Fast Hashing to $\mathbb{G}_2$ on Pairing-friendly Curves with the Lack of Twists

Yu Dai[a], Fangguo Zhang[b,c] and Chang-An Zhao*[a,c]

[a]School of Mathematics, Sun Yat-sen University, Guangzhou, 510275, Guangdong, P.R.China
[b]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, 510006, Guangdong, P.R.China
[c]Guangdong Key Laboratory of Information Security, Guangzhou, 510006, Guangdong, P.R.China

ARTICLE INFO

Keywords:
Pairing based cryptography
Pairing-friendly curves with the lack of twists
Hashing to $\mathbb{G}_2$
BW13-P310 and *BW19-P286*.

ABSTRACT

Pairing-friendly curves with the lack of twists, such as *BW13-P310* and *BW19-P286*, have been receiving attention in pairing-based cryptographic protocols as they provide fast operation in the first pairing subgroup $\mathbb{G}_1$ at the 128-bit security level. However, they also incur a performance penalty for hashing to $\mathbb{G}_2$ simultaneously since $\mathbb{G}_2$ is totally defined over a full extension field. Furthermore, the previous methods for hashing to $\mathbb{G}_2$ focus on pairing-friendly curves admitting a twist, which can not be employed for our selected curves.

In this paper, we propose a general method for hashing to $\mathbb{G}_2$ on curves with the lack of twists. More importantly, we further optimize the general algorithm on curves with non-trival automorphisms, which is certainly suitable for *BW13-P310* and *BW19-P286*. Theoretical estimations show that the latter would be more efficient than the former. For comparing the performance of the two proposed algorithms in detail, high speed software implementation over *BW13-P310* is also provided on a 64-bit processor. Experimental results show that the general algorithm can be sped up by up to 88% if the computational cost of cofactor multiplication for $\mathbb{G}_2$ is only considered, while the improved method is up to 71% faster than the general one for the whole process.

1. Introduction

Pairings are a powerful mathematical tool to construct various cryptographic protocols with novel properties, such as identity-based encryption [5], direct anonymous attestation (DAA) [8, 9], and zero-knowledge succinct non-interactive argument of knowledge (zk-SNARK) [28, 14]. Cryptographic pairings are built on elliptic curves over finite fields. The following standard notation and settings are used throughout the paper. Let p be a large prime and E an elliptic curve over $\mathbb{F}_p$ defined by an equation of the form $y^2 = x^3 + ax + b$. 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 $\mathbb{G}_1 = E(\mathbb{F}_p)[r]$, $\mathbb{G}_2 = E[r] \cap \mathrm{Ker}(\pi - [p])$ and $\mathbb{G}_T$ is a subgroup of order r of $\mathbb{F}^*_{p^k}$.

In implementations of pairing-based protocols, the selection of pairing-friendly curves is closely related to particular requirements given by specifications of the considered application [12, 2, 13]. For example, pairing-friendly curves with fast group exponentiation in $\mathbb{G}_1$ are attractive to the DAA schemes. Before 2015, a 3072-bit finite field $\mathbb{F}_{p^k}$ with non-small characteristic offers the 128-bit security level under the attack of the number field sieve [27]. In this case, the Barreto-Naehrig (BN) curve [4] with embedding degree 12 over a 256-bit prime field provides optimal performance for both group exponentiation in $\mathbb{G}_1$ and pairing computation. However, with the propose of the tower number field sieve [32, 33], the difficulty of solving the discrete logarithm problem in finite fields with composite extension degrees decreased significantly. Therefore, many mainstream curves can not reach the desired security level anymore. Accodring to the estimate in [29], the size of prime field on BN curves must increase to 446 bits for achieving the updated 128-bit security level, which leads to low performance of group exponentiation in $\mathbb{G}_1$. In 2022, Clarisse *et al*. [12] recommended two curves with fast group exponentiation in $\mathbb{G}_1$ at the updated 128-bit security level: *BW13-P310* with embedding degree 13 over a 310-bit prime field, and *BW19-P286* with embedding degree 19 over a 286-bit

*Corresponding author.
daiy39@mail2.sysu.edu.cn (Y. Dai); isszhfg@mail.sysu.edu.cn (F. Zhang); zhaochan3@mail.sysu.edu.cn (C. Zhao*)
ORCID(s):

prime field. In addition, since the embedding degrees of the two curves are prime, they are immune to the attacks of the tower number field sieve. In other words, the two curves are potential for long-term security.

