Optimized Final Exponentiation for Optimal Ate Pairings Using Cyclotomic Cubing

Leila Ben Abdelghani[1] and Walid Haddaji[2,3*]

[1]Laboratory of Analysis, Probability and Fractals, University of Monastir, Tunisia.
[2]Science and technology for defense lab LR19DN01, Center for military research, military academy, Tunis, Tunisia.

*Corresponding author(s). E-mail(s): haddajiwalid95@gmail.com;
Contributing authors: leila.benabdelghani@fsm.rnu.tn;

Abstract

Pairing-based cryptography relies heavily on the efficiency of bilinear pairings, the computation of which is dominated by the final exponentiation step. This paper describes an efficient cubing operation in the cyclotomic subgroup of $\mathbb{F}_{q^6}$ for $q \equiv 1 \mod 6$. As an application, we use existing results for computing Frobenius maps to optimize the cost of the optimal Ate pairing final exponentiation over the SG54 curve. Furthermore, we introduce a novel decomposition for the hard part of the final exponentiation for this curve. Additionally, we apply established methods for cyclotomic cubing to accelerate the final exponentiation for the BLS15 and BLS27 curves. Compared to previous works, our approach achieves efficiency gains of **24%** for SG54 and **22%** for the BLS15 and BLS27 curves.

Keywords: Elliptic curves, pairings, final exponentiation, cyclotomic cubing.

1 Introduction

Pairing-based cryptography has become a cornerstone of modern cryptographic protocols, enabling advanced functionalities such as identity-based encryption [7], short signatures [8], and efficient zero-knowledge proofs [24]. The practical deployment of these protocols depends heavily on the computational efficiency of bilinear pairings. A pairing computation generally consists of two distinct phases: the Miller loop and the final exponentiation. While the Miller loop has been extensively optimized, the final exponentiation remains a significant computational bottleneck, particularly for higher embedding degrees required for adequate security levels.

The efficiency of the final exponentiation is intrinsically related to the arithmetic of the underlying finite fields. Various strategies have been proposed to accelerate this step, such as the use of pairing-friendly fields and cyclotomic squaring. Granger and Scott [14] demonstrated that arithmetic in the cyclotomic subgroup can be optimized using squaring-friendly fields. However, for curves with odd embedding degrees divisible by 3, the cyclotomic squaring is not available. For that, Nanjo et al. [23] proposed cyclotomic cubing as an efficient alternative. The calculation of this operation is improved in [5].

In this paper, we address the challenge of optimizing the final exponentiation for pairing-friendly curves with embedding degrees $k = 15$ and 27, which currently guarantee a 192-bit security level. To securely achieve 256-bit security, Scott and Guillevic considered it prudent to explore curves with embedding degrees greater than 50, leading to the discovery of a new family with embedding degree 54 in [25]. This motivated us to improve the final exponentiation of the optimal Ate pairing for this family, building upon the work presented in [21]. This property facilitates the use of efficient Frobenius operations, thereby reducing the computational overhead of the final exponentiation.

Our contributions are fourfold. First, we derive efficient and explicit formulas for cyclotomic cubing, tailored for even extension degrees of the form 2^i3^j where $i, j > 0$. Second, we apply the method of [5] for

computing Frobenius maps and introduce a novel decomposition of the hard part of the final exponentiation to reduce the computational cost of the optimal Ate pairing over the SG54 curve. We take advantage of compressed squaring to achieve an efficiency gain of 24% over previous results [21]. Finally, we generate new parameter seeds for the BLS15 and BLS27 curves that satisfy $k \mid (p-1)$ and exploit cyclotomic cubing for odd extension degrees divisible by 3 [5, 23]. These seeds reduce final exponentiation costs by approximately 22% compared to the best-known seeds in the literature [10].

The remainder of this paper is organized as follows. Section 2 recalls the definitions of pairing-friendly fields. Section 3 details the optimized arithmetic operations, including cyclotomic cubing formulas and efficient Frobenius computation. Section 4 is devoted to the efficient computation of the optimal Ate final exponentiation over the SG54, BLS15, and BLS27 curves, leveraging the proposed optimizations and providing a thorough complexity analysis along with comparisons against existing methods.

Notations

Throughout this paper, $\mathbb{F}_p$ denotes the finite field of prime order p, and $\mathbb{F}_{p^k}$ denotes its extension field of degree k. We define $\mathbf{M}_p$, $\mathbf{S}_p$, and $\mathbf{I}_p$ as the computational costs of multiplication, squaring, and inversion in the base field $\mathbb{F}_p$, respectively. For operations in extension fields, we use $\mathbf{M}_{p^k}$ and $\mathbf{S}_{p^k}$ to denote multiplication and squaring costs. Operations within cyclotomic subgroups are distinguished by a subscript c; for instance, $\mathbf{S}_{c_{p^k}}$ and $\mathbf{C}_{c_{p^k}}$ represent the costs of cyclotomic squaring and cubing, respectively. The Frobenius endomorphism is denoted by π_p, and its matrix representation with respect to a basis B is written as $\mathrm{Mat}_B(\pi_p)$. The k-th cyclotomic polynomial is denoted by $\Phi_k(x)$, and $G_{\Phi_k(p)}$ refers to the cyclotomic subgroup of $\mathbb{F}_{p^k}^*$. The pairing groups are denoted by $\mathbb{G}_1, \mathbb{G}_2$, and $\mathbb{G}_3$.

2 Pairing, Towering, and Operation-Friendly Fields

In the context of this section, the term "Operation" specifically denotes squaring, cubing, or Frobenius operations. The construction of efficient extension fields is a cornerstone of pairing-based cryptography. Initially, Koblitz and Menezes introduced the concept of *Pairing-Friendly Fields* (PFFs) to standardize and optimize this process [20]. These fields are defined as extensions $\mathbb{F}_{p^k}$ where the characteristic satisfies $p \equiv 1 \pmod{12}$ and the extension degree is of the form $k = 2^i 3^j$ (with $i \geq 1, j \geq 1$). A fundamental result concerning Pairing-Friendly Fields is stated in [14] as follows:

Theorem 1. *Let $\mathbb{F}_{p^k}$ be a PFF, and $\beta \in \mathbb{F}_p$ that is neither a square nor a cube. Then, the polynomial $X^k - \beta$ is irreducible over $\mathbb{F}_p$.*

It is worth noting that the operation of reduction modulo $X^k - \beta$ becomes highly efficient when β is chosen to be small. $\mathbb{F}_{p^k}$ is constructible by repeatedly adjoining square or cube roots of β and its successors until the full extension is obtained. This approach is closely related to the construction of Optimal Extension Fields [2]. The extension degree's form is critical for applications. In particular, if $6 \mid k$, the sextic twist curves enable highly efficient pairing, whereas $4 \mid k$ allows slightly less efficient quartic twists [20].

As demonstrated in [6], the requirement that $p \equiv 1 \pmod{12}$ is somewhat misleading, as Pairing-Friendly Fields do not invariably provide the most efficient extension towers. To accommodate such fields, Benger and Scott [6] proposed the following definition:

Definition 1. *Let q be a power of p and $m \in \mathbb{N}^*$. A field $\mathbb{F}_{q^m}$ is considered towering-friendly if all prime divisors of the extension degree m also divide $q-1$.*

This definition retains the benefits of constructing extensions via successive root extractions but relaxes the congruence constraints on the base field. For example, while the degree 12 extension for BN curves is not a PFF, it can be viewed as a degree 6 extension of $\mathbb{F}_{p^2}$, which qualifies as a TFF since $p^2 - 1$ is divisible by 6.

Granger et al. [14] defined a specific subclass of fields called *Squaring-Friendly Fields* (SFFs). An SFF is defined as a field of the form $\mathbb{F}_{q^6}$ where $q \equiv 1 \pmod{6}$. This specific structure is required for the fast squaring formulae presented in [14]. As we introduce a fast cubing formula for these structures in the subsequent section, we designate them as Cubing-Friendly Fields (CFFs). We further leverage the efficiency of cyclotomic cubing [23] in cases where the extension degree k is odd and divisible by 3. Consequently, we assume the fields discussed herein to be CFFs. Additionally, we impose the condition $k \mid (p-1)$ to facilitate efficient Frobenius operations. For this reason, we refer to the fields presented in this paper as Frobenius-Friendly Fields (FFFs). Hereafter, we restrict our attention to fields that are Pairing, Towering, Squaring, Cubing, and Frobenius-Friendly.

3 Optimized arithmetic in finite field

Let $\mathbb{F}_{p^k}$ be a finite field such that. This section focuses on cyclotomic cubing and Frobenius operations, as these operations will subsequently be used to enhance the efficiency of the final exponentiation in pairing computations.

3.1 Cyclotomic cubing

This section details cyclotomic cubing for two specific scenarios: even extension degrees of the form $k = 2^i 3^j$, with $i, j \in \mathbb{N}^*$, and odd degrees divisible by 3.

