Scalar Multiplication on Elliptic Curves by Frobenius Expansions

Jung Hee Cheon, Sangjoon Park, Choonsik Park, and Sang Geun Hahn

CONTENTS

ABSTRACT

Koblitz has suggested to use "anomalous" elliptic curves defined over $\mathbb{F}_2$, which are non-supersingular and allow for efficient multiplication of a point by an integer. For these curves, Meier and Staffelbach gave a method to find a polynomial of the Frobenius map corresponding to a given multiplier. Muller generalized their method to arbitrary non-supersingular elliptic curves defined over a small field of characteristic 2. In this paper, we propose an algorithm to speed up scalar multiplication on an elliptic curve defined over a small field. The proposed algorithm uses the same technique as Muller's to get an expansion by the Frobenius map, but its expansion length is half of Muller's due to the reduction step (Algorithm 1). Also, it uses a more efficient algorithm (Algorithm 3) to perform multiplication using the Frobenius expansion. Consequently, the proposed algorithm is two times faster than Muller's. Moreover, it can be applied to an elliptic curve defined over a finite field with odd characteristic and does not require any precomputation or additional memory.

I. INTRODUCTION

Elliptic Curve Cryptosystems were developed by Miller and Koblitz [6], independently, in 1985. They are based on the fact that the discrete logarithm problem is intractable on the group of points on an elliptic curve defined over a finite field. To make the elliptic curve cryptosystem practical, there were several problems. One of them is to speed up scalar multiplication, adding a point P in an elliptic curve to itself n times (denoted by nP). This is important because the cost of executing elliptic curve public key cryptosystems depends mostly on the complexity of the scalar multiplication. Especially, this is crucial for the equipment such as smart cards which requires high speed and small size of memory.

In this paper, we propose an algorithm to speed up scalar multiplication of an elliptic curve defined over a small field. The proposed algorithm uses the same technique as Muller's to get an expansion by the Frobenius map, though we established it without knowing his work. However, its expansion length is half of Muller's due to the reduction step (Algorithm 1) which was introduced by Meier and Staffelbach [7] for the anomalous binary curve defined over $\mathbb{F}_2$. Also, it uses a more efficient algorithm (Algorithm 3) to perform multiplication using the Frobenius expansion. Consequently, the proposed algorithm is two times faster than Muller's. Moreover, it can be applied for an elliptic curve defined over finite fields with odd characteristic and does not require any precomputation and additional memory.

We can, for example, multiply a point of $E(\mathbb{F}_{2^{4n}})$ by an integer in at most (not at average) $l/4+7$ elliptic additions if the elliptic curve is defined over $\mathbb{F}_{2^4}$ and has the order less than 2^l in $\mathbb{F}_{2^{4n}}$. Our method is the fastest one among all scalar multiplications of non-singular elliptic curves in the literature.

In Section II, we recall previous work for scalar multiplication on an elliptic curve. In Section III, we show that any integer multiplication can be expressed by the polynomial of the Frobenius map when the elliptic curve is defined over a small finite field, even if it has odd characteristic. Here the expansion length of an integer is a half of Muller's due to the reduction step. In Section IV, we propose an algorithm to perform scalar multiplication using the Frobenius expansion. The algorithm is efficient when elements of the finite field can be represented in a normal basis. In Section V, we present elliptic curves which are suitable for public key cryptosystems. Section VI concludes this paper.

II. PREVIOUS WORKS

Scalar multiplication on an elliptic curve is analogous to exponentiation in the multiplicative group of integers modulo a fixed integer m. To speed up the operation, it is natural to apply the various techniques

which are developed to speed up modular exponentiation. Such methods, for the most part, carry over to elliptic scalar multiplication. There are the binary method, the k-ary method, and several methods using precomputation and memory. We recall some of them.

1. Multiplication for General Elliptic Curves

1. The binary method [9] is to express the multiplier m as binary representation and to repeat doubling and adding. It requires at most $2l$ and at average $3l/2$ elliptic operations to multiply a point by an l-bit integer m.

