An Alternate Decomposition of an Integer for Faster Point Multiplication on Certain Elliptic Curves

Young-Ho Park[1,⋆], Sangtae Jeong[2], Chang Han Kim[3], and Jongin Lim[1]

[1] CIST, Korea Univ., Seoul, Korea
{youngho,jilim}@cist.korea.ac.kr
[2] Dept. of Math., Seoul National Univ., Seoul, Korea
stj@math.snu.ac.kr
[3] Dept. of CAMIS, Semyung Univ., Jechon, Korea
chkim@venus.semyung.ac.kr

Abstract. In this paper the Gallant-Lambert-Vanstone method is re-examined for speeding up scalar multiplication. Using the theory of μ-Euclidian algorithm, we provide a rigorous method to reduce the theoretical bound for the decomposition of an integer k in the endomorphism ring of an elliptic curve. We then compare the two different methods for decomposition through computational implementations.

1 Introduction

Public key cryptosystems based on the discrete log problem on elliptic curves over finite fields(ECC) have gained much attention as a popular and practical scheme for computational advantages as well as for communicational advantages. As the complexity of protocols based on ECC relies mostly on the complexity of scalar multiplication, the dominant cost operation is computing kP for a point P on an elliptic curve.

Various methods for faster scalar multiplication have been devised by selecting relevant objects involving base fields and elliptic curves [1,3]. For example, by considering elliptic curves defined over the binary field, say Koblitz curves, Koblitz [5], Meier and Staffelbach [7] and Solinas [12,13] employed the Frobenius endomorphism to introduce an algorithm for faster scalar multiplication that do not use any point doublings. Extending their ideas, Müller [6] and Smart [11] came up with practical methods which are applicable to elliptic curves over small finite fields of small characteristic.

Recently, Gallant, *et al.* [3] presented a new method for faster scalar multiplication on elliptic curves over (large) prime fields that have an efficiently-computable endomorphism. The key idea of their method is decomposing an

⋆ This work is supported in part by the Ministry of Information & Communication of Korea("Support Project of University Foundation Research $<'$ 00 $>$" supervised by IITA)

D. Naccache and P. Paillier (Eds.): PKC 2002, LNCS 2274, pp. 323–334, 2002.

arbitrary scalar k in terms of an integer eigenvalue λ of the characteristic polynomial of such an endomorphism(See §3). The problem with this method is how efficiently a random integer $k \in [1, n-1]$ could be decomposed into $k = k_1 + k_2\lambda$ modulo n with the bitlengths of k_1 and k_2 half that of k where n is a large prime number. They gave an algorithm for decomposing k into the desired form using the extended Euclidean algorithm and did not derive explicit bounds for decomposition components. However, they expected that the bounds are approximately near to $\sqrt{n}$ on the basis of numerous implementations.

In this paper, we present an alternate algorithm for decomposing an integer k using the theory of μ-Euclidian algorithm. This algorithm runs a little bit faster than that of Gallant *et al.*'s and unlike their algorithm, our algorithm gives explicit bounds for the components. To compare the two algorithms for scalar decomposition we give a precise analysis of all elliptic curves treated in [3].

This paper is arranged as follows. In Section 2, we recall some basic facts on elliptic curves and in Section 3 we briefly discuss the Gallant-Lambert-Vanstone method for comparison with ours. Section 4 is concerned with decomposing an integer k via μ-Euclidian algorithm in the endomorphism rings of elliptic curves. Section 5 contains various examples of elliptic curves and then we give explicit bounds for decomposition components. In the final section we compare two methods to draw our conclusions.

2 Endomorphism Rings

We begin with introducing some basics to elliptic curves. Let $\mathbb{F}_q$ be a finite field of q elements and E be an elliptic curve over $\mathbb{F}_q$ given by a Weierstrass equation

$$E/\mathbb{F}_q : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

with $a_i \in \mathbb{F}_q$. $E(\mathbb{F}_q)$ denotes the set of $\mathbb{F}_q$-rational points on E together with the point at infinity O and $\mathrm{End}(E)$ denotes the ring of $\mathbb{F}_q$-endomorphisms of E. It is well known that (non-supersingular) elliptic curves over finite fields have complex multiplication. Indeed, $\mathrm{End}(E)$ is isomorphic to a complex quadratic order.