3.1.1 Cyclotomic cubing for even extension degrees

The finite field $\mathbb{F}_{p^k}$ is identified with $\mathbb{F}_{q^6}$, where $q = p^{k/6}$ satisfies $q \equiv 1 \pmod 6$. This field admits the tower representation $\mathbb{F}_{q^6} = \mathbb{F}_q[z]/(z^6 - \mathbf{i})$, with $\mathbf{i} \in \mathbb{F}_q$ a sextic non-residue. Following [14], any element $\alpha \in \mathbb{F}_{q^6}$ decomposes as:

$$\alpha = (a_0 + a_1 y) + (b_0 + b_1 y)x + (c_0 + c_1 y)x^2 = a + bx + cx^2, \tag{1}$$

where $a_0, a_1, b_0, b_1, c_0, c_1 \in \mathbb{F}_q$ and $a, b, c \in \mathbb{F}_{q^2}$. Under this construction, $\mathbb{F}_{q^6}$ arises naturally as the compositum of $\mathbb{F}_{q^2}$ and $\mathbb{F}_{q^3}$.

The square of α is given by:

$$\alpha^2 = A + Bx + \mathcal{C}x^2$$

with

$$\begin{cases} A = a^2 + 2\mathbf{i}\, bc \\ B = 2ab + \mathbf{i}c^2 \\ \mathcal{C} = 2ac + b^2 \end{cases} \tag{2}$$

Let $G_{\Phi_6(q)}$ be the cyclotomic subgroup of $\mathbb{F}_{q^6}$ defined by:

$$G_{\Phi_6(q)} = \{g \in \mathbb{F}_{q^6} \mid g^{\Phi_6(q)} = 1\},$$

where Φ_6 is the 6th cyclotomic polynomial. If $\alpha \in G_{\Phi_6(q)}$, then it satisfies the following:

$$\alpha^{q^2 - q + 1} = 1 \tag{3}$$

For $a = a_0 + a_1 y \in \mathbb{F}_{q^2}$, we set

$$\bar{a} = a_0 - a_1 y.$$

According to Granger et al. [14] and using (3), we have the following relations:

$$\begin{cases} \mathbf{i}\, bc = a^2 - \bar{a} \\ ab = \mathbf{i}c^2 - \bar{b} \\ ac = b^2 - \bar{c} \end{cases} \tag{4}$$

The cubing of α is given by:

$$\alpha^3 = \underbrace{(Aa + \mathbf{i}Bc + \mathbf{i}bC)}_{\tilde{A}} + \underbrace{(Ab + aB + \mathbf{i}cC)}_{\tilde{B}}x + \underbrace{(Ac + bB + aC)}_{\tilde{C}}x^2.$$

Using (2) the coefficients A, B, and C can be expressed as follows:

$$\begin{cases} A = 3a^2 - 2\bar{a} \\ B = 3\mathbf{i}c^2 - 2\bar{b} \\ \mathcal{C} = 3b^2 - 2\bar{c}. \end{cases} \tag{5}$$

We substitute (5) into the expressions for $\tilde{A}$, $\tilde{B}$, and $\tilde{\mathcal{C}}$, we get

$$\tilde{A} = 3a^3 + 3\mathbf{i}^2 c^3 + 3\mathbf{i}b^3 - 2a\bar{a} - 2\mathbf{i}\bar{b}c - 2\mathbf{i}b\bar{c}$$

$$\tilde{B} = 3a^2b + 3\mathbf{i}ac^2 + 3\mathbf{i}b^2c - 2\bar{a}b - 2a\bar{b} - 2\mathbf{i}c\bar{c}$$
$$\tilde{\mathcal{C}} = 3a^2c + 3\mathbf{i}bc^2 + 3ab^2 - 2\bar{a}c - 2b\bar{b} - 2a\bar{c}$$

We establish the following proposition, which provides simplified expressions for the coefficients $\tilde{A}$, $\tilde{B}$, and $\tilde{\mathcal{C}}$.

Proposition 2. *The coefficients $\tilde{A}$, $\tilde{B}$, and $\tilde{\mathcal{C}}$ satisfy the following identities:*

1. $\tilde{A} = 9a^3 - 8a\bar{a} + \mathbf{i}(b\bar{c} + \bar{b}c)$.
2. $\tilde{B} = 9a^2b - 3(\bar{a}b - a\bar{b})$.
3. $\tilde{\mathcal{C}} = 9a^2c + 3(a\bar{c} - \bar{a}c)$.

The proof of Proposition 2 relies on the following auxiliary lemma.

Lemma 3. *The coefficients a, b, and c satisfy the following identities:*

1. $\mathbf{i}b^3 = a^3 - a\bar{a} + \mathbf{i}b\bar{c}$.
2. $\mathbf{i}^2c^3 = a^3 - a\bar{a} + \mathbf{i}\bar{b}c$.
3. $\bar{a}b + a\bar{b} = -\mathbf{i}c\bar{c}$.
4. $a\bar{c} + \bar{a}c = -b\bar{b}$.

Proof. By (4), we have $b^3 - b\bar{c} = abc = \mathbf{i}c^3 - \bar{b}c$ and $a^3 - a\bar{a} = \mathbf{i}abc$, which yield

$$\mathbf{i}b^3 - \mathbf{i}b\bar{c} = a^3 - a\bar{a} \quad \text{and} \quad \mathbf{i}^2c^3 - \mathbf{i}\bar{b}c = a^3 - a\bar{a},$$

establishing items (1) and (2). For item (3), applying (4) gives

$$\begin{cases} 3a^2b = 3\mathbf{i}b^2c + 3\bar{a}b, \\ 3\mathbf{i}b^2c = 3\mathbf{i}ac^2 + 3\mathbf{i}c\bar{c}, \\ 3\mathbf{i}ac^2 = 3a^2b + 3a\bar{b}. \end{cases}$$

Summing these three equations and simplifying yields

$$a\bar{b} + \bar{a}b = -\mathbf{i}c\bar{c}.$$

For item (4), a similar application of (4) gives

$$\begin{cases} 3a^2c = 3ab^2 - 3a\bar{c}, \\ 3\mathbf{i}bc^2 = 3a^2c - 3\bar{a}c, \\ 3ab^2 = 3\mathbf{i}bc^2 - 3b\bar{b}. \end{cases}$$

Summing and simplifying gives

$$a\bar{c} + \bar{a}c = -b\bar{b}.$$

□

Proof of Proposition 2. 1. Applying the identities of Lemma 3 to the expression for $\tilde{A}$, we obtain

$$\begin{aligned} \tilde{A} &= 3a^3 + 3\mathbf{i}^2c^3 + 3\mathbf{i}b^3 - 2a\bar{a} - 2\mathbf{i}(b\bar{c} + \bar{b}c) \\ &= 3a^3 + 3(a^3 - a\bar{a} + \mathbf{i}\bar{b}c) + 3(a^3 - a\bar{a} + \mathbf{i}b\bar{c}) - 2a\bar{a} - 2\mathbf{i}(b\bar{c} + \bar{b}c) \\ &= 9a^3 - 8a\bar{a} + \mathbf{i}(b\bar{c} + \bar{b}c). \end{aligned}$$

2. Using Lemma 3 together with the relations in (4), we compute

$$\begin{aligned} \tilde{B} &= 3a^2b + 3\mathbf{i}ac^2 + 3\mathbf{i}b^2c - 2(a\bar{b} + \bar{a}b) - 2\mathbf{i}c\bar{c} \\ &= 3a^2b + 3\mathbf{i}c(ac + b^2) \\ &= 3a^2b + 3\mathbf{i}c(2ac + \bar{c}) \\ &= 3a^2b + 6\mathbf{i}ac^2 + 3\mathbf{i}c\bar{c} \\ &= 3a^2b + 6a(ab + \bar{b}) - 3(a\bar{b} + \bar{a}b) \\ &= 9a^2b + 3a\bar{b} - 3\bar{a}b \\ &= 9a^2b - 3(\bar{a}b - a\bar{b}). \end{aligned}$$

3. Since $\bar{a}c + a\bar{c} = -b\bar{b}$ by Lemma 3(4), we derive

$$\begin{aligned}
\tilde{\mathcal{C}} &= 3a^2c + 3\mathbf{i}bc^2 + 3ab^2 - 2(\bar{a}c + a\bar{c}) - 2b\bar{b} \\
&= 3a^2c + 3\mathbf{i}bc^2 + 3ab^2 \\
&= 3a^2c + 3b(ab + \mathbf{i}c^2) \\
&= 3a^2c + 3b(2ab + \bar{b}) \\
&= 3a^2c + 6ab^2 + 3b\bar{b} \\
&= 3a^2c + 6a(ac + \bar{c}) - 3(a\bar{c} + \bar{a}c) \\
&= 9a^2c + 6a\bar{c} - 3(a\bar{c} + \bar{a}c) \\
&= 9a^2c + 3(a\bar{c} - \bar{a}c).
\end{aligned}$$

□

Lemma 4. *Let $\gamma, \delta \in \mathbb{F}_{q^2}$. $\gamma = \gamma_0 + \gamma_1 y$ and $\delta = \delta_0 + \delta_1 y$, with $\gamma_0, \gamma_1, \delta_0, \delta_1 \in \mathbb{F}_q$, and $y^2 = \mathbf{i}$. Then*

$$\begin{cases}
\gamma\bar{\delta} + \bar{\gamma}\delta = 2(\delta_0\gamma_0 - \mathbf{i}\gamma_1\delta_1) \\
\gamma\bar{\delta} - \bar{\gamma}\delta = 2(\delta_0\gamma_1 - \gamma_0\delta_1)y \\
\gamma\bar{\gamma} = \gamma_0^2 - \mathbf{i}\gamma_1^2
\end{cases}$$