It is often necessary to hash binary strings to $\mathbb{G}_1$ or $\mathbb{G}_2$ in pairing-based cryptographic protocols. The standard approach for hashing to $\mathbb{G}_1$ is to hash to a random point of $E(\mathbb{F}_p)$ [40, 19, 42], followed by a scalar multiplication by the cofactor $h_1 = \#E(\mathbb{F}_p)/r$. The computational cost of the cofactor 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 [15], which may further reduce the computational cost for hashing to $\mathbb{G}_1$.

By contrast, hashing to $\mathbb{G}_2$ is relatively complicated. Denote by $\mathrm{Aut}(E)$ the automorphism group of E and let $d = \gcd(k, \#\mathrm{Aut}(E))$. If $d > 1$ and $r^2 \parallel E(\mathbb{F}_{p^k})$, there exists a unique d-twist E' of E such that $r \parallel \#E'(\mathbb{F}_{p^e})$ [30, 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 leads to an expensive cost for cofactor multiplication. In 2009, Scott *et al*. [39] proposed an efficient method to reduce the computational cost of the cofactor multiplication. This method was further optimized by Fuentes *et al*. [22]. Detailed comparisons of the two methods on different curves were given in [10, 18].

Unfortunately, we find that $\gcd(k, \#\mathrm{Aut}(E)) = 1$ on the *BW13-P310* and *BW19-P286* curves. It implies that $\mathbb{G}_2$ can be only represented as $E[r] \cap \mathrm{Ker}(\pi - [p])$. Following Enge and Milan [17], we refer to this type of curves as **curves with the lack of twists**. In fact, if $p \geq 5$, then $\#\mathrm{Aut}(E) \in \{2, 4, 6\}$ [41, Theorem III.10.1]. In this case, we can see that E is a curve with the lack of twists if and only if $\gcd(k, 6) = 1$. It indicates that the extra effort is required for adapting the above two approaches for hashing to $\mathbb{G}_2$ to pairing-friendly curves with the lack of twists.

Our contributions. In this paper, we investigate the problem of hashing to $\mathbb{G}_2$ on pairing-friendly curves with the lack of twists. Our contributions are mainly divided into the following two parts:

- A general approach for hashing to $\mathbb{G}_2$ is given for curves with the lack of twists. To this end, we first show how to efficiently map a random point of $E(\mathbb{F}_{p^k})$ into a certain cyclic subgroup H. Explicit formulas are also proposed for determining the order of H. We then determine the eigenvalue of the endomorphism π on this subgroup. After tackling these problems, we successfully extend the previous techniques of hashing to $\mathbb{G}_2$ to curves with the lack of twists. After that, an optimized approach is presented tailored to curves equipped with non-trivial automorphisms. Note that the series of *BW* curves used in this paper exactly meet this condition.
- We describe explicit steps for hashing to $\mathbb{G}_2$ on *BW13-P310* and *BW19-P286* based on the above two methods. High speed software implementation on *BW13-P310* is also presented to evaluate the performance of the proposed algorithms. Experimental results show that, using the lower prime field operations provided in the RELIC cryptographic toolkit [1] with Assembly language, the hash function built on the improved method is up to 71% faster than that on the general one. To the best of our knowledge, this is the first software implementation of hashing to $\mathbb{G}_2$ on *BW13-P310*.

Outline of the paper. The remainder of this paper is organized as follows. Section 2 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 3, 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 4. The application of the proposed technique on *BW13-P310* and *BW19-P286* is considered in Section 5. Finally, we draw our conclusion in Section 6.

2. Background

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

2.1. The standard approach for 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 $m \in \mathbb{Z}^+$, we 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 $t_1 = t$, the value t_m for any $m > 1$ can be obtained by evaluation the recursion [43, Lemma 4.13]

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

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:

$$Q \to R = cQ \to \mathrm{Tr}(R) - kR,$$

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

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

for all $S \in E(\mathbb{F}_{p^k})$. The cofactor multiplication maps Q into $E[r]$, and the endomorphism $\mathrm{Tr} - k$ forces R into $\mathbb{G}_2$ [23, Section IX]. It is obvious that the approach is extremely inefficient and shall not be admissible in practical applications.