The Frobenius endomorphism $\Phi \in \mathrm{End}(E)$ is the morphism given by $\Phi(x, y) = (x^q, y^q)$. It satisfies the quadratic relation $\Phi^2 - t\Phi + q = 0$ in $\mathrm{End}(E)$, where t is called the trace of the Frobenius Φ. More importantly, t is related closely to the order of $E(\mathbb{F}_q)$ by the formula: $\#E(\mathbb{F}_q) = q + 1 - t$. By Hasse's remarkable work on $\#E(\mathbb{F}_q)$, we have

Theorem 1. *Let E be an elliptic curve over $\mathbb{F}_q$ and let n denote the number of $E(\mathbb{F}_q)$, then*

$$|t| = |q + 1 - n| < 2\sqrt{q}.$$

For cryptographic applications, one deals with only non-supersinglar elliptic curves E, so the endomorphism ring of E is an order in the imaginary quadratic field $\mathbb{Q}(\sqrt{t^2 - 4q})$. Hence it is easily seen that $\mathbb{Z}[\Phi] \subset \mathrm{End}(E) \subset \mathbb{Q}(\sqrt{t^2 - 4q})$.

3 Gallant-Lambert-Vanstone Method

Let E be an elliptic curve over $\mathbb{F}_q$ and ϕ be an efficiently-computable endomorphism in End(E). For cryptographic purposes, the order of $E(\mathbb{F}_q)$ must have a large prime factor n. Let $P \in E(\mathbb{F}_q)$ be a point of prime order n. Then the map ϕ acts on the subgroup of $E(\mathbb{F}_q)$ generated by P as a multiplication by λ, where λ is a root of the characteristic polynomial of ϕ modulo n. In place of the Frobenius, Gallant *et al.* exploited ϕ to speed up the scalar multiplication by decomposing an integer k into a sum of the form $k = k_1 + k_2\lambda \pmod{n}$, where $k \in [1, n-1]$ and $k_1, k_2 \approx \sqrt{n}$. Now we compute

$$kP = (k_1 + k_2\lambda)P = k_1P + k_2\lambda P = k_1P + k_2\phi(P).$$

Since $\phi(P)$ can be easily computed, a windowed simultaneous multiple exponentiation applies to $k_1P + k_2\phi(P)$ for additional speedup. It is analyzed in [3] that this method improves a running time up to 66 % compared with the general method, thus it is roughly 50 % faster than the best general methods for 160-bit scalar multiplication. The problem we face is how efficiently a randomly chosen k can be decomposed into a sum of the required form and how explicitly upper bounds of the lengths of the components k_1 and k_2 can be given.

For complete comparison with our method we will now describe the algorithm in [3] for decomposing k out of given integers n and λ. It is composed of two steps. By considering the homomorphism $f : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}_n$ defined by $(i, j) \mapsto (i + j\lambda) \pmod{n}$ we first find linearly independent short vectors $v_1, v_2 \in \mathbb{Z} \times \mathbb{Z}$ such that $f(v_1) = f(v_2) = 0$. As a stage of precomputations this process can be done by the Extended Euclidean algorithm, independently of k. Secondly, one needs to find a vector in $\mathbb{Z}v_1 + \mathbb{Z}v_2$ that is close to $(k, 0)$ using linear algebra. Then (k_1, k_2) is determined by the equation:

$$(k_1, k_2) = (k, 0) - (\lfloor b_1 \rceil v_1 + \lfloor b_2 \rceil v_2),$$

where $(k, 0) = b_1v_1 + b_2v_2$ is represented as an element in $\mathbb{Q} \times \mathbb{Q}$ and $\lfloor b \rceil$ denotes the nearest integer to b. We provide an explicit algorithm in [3] as follows:

Algorithm 1 (**Finding** (k_1, k_2))

Input:	$k \approx n$, the short vectors $v_1 = (x_1, y_1), v_2 = (x_2, y_2)$.
Output:	(k_1, k_2) such that $k \equiv k_1 + k_2\lambda \pmod{n}$.
1)	$D = x_1y_2 - x_2y_1,\ a_1 = y_2k,\ a_2 = -y_1k.$
2)	$z_i = \lfloor a_i/D \rceil$ for $i = 1, 2.$
3)	$k_1 = k - (z_1x_1 + z_2x_2),\ k_2 = z_1y_1 + z_2y_2.$
Return:	(k_1, k_2).

This algorithm takes two round operations and eight large integer multiplications. In [3, Lemma 2], an upper bound of the vector (k_1, k_2) obtained from Algorithm 1 is estimated by the Euclidean norm inequality :

$$\| (k_1, k_2) \| \leq \max(\| v_1 \|, \| v_2 \|).$$

In the procedure of finding two independent short vectors v_1, v_2 such that $f(v_1) = f(v_2) = 0$, Gallant, *et al.* showed $\| v_1 \| \leq 2\sqrt{n}$ but could not estimate $\| v_2 \|$ explicitly. However they expected heuristically that v_2 would also be short. For this reason, they could not give explicit upper bounds of k_1 and k_2 although the lengths of components prove to be near to $\sqrt{n}$ through numerous computational experiments.