2. The addition-subtraction method (or the signed binary method) [9] uses the fact that the subtraction is as fast as the addition in an elliptic curve, whereas the modular inverse, corresponding to the elliptic subtraction, is four times or more slower than the modular multiplication. This method is to express a multiplier m as a non-adjacent form (NAF) and repeat doubling and adding (or subtracting). It requires at most $3l/2$ and at average $4l/3$ elliptic operations to multiply a point by an l-bit integer m.

3. The k-ary method [2], [3] requires precomputation and memory. To perform this method, first precompute and store points multiplied by i for $1 \leq i < 2^k$. Next, express a multiplier m as a 2^k-radix representation and to repeat doubling and adding by the stored points. It requires at most $l + l/k + 2^k - 2$ elliptic operations, which is less than $4l/3$ for large l and small $k \geq 3$, to multiply a point by an l-bit integer m.

4. The improved k-ary method [2] is the same as the k-ary method, except that computing repeated point doublings is carried out directly as a closed formula rather than by individual point doublings. It can reduce the number of inversions, which take almost five times longer than multiplications of elements in a finite field.

2. Multiplication for Special Elliptic Curves

There are also efficient methods available for special families of elliptic curves. If an elliptic curve is defined over a small field of characteristic 2, we can represent the multiplier m by the Frobenius map ϕ, which is easily evaluated as a bit-rotate when a point is represented in a normal basis. Using this fact, some methods were developed.

1. Koblitz [5] first proposed the method to use the Frobenius map. He showed that any integer multiplication can be expressed by a polynomial of the Frobenius map when the elliptic curve is defined over $\mathbb{F}_2$ and has the order 2 or 4 over $\mathbb{F}_2$ (It is called an anomalous

binary curve over $\mathbb{F}_2$). His method requires at most $2l$ and at average $3l/4$ elliptic operations to multiply a point by an l-bit integer m.

2. Meier and Staffelbach [7] presented an algorithm to find the polynomial of the Frobenius map with coefficient $0, \pm 1$ for a multiplier m, if the elliptic curve is an anomalous binary curve over $\mathbb{F}_2$. Their method requires at most l and at average $l/2$ elliptic operations to multiply a point by an l-bit integer m.

3. Solinas [16] introduced the notion of NAF to the method of Meier and Staffelbach's. He improved their method so that no adjacent terms of the polynomial have non-zero coefficients. His method requires at most $l/2$ and at average $l/3$ elliptic operations to multiply a point by an l-bit integer m.

4. Cheon *et al.* [1] combined Koblitz's idea and the k-ary method. For an elliptic curve E defined over $\mathbb{F}_{2^k}$, they computed the multiplication by 2^k, rather than by closed formula, by the identity $2^k = t\phi - \phi^2$ with $t = 2^k + 1 - \#E(\mathbb{F}_{2^k})$. Their method can be applied for all elliptic curves defined over a small field, but it is efficient only if t is small. It requires at most $2l/5 + 28$ elliptic operations to multiply a point by an l-bit integer m for some elliptic curves.

5. Muller [10] generalized the method of Meier and Staffelbach's for the anomalous binary curves over $\mathbb{F}_2$ to all non-supersingular elliptic curves defined over a small field of characteristic 2. He presented an algorithm to get an expansion, say Frobenius expansion, of the multiplier m by the Frobenius map with small coefficients for elliptic curves defined over a small field of characteristic 2. Also, he gives an algorithm to efficiently compute the multiplication by m using Frobenius expansions without assuming that elements of the finite field are represented in a normal basis.

III. FROBENIUS EXPANSION OF m

In this section, we give a theorem to enable us to expand a multiplier m by the Frobenius map with small expansion length and small coefficients. This theorem is constructive enough to give an algorithm to compute scalar multiplications efficiently.