2.2. Endomorphism ring of ordinary elliptic curves over finite fields

Let $t^2 - 4p = -Df^2$ $(D, f \in \mathbb{Z})$ where D is square free. 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 $\mathcal{O}$ satisfying that $\mathbb{Z} \subsetneq \mathcal{O} \subseteq O_K$. Any order $\mathcal{O}$ 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 $D_{\mathcal{O}}$ denote the discriminant of order $\mathcal{O}$, which is defined as

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

Let q be a power of the prime p. We define $\mathrm{End}_{\mathbb{F}_q}(E)$ and $\mathrm{End}(E)$ to be the endomorphism rings of E over $\mathbb{F}_q$ and $\bar{\mathbb{F}}_q$, respectively. Since E is ordinary, the ring $\mathrm{End}(E)$ is isomorphic to an order in K [43, Theorem 10.6]. Thus, we conclude that $\mathrm{End}_{\mathbb{F}_q}(E) \subseteq \mathrm{End}(E) \subseteq O_K$. Generally speaking, it is not straightforward to determine $\mathrm{End}_{\mathbb{F}_q}(E)$. However, the question becomes simple on some special ordinary elliptic curves. To be precise,

- if $j(E) = 0$, then $p \equiv 1 \bmod 3$ [43, Proposition 4.33]. There exists an endomorphism $\phi \in \mathrm{End}_{\mathbb{F}_q}(E)$ 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 $\mathrm{End}_{\mathbb{F}_q}(E) = O_K$ as $\mathbb{Z}[\phi] \subseteq \mathrm{End}_{\mathbb{F}_q}(E) \subseteq O_K$;
- if $j(E) = 1728$, then $p \equiv 1 \bmod 4$ [43, Theorem 4.23]. There exists an endomorphism $\phi \in \mathrm{End}_{\mathbb{F}_q}(E)$ 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 $\mathrm{End}_{\mathbb{F}_q}(E) = O_K$.

We can see that the discriminant of $\mathrm{End}_{\mathbb{F}_q}(E)$ 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.

2.3. Group structure of ordinary elliptic curves over finite fields

By the basic theory of elliptic curves over finite fields [43, 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, $E(\mathbb{F}_q)$ is cyclic if and only if $m_1 = 1$. 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 two possibilities: (a) $E[\ell] \subseteq E(\mathbb{F}_q)$; (b) $E(\mathbb{F}_q)$ has a cyclic subgroup of order ℓ^2. The following theorem provides necessary and sufficient conditions for $E[\ell] \subseteq E(\mathbb{F}_q)$.

Theorem 1. [37, 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 $\mathcal{O}(\frac{\tilde{t}^2 - 4q}{\ell^2}) \subseteq \mathrm{End}_{\mathbb{F}_q}(E)$, where the notation $\mathcal{O}(\frac{\tilde{t}^2 - 4q}{\ell^2})$ represents the order with discriminant $\frac{\tilde{t}^2 - 4q}{\ell^2}$.*

For curves with j-invariant 0 or 1728, the necessary condition of Theorem 1 can be further simplified as shown in Corollary 1. It should be noted that the following conclusion is well known to the experts, but we have been unable to find the relevant proof in the literature.

Corollary 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. For the curves with $j(E) = 0$ or 1728, it can be seen from Subsection 2.2 that $\mathrm{End}_{\mathbb{F}_q}(E) = O_K$ and thus the condition $\mathcal{O}(\frac{t^2-4q}{\ell^2}) \subseteq \mathrm{End}_{\mathbb{F}_q}(E)$ holds. By Theorem 1, the rest of the proof is immediate. □

Given the factorization of $\gcd(\#E(\mathbb{F}_q), q-1)$, Corollary 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 [36, Algorithm 3], which is slightly complicated. In the following sections we show how to apply the above results to improve the efficiency of hashing to $\mathbb{G}_2$ on pairing-friendly curves with the lack of twists.

3. 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. One can see that G_0 forms a group. By the definition of G_0, we clearly have $G_0 \subseteq T_k$, where T_k denotes the trace zero subgroup [21] of $E(\mathbb{F}_{p^k})$, i.e.,

$$T_k = \{Q \in E(\mathbb{F}_{p^k}) \mid \mathrm{Tr}(Q) = \mathcal{O}_E\}.$$

In particular, the two subgroups are identical if and only if the embedding degree k is a 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 1. *Let notation be as above. Then the order of the group G_0 is precisely equal to $\prod_{d|k} \#E(\mathbb{F}_{p^d})^{\mu(k/d)}$, where $\mu(.)$ is the Moebius function. In addition, if $r \nmid \Phi_k(1)$, then $E[r] \cap G_0 = \mathbb{G}_2$.*

Proof. By [35, 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 $\#G_0 = \prod_{d|k} \#E(\mathbb{F}_{p^d})^{\mu(k/d)}$.