4 An Alternate Decomposition of k

We are now describing a new method for decomposing k from a viewpoint of algebraic number theory. Recall that $\mathrm{End}(E)$ is a quadratic order of $K = \mathbb{Q}(\sqrt{-D})(D > 0)$, which is contained in the maximal order of K, denoted $\mathcal{O}_K$. Let ϕ be an efficiently-computable endomorphism in $\mathrm{End}(E)$. Then we have $\mathbb{Z}[\phi] \subset \mathrm{End}(E) \subset \mathcal{O}_K$. Since ϕ is in general not a rational integer, it satisfies a quadratic relation

$$\phi^2 - t_\phi \phi + n_\phi = 0. \tag{1}$$

We assume that the discriminant of ϕ defined by $D_\phi = t_\phi^2 - 4n_\phi$ is of the form $-Dm^2$ for some integer m. As usual, for a point $P \in E(\mathbb{F}_q)$ of a large prime order n we want to perform scalar multiplication kP for $k \in [1, n-1]$. Suppose now that there exists an element $\alpha = a + b\phi \in \mathbb{Z}[\phi]$ such that

$$N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha) = s_n n \ \text{ and } (\alpha)P = O \tag{2}$$

for some positive integer s_n, which is relatively small to n. We then want to decompose a scalar k using a division by α in the μ-Euclidean ring $\mathbb{Z}[\phi]$, where μ is some positive real (see Lemma 2 or [11]). First of all, the existence of such α is guaranteed from the following Lemma.

Lemma 1. *There exists an element $\alpha \in \mathbb{Z}[\phi]$ satisfying (2) for some positive integer $s_n \leq 3n_\phi$. Moreover, $s_n = 1$ when $\mathbb{Z}[\phi]$ is a principal maximal order and n splits in $\mathbb{Q}(\phi)/\mathbb{Q}$.*

Proof. Let $v_1 = (a, b)$ be the short vector constructed in [3] such that $f(v_1) = 0$. Since $f(v_1) = a + b\lambda \equiv 0 \pmod{n}$, it is clear that $(a + b\phi)P = O$. Put $\alpha = a + b\phi$ and $n' = N_{\mathbb{Z}[\phi]/\mathbb{Z}}(a + b\phi) \in \mathbb{Z}$. Then we have $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(a + b\phi) = (a + b\bar{\phi})(a + b\phi) = n'$, so $n'P = (a + b\bar{\phi})(a + b\phi)P = O$. It implies that $n' \equiv 0 \pmod{n}$ and $n' = s_n n$ for some integer s_n. Since $a, b \leq \sqrt{n}$ in [3] and $\mid t_\phi \mid < 2\sqrt{n_\phi}$, we have

$$s_n n = a^2 + abt_\phi + b^2 n_\phi \leq a^2 + \mid abt_\phi \mid + b^2 n_\phi \leq n_\phi(a^2 + \mid ab \mid + b^2) \leq 3n_\phi n.$$

The second assertion follows from [14]. □

Motivated by the work of [3] we give an alternate decomposition of k in terms of ϕ in place of λ in [3]. Viewing a k as an element of $\mathbb{Z}[\phi]$ we divide k by α satisfying (2) in $\mathbb{Z}[\phi]$ and write

$$k = \beta\alpha + \rho$$

with $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho) < \mu N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha)$ for some β and $\rho \in \mathbb{Z}[\phi]$. We then compute

$$kP = (\beta\alpha + \rho)(P) = \beta(\alpha(P)) + \rho(P) = \rho(P).$$

From a representation of ρ, that is, $\rho = k_1 + k_2\phi$, it turns out that

$$kP = \rho P = k_1 P + k_2\phi(P).$$

Since $\phi(P)$ is easily computed we can apply a (windowed) simultaneous multiple exponenciation to yield the same running time improvement as in [3]. Unlike [3] our method gives rigorous bounds for the components k_1, k_2 in term of n_ϕ. To see this, we give the following theorem estimating $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho)$.

Theorem 2. *Let $\alpha = a + b\phi \neq 0 \in \mathbb{Z}[\phi]$. If $\beta \in \mathbb{Z}[\phi]$ then there exist $\delta, \rho \in \mathbb{Z}[\phi]$ such that $\beta = \delta\alpha + \rho$ and $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho) < \mu N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha)$ with*

$$0 < \mu \leq \begin{cases} (9 + 4n_\phi)/16 & \text{if } t_\phi \text{ is odd,} \\ (1 + n_\phi)/4 & \text{if } t_\phi \text{ is even.} \end{cases}$$