At first, we introduce the Frobenius map. Consider an elliptic curve E defined over $\mathbb{F}_q$ with q elements. We define the q-th power Frobenius map ϕ_q on $E(\mathbb{F}_q)$ as follows:

$$\phi_q : (x, y) \mapsto (x^q, y^q)$$

Then the following are equivalent (See [14]):

- $\#E(\mathbb{F}_q) = q + 1 - t$

- The trace of ϕ_q is t
- $\phi_q^2 - t\phi_q + q = 0$ in the ring of endomorphisms of E

In particular, E is supersingular if $t = 0$ and E is called an anomalous curve if $t = 1$. From now on, we write ϕ_q as ϕ unless otherwise specified.

Theorem 1. *For any non-supersingular elliptic curve E defined over $\mathbb{F}_q$, let E_n be the curve regarded over the extension field $\mathbb{F}_{q^n}$, $n \geq 2$. Then on E_n multiplication by an integer m can be expressed as*

$$m = \sum_{j=0}^{n+2} c_j \phi^j, \tag{1}$$

with $c_j \in \{i \in \mathbb{Z} | -q/2 < i \leq q/2\}$ for $n \geq \delta(q)$, where $\delta(q) = 1$ for $q > 5$, $\delta(5) = 2$, $\delta(4) = 3$.

If $q \geq 2^4$, the expansion length becomes shorter as follows:

$$m = \sum_{j=0}^{n+1} c_j \phi^j. \tag{2}$$

The proof proceeds in several steps. First observe that the Frobenius map satisfies the equation $\phi^2 - t\phi + q = 0$ for $t = q + 1 - \#E(\mathbb{F}_q)$, and that there is a natural homomorphism from the ring $\mathbb{Z}[\alpha] = \{a + b\alpha | a, b \in \mathbb{Z}\} \subset \mathbb{C}$ to the endomorphism ring $End(E)$ of E which maps $\alpha = (t + \sqrt{t^2 - 4q})/2$ to ϕ. Note that $t^2 - 4q < 0$ since E is non-supersingular. Thus if we have an expansion $m = \sum_j c_j \alpha^j$ in $\mathbb{Z}[\alpha]$, we immediately get a corresponding expansion $m = \sum_j c_j \phi^j$ in $End(E)$. This means that $mP = \sum_j c_j \phi^j(P)$ for every point P on E_n.

Observe that Theorem 1 holds only when $q > 3$. When E is defined over $GF(q)$ for $q = 2$ or 3, we can get a Frobenius expansion of m by applying Theorem 1 for E considered to be defined over $GF(q^2)$. In that case, we get an expansion of m as

$$m = \sum_{j=0}^{\lceil \frac{n}{2} \rceil + 2} c_j \phi_{q^2}^j, \quad c_j \in \{i \in \mathbb{Z} | -q^2/2 < i \leq q^2/2\}.$$

Also, if we distinguish the cases as in Lemma 3 of [7], we can get a method to expand by ϕ_q, not by ϕ_{q^2}, for $q = 2, 3$. However, we do not introduce the method since it is the same as in [7] and the former method is more efficient.

Let $N(\cdot)$ be the norm from $\mathbb{Q}(\sqrt{t^2 - 4q})$ to $\mathbb{Q}$ with $N(a + b\alpha) = |a + b\alpha|^2 = a^2 + tab + qb^2, a, b \in \mathbb{Z}$.

Lemma 1. *For any $s \in \mathbb{Z}[\alpha]$, there is an element $s_1 \in \mathbb{Z}[\alpha]$ and $u \in \{i \in \mathbb{Z} | -q/2 < i \leq q/2\}$ such that*

$$s = s_1\alpha + u. \tag{3}$$

Moreover, u is unique and $|s_1| \leq |s|/\sqrt{q} + \sqrt{q}/2$.

Proof. Let $s = x + y\alpha$ $(x, y \in \mathbb{Z})$, u be an integer with $-q/2 < u \leq q/2$ such that $u \equiv x \mod q$, and $v = (x - u)/q \in \mathbb{Z}[\alpha]$. Using $q = \alpha(t - \alpha)$, we have

$$s - u = (x - u) + y\alpha = vq + y\alpha = (vt + y - v\alpha)\alpha \tag{4}$$

Hence we have $s = s_1\alpha + u$ for $s_1 = vt + y - v\alpha \in \mathbb{Z}[\alpha]$.