We now prove that $E[r] \cap G_0 = \mathbb{G}_2$ under the condition $r \nmid \Phi_k(1)$. By the fact that $\mathbb{G}_2 \subseteq E[r] \cap G_0$, we only need to prove that $E[r] \cap G_0 \subseteq \mathbb{G}_2$. For any $R \in E[r] \cap G_0$, we have

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

by the definition of G_0. It follows from $R \in E[r]$ that there exist $m_1, m_2 \in \mathbb{Z}/r\mathbb{Z}$ such that

$$R = m_1 R_1 + m_2 R_2.$$

where R_1 and R_2 are generators of $\mathbb{G}_1$ and $\mathbb{G}_2$, respectively. Since $r \nmid \Phi_k(1)$ and $r \mid \Phi_k(p)$, we have

$$\Phi_k(\pi)(R_1) = \Phi_k(1)R_1 \neq \mathcal{O}_E,$$
$$\Phi_k(\pi)(R_2) = \Phi_k(p)R_2 = \mathcal{O}_E.$$

On the basis of the above equations, we obtain

$$\Phi_k(\pi)R = \Phi_k(\pi)(m_1 R_1 + m_2 R_2)$$

$$\begin{aligned}&=m_1\Phi_k(\pi)(R_1)+m_2\Phi_k(\pi)(R_2)\\&=m_1\Phi_k(1)R_1+m_2\Phi_k(p)R_2\\&=m_1\Phi_k(1)R_1\\&=\mathcal{O}_E.\end{aligned}$$

The last equality in the above equation indicates that $m_1 = 0$. This implies that $R = m_2R_2$, i.e., $R \in \mathbb{G}_2$ since R_2 is the generator of G_2. By the arbitrary choice of R, we conclude that $E[r] \cap G_0 \subseteq \mathbb{G}_2$, which completes the proof. □

It can be seen from Proposition 1 that once the order of $E(\mathbb{F}_{p^d})$ is given for each $d \mid k$, it is simple to calculate the order 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 2. *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 Corollary 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 can be obtained from Corollary 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 2 induces an efficient way to determine the group structure of G_0 for ordinary elliptic curves with j-invariant 0 or 1728 under a weak condition.

4. Main results

Based on the analysis in Section 3, 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.

4.1. General approach for curves with the lack of twists

It is well-known that efficiently computable endomorphisms are a powerful tool to accelerate elliptic curve scalar multiplication [26, 24]. This technique was further used by Fuentes *et al*. to reduce the overhead of hashing to $\mathbb{G}_2$ on curves admitting a twist [22]. However, few works in the literature study the application of the technique for hashing to $\mathbb{G}_2$ on curves with the lack of twists. Since the endomorphism π on an original curve plays a similar role as the untwist-Frobenius-twist endomorphism [25] on its twist, it seems that the Fuentes *et al*. method can be also applied to pairing-friendly curves with the lack of twists. But there are 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*

$$a_0 + a_1x \equiv 0 \bmod nr, \tag{3}$$

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, which means that the order of P is nr. Then we have $nr\pi(P) = \pi(nrP) = \mathcal{O}_E$ and thus 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 $R \in G_0$, meaning $m\pi(R) = \pi(P)$. 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

$$\Phi_k(\pi) = u(\pi) \cdot g(\pi) + r(\pi), \tag{4}$$

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

$$\Phi_k(\pi) = 0, \;\; g(\pi) = 0. \tag{5}$$

Putting Equations (4) and (5) 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 (3) 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, we can search through all i between 0 and $d-1$ for an $a = x_0 + i\frac{nr}{d}$ such that $\pi(P) = (x_0 + i\frac{nr}{d})P$. 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 [22, 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. □

To obtain a short coefficient vector $(h_0, \cdots, h_{\varphi(k)-1})$, one can apply the LLL algorithm [34] in the following $\varphi(k)$-dimensional lattice:

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

By Theorem 2, the number of point doublings for mapping a random point to $\mathbb{G}_2$ is around $\log n/\varphi(k))$. Apparently, this method is significantly faster than the standard one. In the following we denote this method by Method I.

4.2. Optimized version for curves with non-trivial automorphisms

If $j(E) = 0$ or 1728, there exists another efficiently computable endomorphism ϕ that is given in Section 2. 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 $i = 0, \cdots, 2\varphi(k) - 1$.

Table 1
Parameters for the *BW13-P310* and *BW19-P286*.

Curve	k	seed u	b
BW13-P310	13	−2224	−17
BW19-P286	19	−145	31

Proof. If $j(E) = 0$ (resp.1728), we take $\psi = \pi \circ \phi$, where ϕ is defined in Section 2. 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 conclude that $h(\psi)P = h(\lambda)P \in \mathbb{G}_2$ 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 number of point doublings for mapping a random point to $\mathbb{G}_2$ is reduced to around $\log n/(2\varphi(k))$. This improved method is also called as Method II in the rest of this paper.