Proof. Since $\phi^2 - t_\phi\phi + n_\phi = 0$, we take $\phi = (t_\phi + \sqrt{D_\phi})/2$ where $D_\phi = t_\phi^2 - 4n_\phi$. Put $N_\alpha = N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha)$ and $c = -\lfloor t_\phi/2 \rfloor$. Setting $\phi' = \phi + c$ and changing a $\mathbb{Z}$-basis $\{1, \phi\}$ to $\{1, \phi'\}$, we have $\mathbb{Z}[\phi] = \mathbb{Z}[\phi']$ and

$$\phi' = \begin{cases} (1 + \sqrt{D_\phi})/2 & \text{if } t_\phi \text{ is odd,} \\ \sqrt{D_\phi}/2 & \text{otherwise.} \end{cases}$$

Then α can be written as $a_1 + b_1\phi'$ in term of this new basis. For a given dividend β, we let $\gamma = \beta/\alpha$ and then we have

$$\gamma = \beta/\alpha = \beta\bar{\alpha}/N_\alpha = \frac{x_1 + x_2\phi'}{N_\alpha}$$

where $\bar{\alpha}$ denotes the complex conjugate of α. Take $\delta = y_1 + y_2\phi'$ with $y_i = \lfloor x_i/N_\alpha \rceil$ $(i = 1, 2)$, where $\lfloor x \rceil$ denotes the nearest integer to x. Finally, take $\rho = \alpha(\gamma - \delta)$, then since $\beta = \alpha\gamma, \alpha\delta \in \mathbb{Z}[\phi]$, $\rho \in \mathbb{Z}[\phi]$. It is easily checked that

$$\begin{aligned} N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho)/N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha) = \; & N_{\mathbb{Q}[\phi]/\mathbb{Q}}(\gamma - \delta) \;\leq\; N_{\mathbb{Q}[\phi]/\mathbb{Q}}(\tfrac{1}{2} + \tfrac{1}{2}\phi') \\ = \; & \tfrac{1}{4}N_{\mathbb{Z}[\phi]/\mathbb{Z}}(1 + \phi') = \begin{cases} \tfrac{1}{4}N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\frac{3+\sqrt{D_\phi}}{2}) & \text{if } t_\phi \text{ is odd,} \\ \tfrac{1}{4}N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\frac{2+\sqrt{D_\phi}}{2}) & \text{otherwise.} \end{cases} \\ = \; & \begin{cases} \tfrac{1}{4}(\frac{9-D_\phi}{4}) \leq \tfrac{1}{4}((9 + 4n_\phi)/4) & \text{if } t_\phi \text{ is odd,} \\ \tfrac{1}{4}(\frac{4-D_\phi}{4}) \leq \tfrac{1}{4}((4 + 4n_\phi)/4) & \text{otherwise.} \end{cases} \end{aligned}$$

□

From the proof of Theorem 2, we can produce an efficient algorithm to compute a remainder $\rho = k_1 + k_2\phi$ from k and $\alpha = a + b\phi$. It is also composed of two steps as in [3]. As a stage of precomputations, we first compute

Precomputations

1) $N_\alpha = N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha) = s_n n$, $t_\phi = Tr_{\mathbb{Z}[\phi]/\mathbb{Z}}(\phi)$ and $c = -\lfloor t_\phi/2 \rfloor$.

2) Set $\phi' = \phi + c$. $N = N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\phi')$ and $T = Tr_{\mathbb{Z}[\phi]/\mathbb{Z}}(\phi') = \begin{cases} 1 & \text{if } t_\phi \text{ is odd,} \\ 0 & \text{otherwise.} \end{cases}$

3) $a_1 = a - bc$, $b_1 = b$ (to represent $\alpha = a_1 + b_1\phi'$).

Algorithm 2 (**Divide** k **by** $\alpha = a + b\phi$)

Input:	$k \approx n$ and $N_\alpha, T, N, c, a_1, b_1$.
Output:	$\rho = k_1 + k_2\phi$ such that $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho) < \mu N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha)$.
1)	$x_1 = k(a_1 + b_1 T)$ and $x_2 = -kb_1$.
2)	$y_i = \lfloor x_i / N_\alpha \rceil$ $(i = 1, 2)$.
3)	$k_1' = k - (a_1 y_1 - N b_1 y_2)$ and $k_2' = -(a_1 y_2 + b_1 y_1 + T b_1 y_2)$.
4)	$k_1 = (k_1' + k_2' c)$ and $k_2 = k_2'$.
	Return: k_1, k_2.

Algorithm 2 takes in general two round operations and eight large integer multiplications as in Algorithm 1. But if the values t_ϕ and n_ϕ are rather small, then the values c and N are also expected to be small, which reduces 8 large integer multiplications to 6. From this observation we may expect that the proposed algorithm will be a little bit more efficient than that of [3]. In Table 1 we compare running times of two algorithms applied to Examples 1-4 in §5.1.