Proof. The first two identities come from adding and subtracting the following two equalities:

$$\begin{cases}
\gamma\bar{\delta} = (\gamma_0 + \gamma_1 y)(\delta_0 - \delta_1 y) = \gamma_0\delta_0 - \mathbf{i}\gamma_1\delta_1 + (\delta_0\gamma_1 - \gamma_0\delta_1)y \\
\bar{\gamma}\delta = (\gamma_0 - \gamma_1 y)(\delta_0 + \delta_1 y) = \gamma_0\delta_0 - \mathbf{i}\gamma_1\delta_1 + (-\delta_0\gamma_1 + \gamma_0\delta_1)y
\end{cases}$$

The third identity is a direct calculation. □

Proposition 5. *The cost of cubing of an element in the sixth cyclotomic subgroup $G_{\Phi_6}(\mathbb{F}_q)$ is $\mathbf{C_{c_{q^6}}} = 2\mathbf{S_q} + 17\mathbf{M_q}$.*

Proof. By Proposition 2 and Lemma 4, the cost of cubing in $G_{\Phi_6}(\mathbb{F}_q)$ amounts to $\mathbf{S}_{q^2} + 3\mathbf{M}_{q^2} + 2\mathbf{S}_q + 6\mathbf{M}_q$. Substituting the identities $\mathbf{S}_{q^2} = 2\mathbf{M}_q$ from [14] and $\mathbf{M}_{q^2} = 3\mathbf{M}_q$ from the Karatsuba method [19], we obtain

$$\mathbf{C}_{\mathbf{c}_{q^6}} = 2\mathbf{M}_q + 3(3\mathbf{M}_q) + 2\mathbf{S}_q + 6\mathbf{M}_q = 2\mathbf{S}_q + 17\mathbf{M}_q.$$

□

Observe that the classical cubing of an element α in $G_{\Phi_6(q)}$ classically costs $\mathbf{C}_{q^6} = \mathbf{M}_{q^6} + \mathbf{S}_{\mathbf{c}_{q^6}}$. Applying the Karatsuba method [19] yields $\mathbf{M}_{q^6} = 18\mathbf{M}_q$, while the cyclotomic squaring $\mathbf{S}_{\mathbf{c}_{q^6}}$ in $\mathbb{F}_{q^6}$ costs $6\mathbf{M}_q$ according to [14]. Consequently,

$$\mathbf{C}_{q^6} = 18\mathbf{M}_q + 6\mathbf{M}_q = 24\mathbf{M}_q.$$

The proposed cyclotomic cubing therefore achieves a saving of $7\mathbf{M}_q - 2\mathbf{S}_q$ over the classical method, corresponding to an efficiency improvement of approximately 22.5%. This comparison is summarized in Table 1.

Method	Cost expression	Gain
Classical cubing in $\mathbb{F}_{q^6}$	$6\mathbf{M_q} + 18\mathbf{M_q}$	$7\mathbf{M_q} - 2\mathbf{S_q}$
Cyclotomic cubing in $G_{\Phi_6}(\mathbb{F}_q)$	$2\mathbf{S_q} + 17\mathbf{M_q}$	

Table 1: Comparison between classical cubing in $\mathbb{F}_{q^6}$ and cyclotomic cubing in $G_{\Phi_6}(\mathbb{F}_q)$.

3.1.2 Cyclotomic cubing for odd extension degree divisible by 3

Efficient cubing is crucial for pairing-based cryptosystems. Nanjo et al. [23] noted that while curves with odd embedding degrees (e.g., $k = 15$ and 27) prevent the use of cyclotomic squaring, "cyclotomic cubing" offers an alternative when k is divisible by 3. In [5], the authors improved a method for cyclotomic cubing. For the sakeness of completin, and for a later use, we recall it here. Let p be a prime and $k \in \mathbb{N}^*$ be a

multiple of 3 such that $k \mid (p-1)$, with $q = p^{k/3}$. We define the cubic extension $\mathbb{F}_{q^3} = \mathbb{F}_q[x]/(x^3 - \xi)$, where $\xi \in \mathbb{F}_q^*$ is a non-cube. Let α be an element in the cyclotomic subgroup $G_{\Phi_3(q)} \subset \mathbb{F}_{q^3}$, defined by:

$$G_{\Phi_3(q)} = \{g \in \mathbb{F}_{q^3} \mid g^{\Phi_3(q)} = 1\},$$

where Φ_3 denotes the third cyclotomic polynomial. We represent α as $a_0 + a_1 x + a_2 x^2$, with $a_0, a_1, a_2 \in \mathbb{F}_q$. According to [5], the cubing of α is given by:

$$\alpha^3 = 1 + 9a_0a_1a_2\xi + 3B_1x + 3B_2x^2,$$

where

$$\begin{cases} B_1 = \frac{1}{2}(a_1 + a_0)(a_1 + a_2)(\xi a_2 + a_0) + \frac{1}{2}(a_1 - a_0)(a_1 - a_2)(\xi a_2 - a_0) - a_0a_1a_2, \\ B_2 = \frac{1}{2}(a_1 + a_0)(a_1 + a_2)(\xi a_2 + a_0) - \frac{1}{2}(a_1 - a_0)(a_1 - a_2)(\xi a_2 - a_0) - \xi a_0a_1a_2. \end{cases}$$

Excluding additions and multiplications by small constants, this cubing costs

$$6\mathbf{M}_{p^{k/3}} + 2\mathbf{m}_{p^{k/3},\xi} + (2 \times k/3)\mathbf{M}_{\mathbf{p^{k/3}}}.$$

3.2 Efficient Frobenius Operation in a Tower of Subfields

The Frobenius operation is vital for final exponentiation in pairings. To optimize this, the authors present, in [5], a method using matrix representations and Kronecker products to simplify complexity analysis and the computation of its cost. They demonstrate this structured approach to counting operations on various pairing-friendly fields. We briefly outline this method and apply it to the target fields $\mathbb{F}_{p^{54}}$, $\mathbb{F}_{p^{15}}$, and $\mathbb{F}_{p^{27}}$.

3.2.1 Description

Let p be a prime, $k \geq 2$ an integer, $\beta \in \mathbb{F}_p$, and $X^k - \beta$ is irreducible over $\mathbb{F}_p$. We define the extension $\mathbb{F}_{p^k} = \mathbb{F}_p(\lambda)$ with the standard basis $B = \{1, \lambda, \ldots, \lambda^{k-1}\}$.

Lemma 6. *Let $\pi_p : x \mapsto x^p$ be the p-Frobenius map. Its matrix representation with respect to basis B is given by:*

$$Mat_B(\pi_p) = diag(1, \theta, \ldots, \theta^{k-1}),$$

where $\theta := \beta^{(p-1)/k} \in \mathbb{F}_p$. Furthermore, for $1 \leq i \leq k$, the matrix for the i-th iterate is

$$Mat_B(\pi_p^i) = diag(1, \theta^i, \ldots, \theta^{i(k-1)}).$$

The following proposition is a straightforward generalization of the result given in [5] .

Proposition 7. *Let $k = k_1 \times \cdots \times k_s$ be a composite integer such that $k_i \mid p-1$ for all $1 \leq i \leq s$. Consider a tower of finite fields defined by*

$$\begin{aligned} \mathbb{F}_{p^{k_1}} &= \mathbb{F}_p(\lambda_1), \quad \lambda_1^{k_1} = \beta_1 \in \mathbb{F}_p, \\ &\vdots \\ \mathbb{F}_{p^{k_s \cdots k_1}} &= \mathbb{F}_{p^{k_{s-1} \cdots k_1}}(\lambda_s), \quad \lambda_s^{k_s} = \beta_s \in \mathbb{F}_{p^{k_{s-1} \cdots k_1}}. \end{aligned}$$

For each $1 \leq j \leq s$, let B_j denote the standard basis of the extension $\mathbb{F}_{p^{k_j}}/\mathbb{F}_{p^{k_{j-1}}}$, and let π_p represent the p-Frobenius map over $\mathbb{F}_{p^k}$. Then, the matrix representation of π_p with respect to the tensor product basis $B = B_s \otimes \cdots \otimes B_1$ is given by:

$$\mathrm{Mat}_B(\pi_p) = M_{\theta_s} \otimes \cdots \otimes M_{\theta_1},$$

where $\theta_l := \beta_l^{(p-1)/k_l}$ and $M_{\theta_l} := \mathrm{diag}(1, \theta_l, \ldots, \theta_l^{k_l - 1})$ for $1 \leq l \leq s$.

Building on this proposition, we present the following theorem, which provides the computational cost of π_p^i in a finite tower of finite fields for $1 \leq i \leq k$.