5. Applications

In this section, we give a detailed description for constructing the hashing function $\mathcal{H}_{\mathbb{G}_2} : \{0,1\}^* \to \mathbb{G}_2$ on the *BW13-P310* and *BW19-P286* curves. Both curves are defined by an equation of the form $y^2 = x^3 + b$ and parameterized by u as follows[20, Construction 6.6]:

$$\begin{aligned} p &= \frac{1}{3}(u+1)^2(u^{2k} - u^k + 1) - u^{2k+1}, \\ r &= \Phi_{6k}(u), \\ t &= -u^{k+1} + u + 1. \end{aligned}$$

In Table 1, we summarize the important parameters for the three curves. The hash function $\mathcal{H}_{\mathbb{G}_2}$ is modeled as a random oracle in security proofs. In order to resist timing side-channel attacks, it should be implemented in constant-time [7, 31]. Specifically, it consists of the following three phases:

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

(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 [16], which can be constructed by the Shallue-van de Woestijne (SVW) method [40]. A specialization of the SVW method on BN curves [4] was presented by Fouque and Tibouchi [19]. In fact, these techniques also can be applied to the above three curves. See Algorithm 1 for details. Note that $\chi_p(\cdot)$ and $\chi_{p^k}(\cdot)$ represent the quadratic residuosity testing functions in $\mathbb{F}_p$ and $\mathbb{F}_{p^k}$, respectively. The computational 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 $\mathbb{F}_{p^k}$. Thanks to the odd embedding degrees on the target curves, one quadratic residuosity testing in $\mathbb{F}_{p^k}$ can be transformed into one quadratic residuosity testing in $\mathbb{F}_p$ and one trace mapping in $\mathbb{F}_{p^k}$. This observation is summarized in the following proposition.

Proposition 3. *Let $A \in \mathbb{F}_{p^k}$. If k is odd, then $\chi_{p^k}(A) = \chi_p(\mathrm{Tr}(A))$.*

Proof. If A is a square in $\mathbb{F}_{p^k}$, then there exists an element $B \in \mathbb{F}_{p^k}$ such that $A = B^2$. As a result, we have

$$\mathrm{Tr}(A) = \mathrm{Tr}(B^2) = \big(\mathrm{Tr}(B)\big)^2.$$

Since $\mathrm{Tr}(B) \in \mathbb{F}_p$, we can see that $\mathrm{Tr}(A)$ is a square in $\mathbb{F}_p$. Conversely, if A is a nonsquare in $\mathbb{F}_{p^k}$, we assume that $\mathrm{Tr}(A)$ is a square in $\mathbb{F}_p$. Then, there exists an element $C \in \mathbb{F}_p$ such that $\mathrm{Tr}(A) = C^2$. Since the embedding degree k is odd, the exponent $p + p^2 + \cdots + p^{k-1}$ is even, which implies that A is a square in $\mathbb{F}_{p^k}$ as

$$A = \frac{C^2}{A^{p+p^2+\cdots+p^{k-1}}} = \Big(\frac{C}{A^{(p+p^2+\cdots+p^{k-1})/2}}\Big)^2.$$

This is a contradiction, and so $\mathrm{Tr}(A)$ is a nonsquare in $\mathbb{F}_p$. Putting it all together, we conclude that $\chi_{p^k}(A) = \chi_p(\mathrm{Tr}(A))$, which completes the proof of the proposition. □

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 4. By Proposition 1, we have $\#G_0 = \#E(\mathbb{F}_{p^k})/(\#E(\mathbb{F}_p))$ on the two target curves. Using Magma [6], one can check that $\gcd(p^k-1, \#G_0) = r$ and $r^2 \nmid \#G_0$. Thus, it can be deduced from Proposition 2 that

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

Input: $t \in \mathbb{F}_p^*$, the curve parameter $b \in \mathbb{F}_p^*$
Output: a point $P \in E(\mathbb{F}_{p^k})$

1: $sr3 \leftarrow \sqrt{-3}$ //Precomputation
2: $j \leftarrow (sr3 - 1)/2$
3: $t_0 \leftarrow 0 + t \cdot \alpha + 0 \cdot \alpha^2 + \cdots + 0 \cdot \alpha^{k-1} \in \mathbb{F}_{p^k}$
4: $w \leftarrow sr3 \cdot \frac{t_0}{1+b+t_0^2}$
5: $x_1 \leftarrow j - t_0 \cdot w$
6: $x_2 \leftarrow -1 - x_1$
7: $x_3 \leftarrow 1 + 1/w^2$
8: $r_1, r_2, r_3 \xleftarrow{\$} \mathbb{F}_p^*$
9: $a_1 \leftarrow \chi_{p^k}(r_1^2 \cdot (x_1^3 + b))$
10: $a_2 \leftarrow \chi_{p^k}(r_2^2 \cdot (x_2^3 + b))$
11: $i \leftarrow (a_1 - 1) \cdot a_2 \bmod 3 + 1$
12: **return** $P \leftarrow (x_i, \chi_p(r_2^2 \cdot t) \cdot \sqrt{x_i^3 + b})$

$H = G_0 \cong \mathbb{Z}_{nr}$. 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 4.