Table 1. Comparison of Two Algorithms(on PetiumII 866Mhz)

	$t_\phi = 0$ $n_\phi = 1$	$t_\phi = -1$ $n_\phi = 1$	$t_\phi = 1$ $n_\phi = 2$	$t_\phi = 0$ $n_\phi = 2$
Gallant's Algorithm 1	0.072 ms	0.069 ms	0.071 ms	0.069 ms
Our Algorithm 2	0.053 ms	0.054 ms	0.053 ms	0.054 ms

5 Examples and Upper Bounds

5.1 Examples

In this subsection we list up a family of elliptic curves over a large prime field $\mathbb{F}_p$ with efficient endomorphisms treated in [3] and give the characteristic polynomial of such an endomorphism in each case.

Example 1. Let $p \equiv 1 \pmod 4$ be a prime, and let E_1 be an elliptic curve defined by

$$E_1/\mathbb{F}_p : y^2 = x^3 + ax.$$

Let $\beta \in \mathbb{F}_p$ be an element of order 4. Then the map $\phi : E_1 \to E_1$ defined by $(x, y) \mapsto (-x, \beta y)$ and $O \mapsto O$ belongs to $\mathrm{End}(E_1)$. Moreover, it is easily seen that ϕ satisfies the quadratic equation

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

so $t_\phi = 0, n_\phi = 1$ and $\mathrm{End}(E_1)$ is isomorphic to $\mathbb{Z}[\phi]$, the maximal order of $\mathbb{Q}(\sqrt{-1})$.

Example 2. Let $p \equiv 1 \pmod 3$ be a prime, and let E_2 be an elliptic curve defined by

$$E_2/\mathbb{F}_p : y^2 = x^3 + b.$$

Let $\gamma \in \mathbb{F}_p$ be an element of order 3. Then the map $\phi : E_2 \to E_2$ defined by $(x, y) \mapsto (\gamma x, y)$ and $O \mapsto O$ is an endomorphism defined over $\mathbb{F}_p$. Moreover, the quadratic equation of ϕ is given by

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

so $t_\phi = -1, n_\phi = 1$ and $\mathrm{End}(E_2)$ is isomorphic to $\mathbb{Z}[\phi]$, the maximal order of $\mathbb{Q}(\sqrt{-3})$.

It is noted in both Examples 1 and 2 that the map ϕ can be easily computed using only one multiplication in $\mathbb{F}_p$.

Example 3. Let $p > 3$ be a prime such that -7 is a perfect square in $\mathbb{F}_p$, and let $\omega = (1 + \sqrt{-7})/2$, and let $a = (\omega - 3)/4$. Let E_3 be an elliptic curve defined by

$$E_3/\mathbb{F}_p : y^2 = x^3 - \frac{3}{4}x^2 - 2x - 1.$$

Then the map $\phi : E_3 \to E_3$ defined by

$$(x, y) \mapsto (\omega^{-2}\frac{x^2 - \omega}{x - a},\ \omega^{-3}y\frac{x^2 - 2ax + \omega}{(x - a)^2})$$

and $O \mapsto O$ belongs to $\mathrm{End}(E_3)$. Moreover, ϕ satisfies

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

so $t_\phi = 1, n_\phi = 2$ and $\mathrm{End}(E_3)$ is isomorphic to $\mathbb{Z}[\phi]$, the maximal order of $\mathbb{Q}(\sqrt{-7})$.

Example 4. Let $p > 3$ be a prime such that -2 is a perfect square in $\mathbb{F}_p$. Let E_4 be an elliptic curve defined by

$$E_4/\mathbb{F}_p : y^2 = 4x^3 - 30x - 28.$$

Then the map $\phi : E_4 \to E_4$ defined by

$$(x, y) \mapsto (-\frac{2x^2 + 4x + 9}{4(x + 2)},\ -\frac{2x^2 + 8x - 1}{4\sqrt{-2}(x + 2)^2}y)$$

and $O \mapsto O$ belongs to $\mathrm{End}(E_4)$. Moreover, the quadratic equation of ϕ is given by

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

so $t_\phi = 0, n_\phi = 2$ and $\mathrm{End}(E_4)$ is isomorphic to $\mathbb{Z}[\phi]$, the maximal order of $\mathbb{Q}(\sqrt{-2})$.

In Examples 3 and 4, computing an endomorphism is a little harder than doubling a point.

