_{i+1},\ 2 \leq i \leq k-2,\\
&R_1 = -uR_2 + P,\\
&R_0 = (R_1 + R_4 + \cdots + R_{k-3} - uP) - (R_3 + R_6 + \cdots + R_{k-1}),\\
&R_{13} = (R_1 + R_4 + \cdots + R_{k-3} - uP) - (R_2 + R_5 \cdots + R_{k-2}) - (u-1)P.
\end{aligned}$$

In total, it requires $k+1$ scalar multiplications by u, one point doubling, $2k+4$ point additions, one application of the endomorphism τ and $k+1$ applications of the endomorphism ψ.

Let Z, D and A denote the cost of a scalar multiplication by u, point doubling and point addition, respectively. In Table II, we present the operation counts of the two methods.

A. *Implementation results*

Magma [7] implementation is provided in https://github.com/eccdaiy39/hashing-magma to ensure the two methods are correct. In order to further illustrate the performance benefits resulting from Method II, we also implemented the two methods within the RELIC library [1] on the *BW13-P310* curve. We benchmarked our implementations on a 64-bit Intel Core i7-8550U @1.8GHz processor running Ubuntu 18.04.1 LTS with TurboBoost and hyper-threading features disabled. The open resource code is available at https://github.com/eccdaiy39/hashing. In Table III, we give the timing benchmark results averaged over 10,000 executions. The results show that Method II is faster than Method I by about 93.1% for mapping a random point of $E(\mathbb{F}_{p^k})$ to $\mathbb{G}_2$ on the *BW13-P310* curve. For the whole procedure of hashing to $\mathbb{G}_2$, the proposed method leads to a roughly 77.0% speedup.

TABLE III
Comparison of the running timing for the components of hashing to $\mathbb{G}_2$ on the *BW13-P310* curve.

Phase	Method	Clock cycles($\times 10^4$)
Hashing to $E(\mathbb{F}_{p^{13}})$	SVW	320
Map-point-to-$\mathbb{G}_2$	I	2906
Map-point-to-$\mathbb{G}_2$	II	1505
hashing to $\mathbb{G}_2$	SVW+I	3225
hashing to $\mathbb{G}_2$	SVW+II	1822

VI. Conclusion

The *BW13-P310* and *BW19-P286* curves become competitive in DAA schemes as fast group operation in $\mathbb{G}_1$. In this work, we investigated the problem of hashing to $\mathbb{G}_2$ on the two curves in detail. Several interesting techniques were proposed, which actually can be applied to a large class of curves with the lack of twists. A software implementation were presented on the *BW13-P310* curve to confirm the efficiency of the proposed techniques. The results show that hashing to $\mathbb{G}_2$ can be implemented efficiently on this curve.

Acknowledgement

This work is supported by Guangdong Major Project of Basic and Applied Basic Research(No. 2019B030302008) and the National Natural Science Foundation of China(No.s 61972429 and 61972428).

References

[1] D. F. Aranha and C. P. L. Gouvêa, "RELIC is an Efficient LIbrary for Cryptography," https://github.com/relic-toolkit/relic.

[2] D. F. Aranha, E. Pagnin, and F. Rodríguez-Henríquez, "Love a pairing," in *Progress in Cryptology – LATINCRYPT 2021*, P. Longa and C. Ràfols, Eds. Cham: Springer International Publishing, 2021, pp. 320–340.

[3] R. Balasubramanian and N. Koblitz, "The improbability that an elliptic curve has subexponential discrete log problem under the menezes-okamoto-vanstone algorithm," *Journal of Cryptology*, vol. 11, no. 2, pp. 141–145, 1998.

[4] P. S. L. M. Barreto and M. Naehrig, "Pairing-friendly elliptic curves of prime order," in *Selected Areas in Cryptography – SAC 2005*, B. Preneel and S. Tavares, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 319–331.

[5] I. F. Blake, G. Seroussi, N. P. Smart, and et al., *Advances in elliptic curve cryptography*, ser. London Mathematical Society Student Texts. New York: Cambridge University Press, 2005, vol. 317.

[6] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing," in *Advances in Cryptology — CRYPTO 2001*, J. Kilian, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 213–229.

[7] W. Bosma, J. Cannon, and C. Playoust, "The Magma algebra system. I. The user language," *J. Symbolic Comput.*, vol. 24, no. 3-4, pp. 235–265, 1997, computational algebra and number theory (London, 1993).

[8] E. Brickell, L. Chen, and J. Li, “A new direct anonymous attestation scheme from bilinear maps,” in *Trusted Computing - Challenges and Applications – Trust 2008*, P. Lipp, A.-R. Sadeghi, and K.-M. Koch, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 166–178.

[9] A. Budroni and F. Pintore, “Efficient hash maps to $\mathbb{G}_2$ on BLS curves,” *Applicable Algebra in Engineering, Communication and Computing*, vol. 33, no. 3, pp. 261–281, 2022.

[10] R. Clarisse, S. Duquesne, and O. Sanders, “Curves with fast computations in the first pairing group,” in *Cryptology and Network Security – CANS 2020*, S. Krenn, H. Shulman, and S. Vaudenay, Eds. Cham: Springer International Publishing, 2020, pp. 280–298.

[11] Y. El Housni and A. Guillevic, “Families of snark-friendly 2-chains of elliptic curves,” in *Advances in Cryptology – EUROCRYPT 2022*, O. Dunkelman and S. Dziembowski, Eds. Cham: Springer International Publishing, 2022, pp. 367–396.