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, the value of h_i for each i 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

$$\begin{aligned} R_i &= (2u^{3i+2} + u^{3i+1})P - u^{3i}P, \\ H_i &= 2u^{3i+3}P - (2u^{3i+2}P + u^{3i+1})P. \end{aligned}$$

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

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

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 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.

Method II:

Putting Lemmas 1 and 2 together, one can 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* or *BW19-P286* curve, the value of h_i for each i can be expressed as follows:

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

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

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

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

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

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 U, D and A denote the cost of a scalar multiplication by u, point doubling and point addition, respectively. In Table 2, we present the operation counts of the two methods.

5.1. Implementation results

Magma code is first provided to verify the correctness of the proposed methods. In order to further illustrate the performance benefits resulting from Method II, we also present high-speed software implementation for the two

Table 2
Comparison between the operation count of Method I and Method II.

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

Table 3
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	327
Map-point-to-$\mathbb{G}_2$	I	2586
Map-point-to-$\mathbb{G}_2$	II	1378
Hashing to $\mathbb{G}_2$	SVW+I	2913
Hashing to $\mathbb{G}_2$	SVW+II	1705

methods on the *BW13-P310* curve. The resource code is available at `https://github.com/eccdaiy39/hashing`. For the lower prime field operations, we use the implementation provided in the RELIC cryptographic toolkit [1] with Assembly language. The full extension field operations are implemented using C language. Moreover, the technique of lazy reduction [38, 2] is also employed to reduce the number of modular reduction required for multiplication and squaring in $\mathbb{F}_{p^{13}}$. Our algorithms were integrated into the RELIC library. Our benchmarks were performed on a 64-bit Intel Core i9-12900K @3.2GHz processor running Ubuntu 22.04.1 LTS with TurboBoost and hyper-threading features disabled. Clock cycles were obtained averaged over 10,000 executions. Table 3 reports that, for mapping a random point to $\mathbb{G}_2$, Method II leads to 1.88 improvement compared to Method I. For the whole procedure of hashing to $\mathbb{G}_2$, the hash function built on Method II gives a factor 1.71 improvement over one based on Method I.

6. Conclusion and Future Work

Hashing to $\mathbb{G}_2$ is one of building blocks in many pairing-based cryptographic protocols. In this paper, we investigated this issue focusing on pairing-friendly curves with the lack of twists. To this aim, we revisited the previous leading work used in curves admitting a twist. Inspired by it, a general approach was first proposed. More importantly, we found that this approach can be optimized if curves are equipped with non-trivial automorphisms, which is tailored to *BW13-P310* and *BW19-P286*. Finally, we implemented the two approaches over *BW13-P310* on a 64-bit processor. High-performance implementation of full extension field operations was provided such that elliptic curve operations were efficient. Experimental results showed that the hash function built on the optimized approach is up to 71% faster than that on the general version. Recently, a faster SVW map (SwiftEC) was proposed in in [11](ASIACRYPT 2022), which might be used to optimize Algorithm 1. We leave the application of the new method to curves with the lacks of twists as future work.

Acknowledgment

We would like to thank the anonymous referees for their insightful comments. 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. 61972428 and 61972429).

References

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

[2] Diego F. Aranha, Elena Pagnin, and Francisco Rodríguez-Henríquez. Love a pairing. In Patrick Longa and Carla Ràfols, editors, *Progress in Cryptology – LATINCRYPT 2021*, pages 320–340, Cham, 2021. Springer International Publishing.

[3] R. Balasubramanian and Neal Koblitz. The improbability that an elliptic curve has subexponential discrete log problem under the menezes-okamoto-vanstone algorithm. *Journal of Cryptology*, 11(2):141–145, 1998.

[4] Paulo S. L. M. Barreto and Michael Naehrig. Pairing-friendly elliptic curves of prime order. In Bart Preneel and Stafford Tavares, editors, *Selected Areas in Cryptography – SAC 2005*, pages 319–331, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[5] Dan Boneh and Matt Franklin. Identity-based encryption from the weil pairing. In Joe Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, pages 213–229, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[6] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. Computational algebra and number theory (London, 1993).