Theorem 8. *We adopt the same notation as in Proposition 7. Assuming $\theta_2, \ldots, \theta_s \in \mathbb{F}_p$, then for $1 \leq i \leq k$, we have*

$$\mathrm{Mat}_B(\pi_p^i) = M_{\theta_s^i} \otimes \cdots \otimes M_{\theta_1^i},$$

where $M_{\theta_t^i} = \mathrm{diag}(1, \theta_t^i, \ldots, (\theta_t^i)^{k_t - 1})$. Let $S_{i,t} = \mathrm{Card}\{0 \leq j \leq k_t - 1 \mid (\theta_t^i)^j = \pm 2^l; l \in \mathbb{N}\}$ for $1 \leq t \leq s$. Then, the cost of computing π_p^i is given by:

$$(k - \prod_{t=1}^{s} S_{i,t})\mathbf{M}_p.$$

Proof. The proof is a generalization of the proof of Theorem 3.3 in [5]. □

For further details on the proofs of the preceding lemmas, propositions, and theorems, the reader is referred to Section 3 of [5].

3.2.2 Application to $\mathbb{F}_{p^{54}}$, $\mathbb{F}_{p^{15}}$, $\mathbb{F}_{p^{27}}$

Frobenius Computation in $\mathbb{F}_{p^{54}}$

Assuming $p \equiv 1 \pmod{54}$, we use the tower construction presented in [21]:

$$\mathbb{F}_{p^3} = \mathbb{F}_p[x]/(x^3 - 2)$$

$$\mathbb{F}_{p^9} = \mathbb{F}_{p^3}[y]/\left(y^3 - 2^{1/3}\right)$$

$$\mathbb{F}_{p^{54}} = \mathbb{F}_{p^9}[z]/(z^6 - 2^{\frac{1}{6}})$$

Lemma 9. *In the finite field $\mathbb{F}_{p^{54}}$, the computations of all Frobenius maps are free.*

Proof. Let $k_1 = 3, k_2 = 3, k_3 = 6$ with $\theta_1 = 2^{(p-1)/3}$, $\theta_2 = 2^{(p-1)/9}$, and $\theta_3 = 2^{(p-1)/54}$. For any integer $1 \leq i \leq 54$, the representative matrix of the i-th power of the Frobenius map, π_p^i, is given by:

$$\mathrm{Mat}_B(\pi_p^i) = \mathrm{diag}\left(1, 2^{\frac{(p-1)i}{54}}, 2^{\frac{(p-1)i}{27}}, 2^{\frac{(p-1)i}{18}}, 2^{\frac{2(p-1)i}{27}}, 2^{\frac{5(p-1)i}{54}}\right) \otimes \mathrm{diag}\left(1, 2^{\frac{(p-1)i}{9}}, 2^{\frac{2(p-1)i}{9}}\right) \otimes \mathrm{diag}\left(1, 2^{\frac{(p-1)i}{3}}, 2^{\frac{2(p-1)i}{3}}\right).$$

Since $p - 1$ is divisible by 54, every coefficient in the representative matrix of π_p^i can be expressed as 2^l for some $l \in \mathbb{N}$. By applying Theorem 8, we find that $S_{i,1} = 3$, $S_{i,2} = 3$, and $S_{i,3} = 6$. Consequently, the computational cost of π_p^i amounts to $(54 - 3 \times 3 \times 6)\mathbf{M_p} = 0\mathbf{M_p}$. □

We conclude that the Frobenius maps in $\mathbb{F}_{p^{54}}$ can be computed using inexpensive bit shifts.

Frobenius Computation in $\mathbb{F}_{p^{15}}$

We use the tower extension for $\mathbb{F}_{p^{15}}$ as presented in [5]:

$$\mathbb{F}_{p^5} = \mathbb{F}_p[u]/(u^5 - 7);$$
$$\mathbb{F}_{p^{15}} = \mathbb{F}_{p^5}[v](v^3 - 7^{\frac{1}{5}});$$

Let $\theta_1 = 7^{(p-1)/5}$ and $\theta_2 = 7^{(p-1)/15}$. Then $\theta_1 \neq 1$, $\theta_1 \in \mathbb{F}_p$ and, as $15 \mid p - 1$, $\theta_2 \in \mathbb{F}_p$. Moreover, $\theta_2 \neq 1$, $(\theta_2)^3 = 7^{(p-1)/5} = \theta_1 \neq 1$ and $(\theta_2)^{15} = 1$. Thus,

$$\mathrm{Mat}_B(\pi_p) = \mathrm{diag}(1, \theta_2, \theta_2^2) \otimes \mathrm{diag}(1, \theta_1, \theta_1^2, \theta_1^3, \theta_1^4).$$

Lemma 10. *Consider the finite field $\mathbb{F}_{p^{15}}$ such that $15 \mid (p - 1)$.*

1. *For $i \in \{1, \ldots, 10\} \setminus \{5, 10\}$, the computation of the p^i-Frobenius map costs $14\mathbf{M}$.*
2. *The computation of the p^5- and p^{10}-Frobenius maps costs $10\mathbf{M}$.*

The proof of Lemma 10 was given in [5].

Frobenius Computation in $\mathbb{F}_{p^{27}}$

We use the tower extension for $\mathbb{F}_{p^{27}}$ as presented in [10]:

$$\mathbb{F}_{p^3} = \mathbb{F}_p[u]; \quad \text{with } u^3 = 7$$
$$\mathbb{F}_{p^9} = \mathbb{F}_{p^3}[v]; \quad \text{with } v^3 = 7^{1/3}$$
$$\mathbb{F}_{p^{27}} = \mathbb{F}_{p^9}[w]; \quad \text{with } w^3 = 7^{1/9},$$

where 7 is a non-cubic residue in $\mathbb{F}_p$. Let $\theta_1 = 7^{(p-1)/3}$, $\theta_2 = 7^{(p-1)/9}$, and $\theta_3 = 7^{(p-1)/27}$. The following lemma provides the costs of computation of the Frobenius maps.

Lemma 11. *Assume that 27 divides $p - 1$. Then, the Frobenius maps cost as follows:*

1. *The computation of the $p, p^2, p^4, p^5, p^7, p^8$-Frobenius maps costs $26\mathbf{M}_1$.*
2. *The computation of the p^9-Frobenius maps costs $18\mathbf{M}_1$.*

3. The computation of the p^3, p^6-Frobenius maps costs $24\mathbf{M}_1$.

Proof.

1. For $i \in \{1,2,4,5,7,8\}$, the representative matrix is the full tensor product:

$$\mathrm{Mat}_B(\pi_p^i) = \mathrm{diag}\left(1, \theta_3^i, \theta_3^{2i}\right) \otimes \mathrm{diag}\left(1, \theta_2^i, \theta_2^{2i}\right) \otimes \mathrm{diag}\left(1, \theta_1^i, \theta_1^{2i}\right).$$

   For $j \in \{1,2\}$, ij is coprime to $3, 9, 27$. Thus, $\theta_t^{ij} \neq 2^l$ for $t \in \{1,2,3\}$, implying no simplifications. The cost is $(27-1)\mathbf{M}_p = 26\mathbf{M}_p$.
2. For $i = 3$, since $\theta_1^3 = 1$, the matrix representation of π_p^3 is:

$$\mathrm{Mat}_B(\pi_p^3) = \mathrm{diag}\left(1, \theta_2, \theta_2^2\right) \otimes \mathrm{diag}\left(1, \theta_1, \theta_1^2\right) \otimes I_3.$$

   By Theorem 8, $S_{3,1} = 3, S_{3,2} = S_{3,3} = 1$, yielding a cost of $(27-3)\mathbf{M}_p = 24\mathbf{M}_p$. Similarly, the computation of π_p^6 costs $24\mathbf{M}_p$.
3. For $i = 9$, we have $\theta_1^9 = \theta_2^9 = 1$. The matrix representation of π_p^9 is given by:

$$\mathrm{Mat}_B(\pi_p^9) = \mathrm{diag}\left(1, \theta_1, \theta_1^2\right) \otimes I_3 \otimes I_3.$$

   Thus, computing π_p^9 costs $(27-9)\mathbf{M}_p = 18\mathbf{M}_p$.

□

Building on the results of [1, 5, 10, 21], and extending them in this work, we summarize the costs of arithmetic operations in the target fields in Table 2.