[12] Y. El Housni, A. Guillevic, and T. Piellard, “Co-factor clearing and subgroup membership testing on pairing-friendly curves,” Cryptology ePrint Archive, Report 2022/352, 2022, https://ia.cr/2022/352.

[13] N. El Mrabet and M. Joye, *Guide to pairing-based cryptography*. New York: Chapman and Hall/CRC, 2016.

[14] A. Enge and J. Milan, “Implementing cryptographic pairings at standard security levels,” in *Security, Privacy, and Applied Cryptography Engineering – SPACE 2014*, R. S. Chakraborty, V. Matyas, and P. Schaumont, Eds. Cham: Springer International Publishing, 2014, pp. 28–46.

[15] E. Fouotsa and L. Azebaze Guimagang, “Fast hashing to $\mathbb{G}_2$ on aurifeuillean pairing-friendly elliptic curves,” *SN Computer Science*, vol. 1, no. 1, p. 51, 2019.

[16] P.-A. Fouque and M. Tibouchi, “Indifferentiable hashing to barreto–naehrig curves,” in *Progress in Cryptology – LATINCRYPT 2012*, A. Hevia and G. Neven, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 1–17.

[17] D. Freeman, M. Scott, and E. Teske, “A taxonomy of pairing-friendly elliptic curves,” *Journal of Cryptology*, vol. 23, no. 2, pp. 224–280, 2010.

[18] G. Frey, “Applications of arithmetical geometry to cryptographic constructions,” in *Proceedings of the 5th International Conference on Finite Fields and Applications(Augsburg, 1999)*, D. Jungnickel and H. Niederreiter, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 128–161.

[19] L. Fuentes-Castañeda, E. Knapp, and F. Rodríguez-Henríquez, “Faster hashing to $\mathbb{G}_2$,” in *Selected Areas in Cryptography – SAC 2011*, A. Miri and S. Vaudenay, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 412–430.

[20] S. D. Galbraith, X. Lin, and M. Scott, “Endomorphisms for faster elliptic curve cryptography on a large class of curves,” in *Advances in Cryptology - EUROCRYPT 2009*, A. Joux, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 518–535.

[21] S. D. Galbraith and M. Scott, “Exponentiation in pairing-friendly groups using homomorphisms,” in *Pairing-Based Cryptography – Pairing 2008*, S. D. Galbraith and K. G. Paterson, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 211–224.

[22] R. P. Gallant, R. J. Lambert, and S. A. Vanstone, “Faster point multiplication on elliptic curves with efficient endomorphisms,” in *Advances in Cryptology — CRYPTO 2001*, J. Kilian, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 190–200.

[23] E. Gorla and M. Massierer, “Point compression for the trace zero subgroup over a small degree extension field,” *Designs, Codes and Cryptography*, vol. 75, no. 2, pp. 335–357, 2015.

[24] ——, “An optimal representation for the trace zero subgroup,” *Designs, Codes and Cryptography*, vol. 83, no. 3, pp. 519–548, 2017.

[25] J. Groth, “On the size of pairing-based non-interactive arguments,” in *Advances in Cryptology – EUROCRYPT 2016*, M. Fischlin and J.-S. Coron, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 305–326.

[26] F. Hess, N. P. Smart, and F. Vercauteren, “The Eta pairing revisited,” *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4595–4602, 2006.

[27] A. K. Lenstra, H. W. Lenstra, and L. Lovász, “Factoring polynomials with rational coefficients,” *Mathematische Annalen*, vol. 261, no. 4, 1982.

[28] R. Lidl and H. Niederreiter, *Introduction to finite fields and their applications*. Cambridge: Cambridge University Press, 1994.

[29] V. S. Miller, “The weil pairing, and its efficient calculation,” *Journal of Cryptology*, vol. 17, no. 4, pp. 235–261, 2004.

[30] R. Schoof, “Nonsingular plane cubic curves over finite fields,” *Journal of combinatorial theory, Series A*, vol. 46, no. 2, pp. 183–211, 1987.

[31] M. Scott, N. Benger, M. Charlemagne, L. J. Dominguez Perez, and E. J. Kachisa, “Fast hashing to $\mathbb{G}_2$ on pairing-friendly curves,” in *Pairing-Based Cryptography – Pairing 2009*, H. Shacham and B. Waters, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 102–113.

[32] A. Shallue and C. E. van de Woestijne, “Construction of rational points on elliptic curves over finite fields,” in *Algorithmic Number Theory Symposium– ANTS 2006*, F. Hess, S. Pauli, and M. Pohst, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 510–524.

[33] J. H. Silverman, *The arithmetic of elliptic curves*, ser. Graduate Texts in Mathematics. New York: Springer-Verlag, 1992, vol. 106.

[34] R. S. Wahby and D. Boneh, “Fast and simple constant-time hashing to the bls12-381 elliptic curve,” *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2019, no. 4, p. 154–179, 2019.

[35] L. C. Washington, *Elliptic Curves: Number Theory and Cryptography*. New York: Chapman and Hall/CRC, 2008.

[36] K. Yang, L. Chen, Z. Zhang, C. J. Newton, B. Yang, and L. Xi, “Direct anonymous attestation with optimal TPM signing efficiency,” *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 2260–2275, 2021.