Assume there are u and u' satisfying $s = s_1\alpha + u$ and $s = s_1'\alpha + u'$. Then α divides $u - u'$. Since $N(\alpha) = q$, q must divides $u - u'$.

The last assertion follows from $|s_1\alpha| = |s-u| \le |s| + |u| \le |s| + q/2$. □

Lemma 1 enables us to perform some kind of division algorithm. Using this, we can expand a number s in $\mathbb{Z}[\alpha]$ by Frobenius maps.

Lemma 2. *Let $q \ge 4$ and $q \nmid t$. For any $s \in \mathbb{Z}[\alpha]$ with norm $|s| \le q^{k/2}$, $k \in \mathbb{N}$, there is an expansion*

$$s = \sum_{j=0}^{k+1} c_j \alpha^j \tag{5}$$

with $c_j \in \{i \in \mathbb{Z} | -q/2 < i \le q/2\}$. For $q \ge 2^4$, the expansion length becomes shorter as follows:

$$s = \sum_{j=0}^{k} c_j \alpha^j. \tag{6}$$

Proof. For $k = 1$, s is an integer with the absolute value less than $q/2$ since $\sqrt{q} \le q/2$. Then s trivially satisfies the statement of the lemma.

Assume $k > 1$. By Lemma 1, we know that for any $s \in \mathbb{Z}[\alpha]$, there is an element $s_1 \in \mathbb{Z}[\alpha]$ such that

$$s = s_1\alpha + c_0, \quad |s_1| \le |s|/\sqrt{q} + \sqrt{q}/2, \tag{7}$$

with $c_0 \in \{i \in \mathbb{Z} | -q/2 < i \le q/2\}$. Repeatedly,

$$s_1 = s_2\alpha + c_1, \quad |s_2| \le |s_1|/\sqrt{q} + \sqrt{q}/2, \tag{8}$$

with $c_1 \in \{i \in \mathbb{Z} | -q/2 < i \le q/2\}$. If we continue this process repeatedly, we get the coefficients c_j of a Frobenius expansion of s. We assert that this process terminates in $k+1$ steps, i.e.,

$$s_k = s_{k+1}\alpha + c_k, \quad |s_{k+1}| < \sqrt{q} \tag{9}$$

with $c_k \in \{i \in \mathbb{Z} | -q/2 < i \le q/2\}$. Since any element, whose norm is less than q, in $\mathbb{Z}[\alpha]$ is an integer, we can take $c_{k+1} = s_{k+1}$. Thus

$$s = \sum_{j=0}^{k+1} c_j \alpha^j \tag{10}$$

with $c_j \in \{i \in \mathbb{Z} | -q/2 < i \le q/2\}$, which is the statement of the lemma.

Hence it suffices to prove $|s_{k+1}| < \sqrt{q}$. Note that $|s_{j+1}| \le |s_j|/\sqrt{q} + \sqrt{q}/2$ for $j \ge 0$ ($s_0 = s$). Let $\{a_j\}$ be a sequence satisfying $a_0 = |s_0|$ and $a_{j+1} = a_j/\sqrt{q} + \sqrt{q}/2$. Since the expressions $a_j - \frac{q}{2(\sqrt{q}-1)}$ $(j = 0, 1, 2, \cdots)$ form a geometric progression with the common ratio $1/\sqrt{q}$, a_j has the general term

$$a_j = |s_0| q^{-j/2} + \frac{q}{2(\sqrt{q}-1)}(1 - q^{-j/2}) \quad (j \ge 0).$$

Hence for $q \ge 7$ we have

$$\begin{aligned} |s_{k+1}| &\le a_{k+1} \\ &\le |s_0| q^{-(k+1)/2} + \frac{q}{2(\sqrt{q}-1)}(1 - q^{-(k+1)/2}) \\ &\le 1/\sqrt{q} + \frac{q}{2(\sqrt{q}-1)} \\ &< \sqrt{q}. \end{aligned}$$

For $q = 4, 5$, we have $|s_k| < 3$. If s_k is not an integer with $|s_k| < \sqrt{q}$, then s_k should be ± 3 or of the form $a \pm \alpha$ for an integer a with $|a| < \sqrt{q}$, because $q \nmid t$. In all cases, s_{k+1} is an integer with $|s_{k+1}| < \sqrt{q}$.