5.2 Upper Bounds on the Components k_1, k_2

Now we restrict ourselves to elliptic curves $E(\mathbb{F}_p)$ only in the previous subsection. For cryptographic applications, let P be a point of $E(\mathbb{F}_p)$ of large prime order n, so $\#E(\mathbb{F}_p) = hn$ where h is called the cofactor of $E(\mathbb{F}_p)$. Recall that for each $1 \le i \le 4, \mathrm{End}(E_i) = \mathbb{Z}[\phi]$ is the maximal order of $\mathbb{Q}(\sqrt{-D})$ where $D = 1, 3, 7$ or 2, respectively. By Lemma 1 there exists an element $\alpha = a + b\phi \in \mathbb{Z}[\phi]$ such that $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha) = n$ and $(\alpha)P = O$. Finding such an α boils down to solving out a quadratic equation in $\mathbb{Z}[\phi]$. Indeed, this process can be done using the known methods such as Shanks' algorithm [9] and lattice reduction method [10]. Especially, one can also represent n, which splits in $\mathbb{Q}(\sqrt{-D})/\mathbb{Q}$, by the principal form only by using the Cornacchia's algorithm [2]. We use Theorem 2 to give explicit upper bounds on μ in the μ-Euclidean ring $\mathbb{Z}[\phi]$.

Lemma 2. *Let $\alpha = a + b\phi$ such that $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha) = n$. For any integer k, there exists a remainder $\rho \in \mathbb{Z}[\phi]$ such that $k = \beta\alpha + \rho$ for some $\beta \in \mathbb{Z}[\phi]$ with*

$$N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho) \le \begin{cases} n/2 & \text{for } E_1, \\ 3n/4 & \text{for } E_2, \\ n & \text{for } E_3, \\ 3n/4 & \text{for } E_4. \end{cases}$$

Proof. Recall that t_ϕ is even for E_1 and E_4, and t_ϕ is odd for E_2 and E_3. From the proof of Theorem 2, we get

$$N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho)/N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\alpha) \le \begin{cases} \frac{1}{4}(\frac{9-D_\phi}{4}) & \text{if } t_\phi \text{ is odd}, \\ \frac{1}{4}(\frac{4-D_\phi}{4}) & \text{if } t_\phi \text{ is even}, \end{cases}$$

which gives the desired result. □

Finally, Lemma 2 gives explicit upper bounds on the components of k.

Theorem 3. *For any k, let ρ be a remainder of k divided by α using Algorithm 2 and write $\rho = k_1 + k_2\phi$. Then we have*

$$\max\{|k_1|, |k_2|\} \le \begin{cases} \sqrt{n/2} & \text{for } E_1, \\ \sqrt{n} & \text{for } E_2, \\ \sqrt{8n/7} & \text{for } E_3, \\ \sqrt{3n}/2 & \text{for } E_4. \end{cases}$$

Proof. In case of E_1, it is easy to see that $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho) = N_{\mathbb{Z}[\phi]/\mathbb{Z}}(k_1 + k_2\phi) = k_1^2 + k_2^2$. Lemma 2 immediately gives $k_1^2 + k_2^2 \le n/2$, which completes the proof for E_1.

In case of E_2, we have $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(\rho) = k_1^2 + k_2^2 - k_1k_2$. If $k_1k_2 \le 0$ then it follows from Lemma 2 that $\max\{|k_1|, |k_2|\} \le \sqrt{3n/4}$. Assume $k_1k_2 > 0$ and $|k_2| \ge |k_1| > 0$. Then by Lemma 2, we easily deduce

$$k_1^2 + k_2^2 - k_1k_2 = |k_1|^2 + |k_2|^2 - |k_1||k_2| = (|k_1| - \frac{1}{2}|k_2|)^2 + \frac{3}{4}|k_2|^2 \le 3n/4.$$

Hence $|k_1|^2 \le |k_2|^2 \le n$ implies that $\max\{|k_1|, |k_2|\} = |k_2| \le \sqrt{n}$, completing the proof for E_2. The other cases are also done similarly. □

6 Comparisons of the Two Methods and Conclusion

6.1 Comparisons

In this section we compare the two methods by decomposing many integral scalars on all elliptic curves in Section 5. To protect Pohlig-Hellman attack [8] the group order of $E(\mathbb{F}_p)$ has a large prime factor n at least 160-bit. The problem of determining the group order of a given elliptic curve is not an easy task in general but thanks to an improved Schoof's algorithm one can figure out the group order of an elliptic curve. However, in the case where the endomorphism ring is known, computing the group order of $E(\mathbb{F}_p)$ is rather easy and it is explicitly given by a well known formula in [4]. Conversely, determining the elliptic curve having a given group order is not easy. For this reason, it is not easy to take 'cryptographically good' elliptic curves whose the group order has a large prime factor n and has a small cofactor. Without knowing the exact group order of elliptic curves we here decompose scalars by the two methods under the assumption that elliptic curves in consideration are good cryptographically. Indeed our method gives a decomposition of a scalar if only we know the quadratic equation satisfied by an efficient endomorphism on elliptic curves.