Fields	Operations	Costs
$\mathbb{F}_{p^5}$	Multiplication $\mathbf{M_{p^5}}$	$13\mathbf{M_p}$
	Squaring $\mathbf{S_{p^5}}$	$13\mathbf{S_1}$
	Inversion $\mathbf{I_{p^5}}$	$73\mathbf{M_p}$
	Frobenius maps	$4\mathbf{M_p}$
$\mathbb{F}_{p^9}$	Multiplication $\mathbf{M_{p^9}}$	$36\mathbf{M_p}$
	Squaring $\mathbf{S_{p^9}}$	$27\mathbf{M_p}$
	Inversion $\mathbf{I_{p^9}}$	$106\mathbf{M_p}$
	Frobenius maps:	
	p^3, p^9	$6\mathbf{M_p}$
	$p^i; i \in \{1,2,4,5,7,8\}$	$8\mathbf{M_p}$
	Cyclotomic inversion $\mathbf{I_{c_{p^9}}}$	$33\mathbf{M_p}$
	Cyclotomic cubing $\mathbf{C_{c_{p^9}}}$	$48\mathbf{M_p}$
$\mathbb{F}_{p^{15}}$	Multiplication $\mathbf{M_{p^{15}}}$	$78\mathbf{M_p}$
	Squaring $\mathbf{S_{p^{15}}}$	$65\mathbf{M_p}$
	Inversion $\mathbf{I_{p^{15}}}$	$229\mathbf{M_p}$
	Frobenius:	
	p^5, p^{10}	$10\mathbf{M_p}$
	$p^i; i \in \{1,\ldots,10\} \setminus \{5,10\}$	$14\mathbf{M_p}$
	Cyclotomic inversion $\mathbf{I_{c_{p^{15}}}}$	$78\mathbf{M_p}$
	Cyclotomic cubing $\mathbf{C_{c_{p^{15}}}}$	$6\mathbf{M}_{p^5} + 2\mathbf{m}_{p^5,\xi} + (2\times 5)\mathbf{M_{p^5}} \approx 98\mathbf{M_p}$
$\mathbb{F}_{p^{27}}$	Multiplication $\mathbf{M_{27}}$	$216\mathbf{M_p}$
	Squaring $\mathbf{S_{27}}$	$153\mathbf{M_p}$
	Inversion $\mathbf{I_{27}}$	$536\mathbf{M_p}$
	Frobenius:	
	p^9	$18\mathbf{M_p}$
	p^3, p^6	$24\mathbf{M_p}$
	$p^i; i \in \{1,2,4,5,7,8\}$	$26\mathbf{M_p}$
	Cyclotomic inversion $\mathbf{I_{c_{27}}}$	$189\mathbf{M_p}$
	Cyclotomic cubing $\mathbf{C_{c_{27}}}$	$6\mathbf{M}_{p^9} + 2\mathbf{m}_{p^9,\xi} + (2\times 9)\mathbf{M_{p^9}} \approx 252\mathbf{M_p}$
$\mathbb{F}_{p^{54}}$	Multiplication $\mathbf{M_{p^{54}}}$	$648\mathbf{M_p}$
	Cyclotomic squaring $\mathbf{S_{c_{p^{54}}}}$	$6 \times \mathbf{M_{p^9}} \approx 216\mathbf{M_p}$
	Inversion $\mathbf{I_{p^{54}}}$	$\mathbf{I_p} + 1251\mathbf{M_p} + 86\mathbf{S_p} \approx 1344\mathbf{M_p}$
	Frobenius maps	for free
	Cyclotomic inversion $\mathbf{I_{c_{p^{54}}}}$	for free
	Cyclotomic cubing $\mathbf{C_{c_{p^{54}}}}$	$2\mathbf{S_{p^9}} + 17\mathbf{M_{p^9}} \approx 655\mathbf{M_p}$
	Compressed squaring $\mathbf{S_{comp}}$	$144\mathbf{M_p}$

Table 2: The costs of all operations in extension fields $\mathbb{F}_{p^k}$, for $k \in \{5, 9, 15, 27, 54\}$.

4 Application to pairings

This section applies the arithmetic techniques established above to the computation of the final exponentiation of the Ate pairing for BLS54, BLS27, and SG54 elliptic curves.

4.1 Pairings

Consider an elliptic curve E defined over $\mathbb{F}_p$. Let r be a large prime factor of $\#E(\mathbb{F}_p)$ and let k denote the embedding degree, which is the smallest positive integer such that $r \mid (p^k - 1)$. Let $P \in E(\mathbb{F}_p)[r]$ be a point of order r, associated with a rational function $f_{r,P}$ satisfying $\mathrm{Div}(f_{r,P}) = r(P) - r(P_\infty)$. Similarly, let $Q \in E(\mathbb{F}_{p^k})[r]$ be a point of order r, and let $\mu_r \subset \mathbb{F}_{p^k}^*$ represent the group of r-th roots of unity. The optimal Ate pairing, introduced in [27], defines a bilinear map $e_o : \mathbb{G}_2 \times \mathbb{G}_1 \to \mathbb{G}_3$ by:

$$(Q, P) \longmapsto f_{t-1,Q}(P)^{\frac{p^k-1}{r}},$$

where t denotes the trace of the Frobenius endomorphism π_p. Evaluating this map requires $\frac{\log_2 r}{\varphi(k)} + \epsilon(k)$ iterations of Miller's algorithm, where the term $\epsilon(k)$ is bounded by $\log_2 k$. The pairing groups are defined with respect to the Frobenius endomorphism π_p as follows:

- $\mathbb{G}_1 = E(\overline{\mathbb{F}}_p)[r] \cap \ker(\pi_p - 1) = E(\mathbb{F}_p)[r]$,
- $\mathbb{G}_2 = E(\overline{\mathbb{F}}_p)[r] \cap \ker(\pi_p - p)$,
- $\mathbb{G}_3 = \{\mu \in \mathbb{F}_{p^k}^* \mid \mu^r = 1\}$.

The pairing computation consists of two primary stages: the construction of $f_{t-1,Q}(P)$ via Miller's algorithm [22], followed by the final exponentiation to the power of $\frac{p^k-1}{r}$. This final step is typically decomposed into an efficient *easy part* and a computationally intensive *hard part*, for which various optimization strategies have been proposed [10–13, 26]. In particular, Zhang et al. [28] optimized the hard part for $k = 27$ through recursive p-expansion of $\frac{\phi_k(p)}{r}$. Hayashida et al. extended this to arbitrary embedding degrees using homogeneous cyclotomic polynomials from standard ones. The computation of the final exponentiation for the Ate pairing on elliptic curves with embedding degree 54 was also investigated in [21]. In the following sections, we apply the proposed arithmetic to efficiently compute the final exponentiation of the optimal Ate pairing for embedding degrees $54, 15$, and 27, introducing further optimizations to accelerate the process.

4.2 Application to pairing on curve with embedding degree 54

4.2.1 Exploring the SG54 Family

Scott and Guillevic [21] recently introduced a family of elliptic curves with embedding degree $k = 54$, parameterized as follows:

$$\begin{aligned} p(u) &= 1 + 3u + 3u^2 + 3^5u^9 + 3^5u^{10} + 3^6u^{10} + 3^6u^{11} + 3^9u^{18} + 3^{10}u^{19} + 3^{10}u^{20}, \\ r(u) &= 1 + 3^5u^9 + 3^9u^{18}, \\ t(u) &= 1 + 3^5u^{10}, \end{aligned}$$

with seed u selected such that p and r are prime, achieving 256-bit security. The corresponding curve is defined by the following equation:

$$y^2 = x^3 + 12.$$

We represent the fields $\mathbb{F}_{p^{54}}$ as follows:

$$\mathbb{F}_{p^{54}} = \mathbb{F}_{q^6} = \mathbb{F}_q[z]/(z^6 - \mathbf{i}),$$

where $q = p^9$. We henceforth refer to this family as SG54.

The final exponentiation of the Ate pairing on SG54 is decomposed into an easy part and a hard part. The easy part corresponds to the exponent $p^{36} + p^{27} - p^9 - 1$, while the hard part is given by $\frac{\Phi_{54}(p)}{r}$, which expands as follows:

$$\frac{\Phi_{54}(p)}{r} = \sum_{i=0}^{17} m_i p^i,$$

where coefficients satisfy [21]:

$$\begin{array}{lll}
m_{17} = 3u^2+3u+1, & m_4 = 9u^4 m_{17}, & m_{11} = -3u^2 m_4, \\
m_8 = -9u^3 m_{11} + m_{17} + 3u, & m_{16} = um_8 - um_{17}, & m_7 = -2m_{16} - 3um_{17}, \\
m_6 = 3u^2 m_8 + 3u^2 m_{17}, & m_5 = -um_6 - 3u^3 m_{17}, & m_{15} = 3u^2 m_{17} - m_6, \\
m_{13} = 3um_5 + m_4, & m_{12} = um_4 - um_{13}, & m_3 = m_{12} - 3um_4, \\
m_{14} = 3u^3 m_{17} - 2um_{15}, & m_2 = -3um_3 - 3u^2 m_4, & m_{10} = um_2 - um_{11}, \\
m_9 = -3um_{10} - 3u^2 m_{11} - 2, & m_1 = -2um_2 - um_{11}, & m_0 = 3u^2 m_{11} - m_9 - 1.
\end{array} \tag{6}$$

We refine this decomposition in the following paragraph.

4.2.2 New hard part Decomposition

In order to optimize the final exponentiation of the optimal Ate pairing on the $SG54$ curve, we exploit redundancy in the decomposition (6) and derive the following alternative form:

$$\begin{array}{lll}
\hat{u} = 3u, & m_{17} = \hat{u}u + \hat{u} + 1, & t = um_{17}, \\
s = \hat{u}t, & z = us, & m_4 = \hat{u}z, \\
w = um_4, & y = -3w, & m_{11} = uy, \\
r = -um_{11}, & k = \hat{u}r, & m_8 = \hat{u}k + m_{17} + \hat{u}, \\
m = um_8, & m_{16} = m - t, & m_7 = -2m - t, \\
m_6 = \hat{u}m + s, & m_5 = -um_6 - z, & m_{15} = s - m_6, \\
\tilde{m}_5 = \hat{u}m_5, & m_{13} = \tilde{m}_5 + m_4, & m_{12} = -u\tilde{m}_5, \\
m_3 = m_{12} + y, & m_{14} = -2m_5 - 3z, & m_2 = -\hat{u}m_3 - \hat{u}w, \\
\tilde{m}_2 = um_2, & m_{10} = \tilde{m}_2 - r, & \tilde{m}_{10} = \hat{u}m_{10}, \\
m_9 = -\tilde{m}_{10} + k - 2, & m_1 = -2\tilde{m}_2 + r, & m_0 = -m_9 - k - 1.
\end{array}$$

Using this decomposition, we introduce Algorithm 1, which enables an efficient computation of the hard part of the final exponentiation in the optimal Ate pairing on the SG54 curve.