[7] Colin Boyd, Paul Montague, and Khanh Nguyen. Elliptic curve based password authenticated key exchange protocols. In Vijay Varadharajan and Yi Mu, editors, *Information Security and Privacy –ACISP 2001*, pages 487–501, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[8] Ernie Brickell, Jan Camenisch, and Liqun Chen. Direct anonymous attestation. In *Proceedings of the 11th ACM Conference on Computer and Communications Security*, pages 132–145, New York, NY, USA, 2004. Association for Computing Machinery.

[9] Ernie Brickell, Liqun Chen, and Jiangtao Li. A new direct anonymous attestation scheme from bilinear maps. In Peter Lipp, Ahmad-Reza Sadeghi, and Klaus-Michael Koch, editors, *Trusted Computing - Challenges and Applications – Trust 2008*, pages 166–178, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[10] Alessandro Budroni and Federico Pintore. Efficient hash maps to $\mathbb{G}_2$ on BLS curves. *Applicable Algebra in Engineering, Communication and Computing*, 33(3):261–281, 2022.

[11] Jorge Chavez-Saab, Francisco Rodríguez-Henríquez, and Mehdi Tibouchi. Swiftec: Shallue-van de woestijne indifferentiable function to elliptic curves: Faster indifferentiable hashing to elliptic curves. In *Advances in Cryptology – ASIACRYPT 2022*, page 6392, Berlin, Heidelberg, 2023. Springer-Verlag.

[12] Rémi Clarisse, Sylvain Duquesne, and Olivier Sanders. Curves with fast computations in the first pairing group. In Stephan Krenn, Haya Shulman, and Serge Vaudenay, editors, *Cryptology and Network Security – CANS 2020*, pages 280–298, Cham, 2020. Springer International Publishing.

[13] Youssef El Housni and Aurore Guillevic. Optimized and secure pairing-friendly elliptic curves suitable for one layer proof composition. In Stephan Krenn, Haya Shulman, and Serge Vaudenay, editors, *Cryptology and Network Security – CANS 2020*, pages 259–279, Cham, 2020. Springer International Publishing.

[14] Youssef El Housni and Aurore Guillevic. Families ofăsnark-friendly 2-chains ofăelliptic curves. In Orr Dunkelman and Stefan Dziembowski, editors, *Advances in Cryptology – EUROCRYPT 2022*, pages 367–396, Cham, 2022. Springer International Publishing.

[15] Youssef El Housni, Aurore Guillevic, and Thomas Piellard. Co-factor clearing andăsubgroup membership testing onăpairing-friendly curves. In Lejla Batina and Joan Daemen, editors, *Progress in Cryptology – AFRICACRYPT 2022*, pages 518–536, Cham, 2022. Springer Nature Switzerland.

[16] Nadia El Mrabet and Marc Joye. *Guide to pairing-based cryptography*. Chapman and Hall/CRC, New York, 2016.

[17] Andreas Enge and Jérôme Milan. Implementing cryptographic pairings at standard security levels. In Rajat Subhra Chakraborty, Vashek Matyas, and Patrick Schaumont, editors, *Security, Privacy, and Applied Cryptography Engineering – SPACE 2014*, pages 28–46, Cham, 2014. Springer International Publishing.

[18] Emmanuel Fouotsa and Laurian Azebaze Guimagang. Fast hashing to $\mathbb{G}_2$ on aurifeuillean pairing-friendly elliptic curves. *SN Computer Science*, 1(1):51, 2019.

[19] Pierre-Alain Fouque and Mehdi Tibouchi. Indifferentiable hashing to barreto–naehrig curves. In Alejandro Hevia and Gregory Neven, editors, *Progress in Cryptology – LATINCRYPT 2012*, pages 1–17, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[20] David Freeman, Michael Scott, and Edlyn Teske. A taxonomy of pairing-friendly elliptic curves. *Journal of Cryptology*, 23(2):224–280, 2010.

[21] Gerhard Frey. Applications of arithmetical geometry to cryptographic constructions. In Dieter Jungnickel and Harald Niederreiter, editors, *Proceedings of the 5th International Conference on Finite Fields and Applications(Augsburg, 1999)*, pages 128–161, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[22] Laura Fuentes-Castañeda, Edward Knapp, and Francisco Rodríguez-Henríquez. Faster hashing to $\mathbb{G}_2$. In Ali Miri and Serge Vaudenay, editors, *Selected Areas in Cryptography – SAC 2011*, pages 412–430, Berlin, Heidelberg, 2012. Springer Berlin Heidelberg.