For each example in subsection 5.1, we considered various primes p where p is the norm of some element $\pi \in \mathbb{Z}[\phi]$ satisfying $N_{\mathbb{Z}[\phi]/\mathbb{Z}}(1-\pi) = nh$ for a large prime n and a small h. We then decomposed 10^5 random integers $k \in [1, n-1]$. In an appendix we put a list of tables showing comparable data in two decompositions. Here we briefly describe implementation results. For Example 1 the two decompositions are identically same for 20 different primes p. In other examples different decompositions of k occurred but for most of scalars k the decompositions are exactly same and in different cases the length differences for components are within 2 bits because the ratios of maximum lengths are less than 3, so it makes no big difference in applying the simultaneous windowed techniques. On the whole, we can analyze that the two decompositions are same for more than 80 % out of all cases we have investigated. In different decompositions, the length differences are almost negligible.

6.2 Conclusion

We described an alternate method of decomposing k using the theory of μ-Euclidian algorithm. The proposed method gives not only a different decomposition of a scalar k but also produces explicit upper bounds for the components by computing norms in the complex quadratic orders. We then compare the two different methods for decomposition through computational implementations. From these we conclude that the two decompositions are same for most of cases of elliptic curves we have considered. Even in different decompositions of a same scalar, the two methods makes no big difference in a sense that the length differences of components are very small. So this shows that the algorithm of [3] runs smoothly with desired bounds for components, as expected.

References

1. Ian Blake, Gadiel Seroussi and Nigel Smart, 'Elliptic Curves in Cryptography', London Mathematical Society Lecture Note Series. 265, Cambridge University Press, (1999).
2. G. Cornacchia, *"Su di un metodo per la risoluzione in numeri interi dell' equazione* $\sum_{h=0}^{n} C_h x^{n-h} y^h = P$", Giornale di Matematiche di Battaglini, 46, (1908),33-90.
3. R. Gallant, R. Lambert and S. Vanstone, *"Faster Point Multiplication on Elliptic Curves with Efficient Endomorphisms"*, Advances in Cryptology-Crypto 2001, LNCS 2139, Springer-Verlag (2001), 190-200.
4. K.Ireland and M.Rosen, 'A classical introduction to modern number theory', Graduate Texts in Mathematics, vol 84, Springer-Verlag, (1982).
5. N. Koblitz, *"CM-curves with good cryptographic properties"*, Advances in Cryptology-Crypto '91, LNCS 576, Springer-Verlag (1992), 279-287.
6. V. Müller, *" Fast multiplication in elliptic curves over small fields of characteristic two"*, Journal of Cryptology, **11** (1998), 219-234.
7. W. Meier and O. Staffelbach, *"Efficient multiplication on certain nonsupersingular elliptic curves"*, Advances in Cryptology-Crypto'92, Springer-Verlag (1992), 333-344.
8. S. Pohlig, M. Hellman,*An improved algorithm for computing logarithms over* $GF(p)$ *its cryptographic significance,"*, IEEE Trans. Inform. Theory, **24** (1978), 106-110.
9. D. Shanks, *"Five number theoretic algorithms"* In Proc. 2nd Manitoba Conference on Numerical Mathematics (1972), 51-70.
10. B.Vallée,*Une approche géométrique des algorithmes de réduction des réseaux en petite dimension"*, (1986) Thése, Université de Caen.
11. N. Smart, *"Elliptic curve cryptosystems over small fields of odd characteristic "*, Journal of Cryptology, **12** (1999), 141-145.
12. J. Solinas, *"An improved algorithm for arithmetic on a family of elliptic curves"*, Advances in Cryptology-Crypto '97, LNCS 1294, Springer-Verlag (1997), 357-371.
13. J. Solinas, *"Efficient arithmetic on Koblitz curves",* Design, Codes and Cryptography, **19** (2000), 195-249.
14. I. Stewart and D. Tall, 'Algebraic Number Theory', Chapman and Hall, Halsted Press, (1979).

Appendix: Implementation Results

We list up tables showing comparable data in two decompositions. For each example in subsection 5.1 we considered 3 different primes p and then decomposed 10^5 random integers $k \in [1, n-1]$ for each p. Each example consists of 3 tables and each table pairs with two parts. One part in each table consists of 4 data, p, n, α and λ. The other part shows the degree of likeness in two decompositions. It contains the ratio of the same decompositions to different ones and the ratio of maximum lengths, $\max_{\{i=1,2\}}$ to $\sqrt{n}$ where $\max_1$ denotes the maximum length of the components by our method and $\max_2$ denotes that by Gallant *et al.*'s method.