Algorithm 1 Hard Part Computation of the Optimal Ate Pairing on SG54

Input: $f \in G_{\Phi_{54}(p)}$ (The output of the Miller loop).

Output: $f^{\frac{\Phi_{54}(p)}{r}}$.

$Q \leftarrow 1$
$\hat{U} \leftarrow f^u$
$\hat{U} \leftarrow \hat{U}^3$
$V17 \leftarrow \hat{U}^u$
$V17 \leftarrow V17 \cdot \hat{U}$
$V17 \leftarrow V17 \cdot f$
$Q \leftarrow Q \cdot V17^{p^{17}}$
$T \leftarrow V17^u$
$S \leftarrow T^u$
$S \leftarrow S^3$
$Z \leftarrow S^u$
$V4 \leftarrow Z^u$
$V4 \leftarrow V4^3$
$Q \leftarrow Q \cdot V4^{p^4}$
$W \leftarrow V4^u$
$Y \leftarrow 1/W$
$Y \leftarrow Y^3$
$V11 \leftarrow Y^u$
$Q \leftarrow Q \cdot V11^{p^{11}}$
$R \leftarrow V11^u$
$RINV \leftarrow 1/R$
$K \leftarrow RINV^u$
$K \leftarrow K^3$
$V8 \leftarrow K^u$
$V8 \leftarrow V8^3$
$V8 \leftarrow V8 \cdot V17$
$V8 \leftarrow V8 \cdot \hat{U}$
$Q \leftarrow Q \cdot V8^{p^8}$
$M \leftarrow V8^u$
$V16 \leftarrow 1/T$
$V16 \leftarrow V16 \cdot M$
$Q \leftarrow Q \cdot V16^{p^{16}}$
$V7 \leftarrow M^2$
$V7 \leftarrow V7 \cdot T$
$V7 \leftarrow 1/V7$
$Q \leftarrow Q \cdot V7^{p^7}$
$V6 \leftarrow M^u$
$V6 \leftarrow V6^3$
$V6 \leftarrow V6 \cdot S$
$Q \leftarrow Q \cdot V6^{p^6}$
$V5 \leftarrow V6^u$
$V5 \leftarrow V5 \cdot Z$
$V5 \leftarrow 1/V5$
$Q \leftarrow Q \cdot V5^{p^5}$
$V15 \leftarrow 1/V6$
$V15 \leftarrow V15 \cdot S$
$Q \leftarrow Q \cdot V15^{p^{15}}$
$\tilde{V}5 \leftarrow V5^u$
$\tilde{V}5 \leftarrow \tilde{V}5^3$
$V13 \leftarrow \tilde{V}5 \cdot V4$
$Q \leftarrow Q \cdot V13^{p^{13}}$
$V12 \leftarrow \tilde{V}5^u$
$V12 \leftarrow 1/V12$
$Q \leftarrow Q \cdot V12^{p^{12}}$
$V3 \leftarrow V12 \cdot Y$
$Q \leftarrow Q \cdot V3^{p^3}$
$V14 \leftarrow V5^2$
$V14 \leftarrow V14 \cdot Z^3$
$V14 \leftarrow 1/V14$
$Q \leftarrow Q \cdot V14^{p^{14}}$
$V2 \leftarrow V3 \cdot W$
$V2 \leftarrow V2^u$
$V2 \leftarrow V2^3$
$V2 \leftarrow 1/V2$
$Q \leftarrow Q \cdot V2^{p^2}$
$\tilde{V}2 \leftarrow V2^u$
$V10 \leftarrow \tilde{V}2 \cdot RINV$
$Q \leftarrow Q \cdot V10^{p^{10}}$
$\tilde{V}10 \leftarrow V10^u$
$\tilde{V}10 \leftarrow \tilde{V}10^3$
$V9 \leftarrow 1/\tilde{V}10$
$V9 \leftarrow V9 \cdot K$
$V9 \leftarrow V9 \cdot f^{-2}$
$Q \leftarrow Q \cdot V9^{p^9}$
$V1 \leftarrow \tilde{V}2^2$
$V1 \leftarrow 1/V1$
$V1 \leftarrow V1 \cdot RINV$
$Q \leftarrow Q \cdot V1^p$
$V0 \leftarrow V9 \cdot K$
$V0 \leftarrow V0 \cdot f$
$V0 \leftarrow 1/V0$
$Q \leftarrow Q \cdot V0$
return(Q)

4.2.3 Complexity and comparison

Using the parameter seed

$$u = 2^{27} + 2^{26} + 2^{22} + 2^{14} + 2^6 + 2,$$

one can verify with SageMath that $54 \mid (p-1)$. Consequently, cyclotomic inversion in $\mathbb{F}_{p^{54}}$ is negligible, and all Frobenius maps can be computed efficiently via inexpensive bit shifts. The complexity of the hard part (Algorithm 1) is:

$$19\mathbf{E_u} + 37\mathbf{M_{p^{54}}} + 4\mathbf{S_{c_{p^{54}}}} + 11\mathbf{C_{c_{p^{54}}}}.$$

Based on the findings of [18] and [21], compressed squaring is more efficient than the cyclotomic squaring method described in [14] for exponentiation by the seed u. Thus, using the results of [1, 18, 21] and assuming that $\mathbf{S_p} = 0.8\mathbf{M_p}$, the cost of exponentiation by u is $\mathbf{E_u} \approx 8773\mathbf{M_p}$. Referring to Table 2, the

hard part cost is calculated as follows:

$$19(8773\mathbf{M_p}) + 37(648\mathbf{M_p}) + 4(216\mathbf{S_p}) + 11(655\mathbf{M_p}) \approx 198732\mathbf{M_p}.$$

According to [21], the easy part costs:

$$\mathbf{I_{p^{54}}} + 3\mathbf{M_{p^{54}}} + 3\mathbf{p^9} + \mathbf{p^{18}} = \mathbf{I_{p^{54}}} + 3\mathbf{M_{p^{54}}} = (1251\mathbf{M_p} + 62\mathbf{S_p} + \mathbf{I_p}) + 3(648\mathbf{M_p}) \approx 3269\mathbf{M_p}.$$

Consequently, the total cost of the final exponentiation is $202001\mathbf{M_p}$. Compared to the original cost of $266481\mathbf{M_p}$ reported in [21], our approach achieves a saving of:

$$266481\mathbf{M_p} - 202001\mathbf{M_p} = 64480\mathbf{M_p}.$$

We summarize the comparison in Table 3.

Approach	**Cost ($\mathbf{M}_p$)**	**Savings**
Original [21]	266481	-
Optimized (This work)	202001	**24%**

Table 3: Comparison of final exponentiation costs.

4.3 Application to pairing on curves with odd embedding degree divisible by 3

4.3.1 Optimal Ate Pairing on BLS15, BLS27

A BLS curve [4] is a pairing-friendly elliptic curve over $\mathbb{F}_p$ given by $y^2 = x^3 + b$ with $b \in \mathbb{F}_p^*$, designed for efficient pairing computation.

BLS15

The BLS15 family [9] comprises embedding degree $k = 15$ curves parametrized by:

$$\begin{aligned} p &= \frac{u^{12} - 2u^{11} + u^{10} + u^7 - 2u^6 + u^5 + u^2 + u + 1}{3}, \\ r &= u^8 - u^7 + u^5 - u^4 + u^3 - u + 1, \\ t &= u + 1, \end{aligned}$$

where the seed u is selected such that p and r are both prime, achieving secure and efficient pairings at 128- and 192-bit security levels. The optimal Ate pairing on the curve $BLS15$ is:

$$\begin{aligned} e_o : \mathbb{G}_2 \times \mathbb{G}_1 &\longrightarrow \mathbb{G}_3 \\ (Q, P) &\longmapsto f_{u,Q}(P)^{\frac{p^{15}-1}{r}}. \end{aligned}$$

The easy final exponentiation raises Miller's result to $(p^5 - 1)(p^2 + p + 1)$, while the hard part requires exponentiation by $\frac{\Phi_{15}(p)}{r}$. For efficiency, Hayashida et al. [17] use the multiple $3 \cdot \frac{\Phi_{15}(p)}{r}$ expressed as:

$$3 \cdot \frac{\Phi_{15}(p)}{r} = (u-1)^2(u^2 + u + 1) + \sum_{i=0}^{7} \lambda_i p^i + 3,$$

with coefficients $\lambda_7 = 1$, $\lambda_6 = u\lambda_7 - 1$, $\lambda_5 = u\lambda_6$, $\lambda_4 = u\lambda_5 + 1$, $\lambda_3 = u\lambda_4 - 1$, $\lambda_2 = u\lambda_3 + 1$, $\lambda_1 = u\lambda_2$, $\lambda_0 = u\lambda_1 - 1$.