[23] Steven D. Galbraith. Pairings. In Ian F. Blake, Gadiel Seroussi, and Nigel P. Smart, editors, *Advances in Elliptic Curve Cryptography*, volume 317 of *London Mathematical Society Lecture Note Series*, pages 183–214. Cambridge University Press, Cambridge, 2005.

[24] Steven D. Galbraith, Xibin Lin, and Michael Scott. Endomorphisms for faster elliptic curve cryptography on a large class of curves. In Antoine Joux, editor, *Advances in Cryptology - EUROCRYPT 2009*, pages 518–535, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[25] Steven D. Galbraith and Michael Scott. Exponentiation in pairing-friendly groups using homomorphisms. In Steven D. Galbraith and Kenneth G. Paterson, editors, *Pairing-Based Cryptography – Pairing 2008*, pages 211–224, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.

[26] Robert P. Gallant, Robert J. Lambert, and Scott A. Vanstone. Faster point multiplication on elliptic curves with efficient endomorphisms. In Joe Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, pages 190–200, Berlin, Heidelberg, 2001. Springer Berlin Heidelberg.

[27] Daniel M Gordon. Discrete logarithms in GF(p) using the number field sieve. *SIAM Journal on Discrete Mathematics*, 6(1):124–138, 1993.

[28] Jens Groth. On the size of pairing-based non-interactive arguments. In Marc Fischlin and Jean-Sébastien Coron, editors, *Advances in Cryptology – EUROCRYPT 2016*, pages 305–326, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[29] Aurore Guillevic. A short-list of pairing-friendly curves resistant to special tnfs at the 128-bit security level. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *Public-Key Cryptography – PKC 2020*, pages 535–564, Cham, 2020. Springer International Publishing.

[30] F. Hess, N. P. Smart, and F. Vercauteren. The Eta pairing revisited. *IEEE Transactions on Information Theory*, 52(10):4595–4602, 2006.

[31] Thomas Icart. How to hash into elliptic curves. In Shai Halevi, editor, *Advances in Cryptology – CRYPTO 2009*, pages 303–316, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[32] Taechan Kim and Razvan Barbulescu. Extended tower number field sieve: A new complexity for the medium prime case. In Matthew Robshaw and Jonathan Katz, editors, *Advances in Cryptology – CRYPTO 2016*, pages 543–571, Berlin, Heidelberg, 2016. Springer Berlin Heidelberg.

[33] Taechan Kim and Jinhyuck Jeong. Extended tower number field sieve with application to finite fields of arbitrary composite extension degree. In Serge Fehr, editor, *Public-Key Cryptography – PKC 2017*, pages 388–408, Berlin, Heidelberg, 2017. Springer Berlin Heidelberg.

[34] A. K. Lenstra, H. W. Lenstra, and L Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261(4), 1982.

[35] Rudolf Lidl and Harald Niederreiter. *Introduction to finite fields and their applications*. Cambridge University Press, Cambridge, 1994.

[36] Victor S. Miller. The weil pairing, and its efficient calculation. *Journal of Cryptology*, 17(4):235–261, 2004.

[37] René Schoof. Nonsingular plane cubic curves over finite fields. *Journal of combinatorial theory, Series A*, 46(2):183–211, 1987.

[38] Michael Scott. Implementing cryptographic pairings. In *Proceedings of the First International Conference on Pairing-Based Cryptography*, Pairing 2007, page 177196, Berlin, Heidelberg, 2007. Springer-Verlag.

[39] Michael Scott, Naomi Benger, Manuel Charlemagne, Luis J. Dominguez Perez, and Ezekiel J. Kachisa. Fast hashing to $\mathbb{G}_2$ on pairing-friendly curves. In Hovav Shacham and Brent Waters, editors, *Pairing-Based Cryptography – Pairing 2009*, pages 102–113, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[40] Andrew Shallue and Christiaan E. van de Woestijne. Construction of rational points on elliptic curves over finite fields. In Florian Hess, Sebastian Pauli, and Michael Pohst, editors, *Algorithmic Number Theory Symposium– ANTS 2006*, pages 510–524, Berlin, Heidelberg, 2006. Springer Berlin Heidelberg.

[41] Joseph H. Silverman. *The arithmetic of elliptic curves*, volume 106 of *Graduate Texts in Mathematics*. Springer-Verlag, New York, 1992.

[42] Riad S. Wahby and Dan Boneh. Fast and simple constant-time hashing to the bls12-381 elliptic curve. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2019(4):154179, 2019.

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