Example 1: $\phi^2 + 1 = 0$

p	24356507785617894243567506464865356531982256141
n	12178253892808947121783712971871627547782377878 1
λ	46308316286753456460287381300232203960042557786
α	34402088421024910517643 0 + 58584726296944062172859 ϕ

same	different	$\max_1 / \sqrt{n}$	$\max_2 / \sqrt{n}$
100%	0%	0.576	0.576

p	732185185678629514184124961158879780124880223 77
n	366092592839314757092063723562912834023279404 73
λ	19769856633674487989568880377360877117950324407
α	18232450329907027798804 8 + 58026156004674194058763 ϕ

same	different	$\max_1 / \sqrt{n}$	$\max_2 / \sqrt{n}$
100%	0%	0.628	0.628

p	39306788880749977418085952526890142297147956663 37
n	19653394440374988709042975550407314242331941753793
λ	13098140680635734402854668567608237356814672477 54
α	95500378639872919560995 3+ 10263075620892546516893 28 ϕ

same	different	$\max_1 / \sqrt{n}$	$\max_2 / \sqrt{n}$
100%	0%	0.706	0.705

Example 2: $\phi^2 + \phi + 1 = 0$

q	12206619750066739109030678133812471429623409967 67
n	30516549375166847772576723956401265253533303951 11
λ	25683076175890603286873003602277449197813683329 5
α	13427969703513498583905 -5455814623265624931240 29]ϕ

same	different	$\max_1 > \max_2$	$\max_1 / \sqrt{n}$	$\max_2 / \sqrt{n}$
100 %	0 %	0 %	0.517	0.517

p	95022144632423596840305985525769665138717618573 9
n	31674048210807865613435371239381243362853858688 3
l	28235148507789851373783269511181887263548346457 5
α	42730791354983281319163 7 -21037014910981405645074 9 ϕ

same	different	$\max_1 > \max_2$	$\max_1 / \sqrt{n}$	$\max_2 / \sqrt{n}$
75%	25%	21 %	0.938	0.743

p	10214108835130582912738498272919391375692775122 3
n	10214108835130582912738443781643924839062447825 1
λ	29377624209728221104940825415138564729613238982
α	19290277620969726088368 9-17600498922783452119464 1 ϕ

same	different	$\max_1 > \max_2$	$\max_1 / \sqrt{n}$	$\max_2 / \sqrt{n}$
74%	26%	14%	0.878	0.844

Example 3: $\phi^2 - \phi + 2 = 0$

p	122066197500667391090306781338124714296234099676 7
n	305165493751668477725767239564012652535330395111
λ	256830761758906032868730036022774491978136833295
α	134279697035134985839 05 -545581462326562493124029 ϕ

same	different	$\max_1 > \max_2$	$\max_1/\sqrt{n}$	$\max_2/\sqrt{n}$
100%	0%	0%	0.518	0.518

p	27811890929441973874395311564036633313232309032 97
n	92706303098139912914650953243482719649855589678 1
λ	21918279997822803296613353806449073940359362677 5
α	853033060580609187737651 -190989728098028002286769ϕ

same	different	$\max_1 > \max_2$	$\max_1/\sqrt{n}$	$\max_2/\sqrt{n}$
75%	25%	24%	0.982	0.640

p	95022144632423596840305985525769665138717618573 9
n	31674048210807865613435371239381243362853858688 3
λ	28235148507789851373783269511181887263548346457 5
α	427307913549832813191637-2103701491098140564507 49 ϕ

same	different	$\max_1 > \max_2$	$\max_1/\sqrt{n}$	$\max_2/\sqrt{n}$
75%	25%	21%	0.941	0.750

Example 4: $\phi^2 + 2 = 0$

p	56363293711595169407644604885168816934193385874 7
n	28181646855797584703822270466344994215707990897 1
λ	89410463644172197664541344572565104224954335021
α	480304564005069232838013-159881197071256943510201 ϕ

same	different	$\max_1 > \max_2$	$\max_1/\sqrt{n}$	$\max_2/\sqrt{n}$
100%	0%	0%	0.753	0.753

p	68146790876522902430524762995908603350973574645 1
n	34073395438261451215262315574947704358870047811 3
λ	92899111242628958306702491539054896203949387582
α	351342873047286962313279-329615032986583083697556 ϕ

same	different	$\max_1 > \max_2$	$\max_1/\sqrt{n}$	$\max_2/\sqrt{n}$
75%	25%	13%	0.864	0.864

p	56363293711595169407644604885168816934193385874 7
n	28181646855797584703822270466344994215707990897 1
λ	89410463644172197664541344572565104224954335021
α	480304564005069232838013-159881197071256943510 2 ϕ

same	different	$\max_1 > \max_2$	$\max_1/\sqrt{n}$	$\max_2/\sqrt{n}$
100%	0%	0%	0.752	0.752