BLS27

The BLS27 family [28] has embedding degree $k = 27$ with parameters:

$$r = \frac{u^{18} + u^9 + 1}{3},$$

$$p = (u-1)^2 r + u,$$
$$t = u + 1,$$

where u ensures p, r prime for target security. The optimal Ate pairing becomes:

$$e_o : \mathbb{G}_2 \times \mathbb{G}_1 \longrightarrow \mathbb{G}_3$$
$$(Q, P) \longmapsto f_{u,Q}(P)^{\frac{p^{27}-1}{r}}.$$

The easy part elevates Miller's result to $p^9 - 1$; the hard part applies $\frac{\Phi_{27}(p)}{r}$:

$$\frac{\Phi_{27}(p)}{r} = (u-1)^2(u^2 + pu + p^2)(u^6 + p^3u^3 + p^6)(u^9 + p^9 + 1) + 3.$$

4.3.2 Cyclotomic cubing friendly seeds

Prior work [15] optimized final exponentiation via cyclotomic cubing in the **TCAB** method. We extend this by constructing seeds with most significant bits formed by products of powers of 2 and 3, thereby maximizing the use of cyclotomic cubing. To facilitate cyclotomic cubing in the final exponentiation while preserving binary sparsity for the Miller loop, we propose to generate new seeds of the form:

$$u = c_1 \times 2^{i_1} + \cdots + c_{m-1} \times 2^{i_{m-1}} + c_m \times 2^{i_m} + 2^s \times 3^c, \tag{7}$$

where $m, s, c, i_m, i_{i-1}, \cdots, i_1$ are positive integers such that $s > i_r, \forall r \in \{1, 2, \cdots, m\}$, $c \geq 1$, and $c_i \in \{-1, 0, 1\}$.

This representation enables us to maintain binary sparsity while minimizing the parameter m, control the number of squarings via the parameter s, and merge specific squarings and multiplications with cyclotomic cubings to reduce the overall cost of exponentiation by u within the cyclotomic subgroup. Consequently, to generate a new efficient seed, we fix the parameters s and c according to the security criteria specified in [3]. We then begin the generation process with a minimum value for the parameter m and incrementally increase it until a suitable seed is obtained.

As an application, we target the BLS15 and BLS27 curves, which have recently been studied as pairing-friendly curves. To generate the new seeds for the $BLS15$ and $BLS27$ curves, we take into account the following constraints:

1. The seeds must be chosen such that both p and r are prime.
2. Follow the security recommendations outlined in [3].
3. The generated seeds are expected to offer greater efficiency than those presented in [10].
4. The seed must satisfy the property $k \mid (p-1)$.

We generated several seeds satisfying both the representation (7) and the primality conditions for p and r. However, only two of them fulfill the condition $k \mid (p-1)$. The efficiency our seeds is compared with previous work. The resulting seeds are presented in Table 4.

Seed	Security Level	k	Size(r)	Size(p)	Size(p^k)	b	DL algorithm
$-2^{41} + 2^{55} \times 3$	192	15	453	678	10162	6	exTNFS
$-2^2 - 2^{14} + 2^{22} \times 3^{11}$	256	27	709	788	21252	2	exTNFS

Table 4: New valid seeds for BLS15 and BLS27 verifying $k \mid (p-1)$.

Remark 1. *The seed u is also used in the computation of the Miller loop. Due to the presence of 3^c in the structure of the proposed seeds, their binary representations may contain runs of consecutive ones. Since point negation on an elliptic curve is free, it is preferable to use the Non-Adjacent Form (NAF) [16] of the seed to carry out the Miller loop computation efficiently and reduce the number of point additions.*

4.3.3 Complexity Analysis

This section evaluates the computational cost of a single cyclotomic exponentiation and the overall final exponentiation for pairings on BLS15 and BLS27 curves using the generated seeds.

BLS15 Curve

The cost of the cyclotomic exponentiation by the seed

$$u_{19215} = -2^{41} + 2^{55} \times 3$$

is given by:

$$\mathbf{E_{u_{19215}}} = \mathbf{M_{p^{15}}} + 55\mathbf{S_{p^{15}}} + \mathbf{C_{c_{p^{15}}}} + \mathbf{I_{c_{p^{15}}}}.$$

The cost of exponentiation by $u_{19215} - 1$ is:

$$\mathbf{E_{u_{19215}-1}} = 2\mathbf{M_{p^{15}}} + 55\mathbf{S_{p^{15}}} + \mathbf{C_{c_{p^{15}}}} + \mathbf{I_{c_{p^{15}}}}.$$

Using Table 2, we get $\mathbf{E_{u_{19215}}} = 3829\mathbf{M_p}$ and $\mathbf{E_{u_{19215}-1}} = 3907\mathbf{M_p}$. Utilizing cyclotomic cubing, the complexity of the final exponentiation for BLS15 is presented in [**?**] as follows:

$$\mathbf{I_{p^{15}}} + 18 \times \mathbf{M_{p^{15}}} + \mathbf{C_{c_{p^{15}}}} + 10 \times \mathbf{F_{15}} + \mathbf{I_{c_{p^{15}}}} + 2 \times \mathbf{E_{u_{19215}-1}} + 9 \times \mathbf{E_{u_{19215}}}$$

where $\mathbf{F_{15}}$ is the cost of the Frobenius maps that is given by:

$$\mathbf{F_{15}} = 2\mathbf{p^5} + 3\mathbf{p} + (\mathbf{p^2} + \mathbf{p^3} + \mathbf{p^4} + \mathbf{p^6} + \mathbf{p^7}) = (2 \times 10 + 8 \times 14)\mathbf{M_p} = 132\mathbf{M_p}.$$

Thus, the cost of the final exponentiation is:

$$(229 + 18 \times 78 + 98 + 132 + 78 + 2 \times 3907 + 9 \times 3829)\mathbf{M_p} = 44216\mathbf{M_p}$$

BLS27 Curves

The cyclotomic exponentiation costs for the seed $u_{25627} = -2^2 - 2^{14} + 2^{22} \times 3^{11}$ and its decrement are given by:

$$\mathbf{E_{u_{25627}}} = 2\mathbf{M_{p^{27}}} + 22\mathbf{S_{p^{27}}} + 11\mathbf{C_{c_{p^{27}}}} + \mathbf{I_{c_{p^{27}}}},$$
$$\mathbf{E_{u_{25627}-1}} = 3\mathbf{M_{p^{27}}} + 22\mathbf{S_{p^{27}}} + 11\mathbf{C_{c_{p^{27}}}} + \mathbf{I_{c_{p^{27}}}}.$$

Using Table 2, these evaluate to $\mathbf{E_{u_{25627}}} = 6759\mathbf{M_p}$ and $\mathbf{E_{u_{25627}-1}} = 6975\mathbf{M_p}$. Following the approach in [**?**], the total complexity for the BLS27 final exponentiation using cyclotomic cubing is:

$$\mathbf{I_{27}} + 8\mathbf{M_{27}} + \mathbf{C_{c_{27}}} + 6\mathbf{F_{27}} + 2\mathbf{E_{u_{25627}-1}} + 17\mathbf{E_{u_{25627}}},$$

where $\mathbf{F_{27}}$ denotes the total cost of the Frobenius maps and it is calculated as:

$$\mathbf{F_{27}} = \mathbf{p} + \mathbf{p^2} + \mathbf{p^3} + \mathbf{p^6} + 2\mathbf{p^9} = (2 \times 26 + 2 \times 24 + 2 \times 18)\mathbf{M_p} = 136\mathbf{M_p}.$$

Summing these components, the total cost amounts to:

$$(536 + 8 \times 216 + 252 + 136 + 2 \times 6975 + 17 \times 6759)\mathbf{M_p} = 131505\mathbf{M_p}.$$

4.3.4 Comparison

This section presents a comparative analysis of the efficiency of our newly generated seeds against those proposed in [10].

For a 192-bit security level, the authors in [10] proposed the seed

$$u_1 = 1 + 2^5 + 2^9 + 2^{40} + 2^{72}$$

for BLS15. This seed satisfies the property $k \mid (p - 1)$, similar to our new seed proposed for the same security level. In [10], the square-and-multiply algorithm is employed for cyclotomic exponentiation. The cost of exponentiation by u_1 is

$$\mathbf{E_{u_1}} = 72\mathbf{S_{p^{15}}} + 4\mathbf{M_{p^{15}}} = 4992\mathbf{M_p},$$

while the cost for $u_1 - 1$ is

$$\mathbf{E_{u_1-1}} = 72\mathbf{S_{p^{15}}} + 3\mathbf{M_{p^{15}}} = 4914\mathbf{M_p}.$$

For the seed u_1, the computational cost of the final exponentiation of the Ate pairing on BLS15 is:

$$\mathbf{I_{p^{15}}} + 19\mathbf{M_{p^{15}}} + \mathbf{S_{p^{15}}} + \mathbf{F_{15}} + \mathbf{I_{c_{p^{15}}}} + 2\mathbf{E_{u_1-1}} + 9\mathbf{E_{u_1}},$$

where $\mathbf{F_{15}}$ is derived similarly to the previous section, given that $15 \mid (p-1)$. Thus, the cost of the final exponentiation is:

$$(229 + 19 \times 78 + 65 + 132 + 78 + 2 \times 4914 + 9 \times 4992)\mathbf{M_p} = 56742\mathbf{M_p}$$

Our new seed for the 192-bit security level achieves a 22% efficiency improvement over the proposal in [10].

For a 256-bit security level, the authors in [10] proposed the seed

$$u_2 = 2^{51} + 2^{42} + 2^{28} + 2^9 + 1$$

for BLS27. Unlike our proposal, this seed does not satisfy the property $k \mid (p-1)$. The cost of cyclotomic exponentiation by u_2 is

$$\mathbf{E_{u_2}} = 51\mathbf{S_{p^{27}}} + 4\mathbf{M_{p^{27}}} = 8667\mathbf{M_p},$$

while the cost for $u_2 - 1$ is

$$\mathbf{E_{u_2-1}} = 51\mathbf{S_{p^{27}}} + 3\mathbf{M_{p^{27}}} = 8451\mathbf{M_p}.$$

The computational cost of the final exponentiation of the Ate pairing on BLS27 using u_2 is:

$$\mathbf{I_{p^{27}}} + 9\mathbf{M_{p^{27}}} + \mathbf{S_{p^{27}}} + \mathbf{F_{27}} + 2\mathbf{E_{u_2-1}} + 17\mathbf{E_{u_2}}.$$

The Frobenius cost $\mathbf{F_{27}}$ is derived differently from the previous section since $27 \nmid (p-1)$, and is given by:

$$\mathbf{F_{27}} = \mathbf{p} + \mathbf{p^2} + \mathbf{p^3} + \mathbf{p^6} + 2\mathbf{p^9} = (2 \times 686 + 2 \times 114 + 2 \times 18)\mathbf{M_p} = 1636\mathbf{M_p}. \quad (8)$$

Thus, the total cost of the final exponentiation is:

$$(536 + 9 \times 216 + 153 + 1636 + 2 \times 8451 + 17 \times 8667)\mathbf{M_p} = 168510\mathbf{M_p}.$$

Consequently, our new seed for the 256-bit security level achieves a 22% efficiency improvement over the proposal in [10].

Table 5 outlines the structural properties of the generated seeds. A notable distinction is that the proposed seed for BLS27 satisfies the condition $k \mid (p-1)$, which enables efficient computation of Frobenius. In contrast, the seed of [10] does not satisfy this property, i.e., $k \nmid (p-1)$, resulting in higher Frobenius map costs. Accordingly, we recompute and correct these costs in (8).

Proposal	**Curve**	**Security Level**	**Property** $k \mid (p-1)$
Our Seed	BLS15	192-bit	Yes
Fouotsa et al. [10]	BLS15	192-bit	Yes
Our Seed	BLS27	256-bit	Yes
Fouotsa et al. [10]	BLS27	256-bit	No

Table 5: Comparison of the property $k \mid (p-1)$ for generated seeds.

The computational costs of the final exponentiation are compared in Table 6. The results demonstrate that our newly generated seeds consistently outperform those proposed in [10]. Specifically, our approach reduces the cost of the final exponentiation by approximately 22% for both the BLS15 and BLS27 curves.

Curve	Proposal	Cost	Gain	Gain (%)
BLS15	Our Seed	44216	12,526	22%
	Fouotsa et al. [10]	56742		
BLS27	Our Seed	131505	37,005	22%
	Fouotsa et al. [10]	168510		

Table 6: Cost comparison of final exponentiation (in $\mathbf{M_p}$).

5 Conclusion

In this work, we have presented optimizations of the final exponentiation step of the optimal Ate pairing for pairing-friendly curves with embedding degrees $k = 15$, 27, and 54. We derived explicit and efficient formulas for cyclotomic cubing for even extension degrees of the form 2^i3^j, with $i, j > 0$. For the SG54 curve, we introduced a novel decomposition of the hard part of the final exponentiation, achieving a 24% efficiency improvement over [21]. For the BLS15 and BLS27 curves, we generated new parameter seeds satisfying $k \mid (p-1)$, reducing the final exponentiation cost by approximately 22% compared to [10], as summarized in Table 6. These results demonstrate that a careful combination of seed selection, cyclotomic arithmetic, and field structure leads to substantial efficiency gains in pairing-based cryptography. Future work may explore the application of these techniques to other families of pairing-friendly curves with higher embedding degrees.

References

[1] Aranha DF, Fotiadis G, Guillevic A (2024) A short-list of pairing-friendly curves resistant to the special tnfs algorithm at the 192-bit security level. Cryptology ePrint Archive Report 2024/294

[2] Bailey DV, Paar C (1998) Optimal extension fields for fast arithmetic in public-key algorithms. In: Annual International Cryptology Conference, Springer, pp 472–485

[3] Barbulescu R, Duquesne S (2019) Updating key size estimations for pairings. Journal of cryptology 32:1298–1336

[4] Barreto PS, Lynn B, Scott M (2003) Constructing elliptic curves with prescribed embedding degrees. In: Security in Communication Networks: Third International Conference, SCN 2002 Amalfi, Italy, September 11–13, 2002 Revised Papers 3, Springer, pp 257–267

[5] Ben Abdelghani L, El Mrabet N, Ghammam L, et al (2025) Optimized frobenius and cyclotomic cubing for enhanced pairing computation. Cryptology ePrint Archive

[6] Benger N, Scott M (2010) Constructing tower extensions of finite fields for implementation of pairing-based cryptography. In: International workshop on the arithmetic of finite fields, Springer, pp 180–195

[7] Boneh D, Franklin M (2001) Identity-based encryption from the weil pairing. In: Annual international cryptology conference, Springer, pp 213–229

[8] Boneh D, Lynn B, Shacham H (2001) Short signatures from the weil pairing. In: International conference on the theory and application of cryptology and information security, Springer, pp 514–532

[9] Duan P, Cui S, Chan CW (2005) Special polynomial families for generating more suitable elliptic curves for pairing-based cryptosystems. Cryptology ePrint Archive

[10] Fouotsa E, El Mrabet N, Pecha A (2020) Optimal ate pairing on elliptic curves with embedding degree $9, 15$ and 27. Journal of groups, complexity, cryptology 12

[11] Fuentes-Castaneda L, Knapp E, Rodríguez-Henríquez F (2011) Faster hashing to. In: International workshop on selected areas in cryptography, Springer, pp 412–430

[12] Ghammam L, Fouotsa E (2016) Adequate elliptic curves for computing the product of n pairings. In: International workshop on the arithmetic of finite fields, Springer, pp 36–53

[13] Ghammam L, Fouotsa E (2019) Improving the computation of the optimal ate pairing for a high security level. Journal of Applied Mathematics and Computing 59(1):21–36

[14] Granger R, Scott M (2010) Faster squaring in the cyclotomic subgroup of sixth degree extensions. In: International Workshop on Public Key Cryptography, Springer, pp 209–223

[15] Haddaji W, Ghammam L, El Mrabet N, et al (2025) Efficient pairings final exponentiation using cyclotomic cubing for odd embedding degrees curves. Cryptology ePrint Archive

[16] Hankerson D, Menezes AJ, Vanstone SA (2004) Guide to Elliptic Curve Cryptography. Springer

[17] Hayashida D, Hayasaka K, Teruya T (2020) Efficient final exponentiation via cyclotomic structure for pairings over families of elliptic curves. Cryptology ePrint Archive

[18] Karabina K (2013) Squaring in cyclotomic subgroups. Mathematics of Computation 82(281):555–579

[19] Karatsuba A (1963) Multiplication of multidigit numbers on automata. In: Soviet physics doklady, pp 595–596

[20] Koblitz N, Menezes A (2005) Pairing-based cryptography at high security levels. In: IMA International Conference on Cryptography and Coding, Springer, pp 13–36

[21] Mbiang NB, Aranha DDF, Fouotsa E (2020) Computing the optimal ate pairing over elliptic curves with embedding degrees 54 and 48 at the 256-bit security level. International Journal of Applied Cryptography 4(1):45–59

[22] Miller VS (2004) The weil pairing, and its efficient calculation. Journal of cryptology 17(4):235–261

[23] Nanjo Y, Shirase M, Kusaka T, et al (2020) An explicit formula of cyclotomic cubing available for pairings on elliptic curves with embedding degrees of multiple of three. In: 2020 35th International Technical Conference on Circuits/Systems, Computers and Communications (ITC-CSCC), IEEE, pp 288–292

[24] Peng Z, Zhao C, Wang T, et al (2025) A survey of zero-knowledge proof based verifiable machine learning. arXiv preprint arXiv:250218535

[25] Scott M, Guillevic A (2018) A new family of pairing-friendly elliptic curves. In: Arithmetic of Finite Fields: 7th International Workshop, WAIFI 2018, Bergen, Norway, June 14-16, 2018, Revised Selected Papers 7, Springer, pp 43–57

[26] Scott M, Benger N, Charlemagne M, et al (2009) On the final exponentiation for calculating pairings on ordinary elliptic curves. In: Pairing-Based Cryptography–Pairing 2009: Third International Conference Palo Alto, CA, USA, August 12-14, 2009 Proceedings 3, Springer, pp 78–88

[27] Vercauteren F (2009) Optimal pairings. IEEE transactions on information theory 56(1):455–461

[28] Zhang X, Lin D (2012) Analysis of optimum pairing products at high security levels. In: Progress in Cryptology-INDOCRYPT 2012: 13th International Conference on Cryptology in India, Kolkata, India, December 9-12, 2012. Proceedings 13, Springer, pp 412–430