For $q \ge 2^4$, we have $|s_k| \le 1 + \frac{q}{2(\sqrt{q}-1)} < \sqrt{q}$, which completes the proof. □

By the above lemma, we get a Frobenius expansion of s. But an integer m near q^n has the norm near q^{2n}. To reduce m to some number m' with small norm, we need another division algorithm.

Lemma 3. *Assume* $|t| < 2\sqrt{q}$. *For any* $s, a \in \mathbb{Z}[\alpha]$, $a \neq 0$, *there exist* $b, c \in \mathbb{Z}[\alpha]$ *such that*

$$s = ba + c, \quad |c| \leq (\frac{1+\sqrt{q}}{2})|a|. \qquad (11)$$

Proof. Given any $s \in \mathbb{Z}[\alpha]$, For $z = s/a$, we can write $z = x' + y'\alpha$ for

$$x' = Re(z) - t\frac{Im(z)}{4q - t^2} \qquad (12)$$

$$y' = \frac{2Im(z)}{4q - t^2} \quad (x', y' \in \mathbb{R}). \qquad (13)$$

Let x, y be the nearest integers to x', y', respectively. Then we have $z = (x + y\alpha) + (x' - x + (y' - y)\alpha)$ with $x, y \in \mathbb{Z}$. If we let $b = x + y\alpha$, $c = (x' - x + (y' - y)\alpha)a$, then $z = b + c/a$ so that $s = ba + c$. Moreover, $b = x + y\alpha \in \mathbb{Z}[\alpha]$, $c = s - ba \in \mathbb{Z}[\alpha]$ and $|c| \leq (|x' - x| + |y' - y||\alpha|)|a| \leq (\frac{1+\sqrt{q}}{2})|a|$, which proves the lemma. □

Proof. (Theorem 1) Now we are in position to prove Theorem 1. As the curve E_n is regarded over the extension field $\mathbb{F}_{q^n}$, the Frobenius map satisfies the equation $\phi^n = 1$. It follows that for any α-expansion which is congruent modulo $\alpha^n - 1$, the corresponding ϕ-expansion yields the same endomorphism on E_n.

Therefore we first compute the remainder m' of the division of m by $\alpha^n - 1$ by Lemma 3 and compute an α-expansion of m' such that

$$m = b(\alpha^n - 1) + m' \quad b, m' \in \mathbb{Z}[\alpha]. \qquad (14)$$

According to Lemma 3,

$$|m'| \leq (\sqrt{q} + 1)/2 \cdot |\alpha^n - 1| = (\sqrt{q} + 1)/2 \cdot \#E(\mathbb{F}_{q^n})$$

since $\#E(\mathbb{F}_{q^n}) = N(\alpha^n - 1)$. By Hasse's theorem [14], we know $\#E(\mathbb{F}_{q^n}) \leq q^n + 1 + 2\sqrt{q}^n$ so that

$$|m'| \leq (\sqrt{q}+1)(\sqrt{q}^n + 1)/2.$$

For $q > 5$, we have $(\sqrt{q}+1)(\sqrt{q}^n+1)/2 \leq q^{(n+1)/2}$ for $n \geq 1$ so that the theorem in case $q > 5$ follows from Lemma 2. For $q = 4, 5$, we can prove by induction on n that

$$(\sqrt{q}+1)(\sqrt{q}^n+1)/2 \leq q^{(n+1)/2} \quad \text{for } n \geq \delta(q),$$

where $\delta(4) = 3$ and $\delta(5) = 2$, which completes the proof of the theorem. □

IV. SCALAR MULTIPLICATIONS

In this section, we introduce an efficient method to compute scalar multiplications using Theorem 1. We assume in this section that elements of finite fields can be represented in a normal basis so that evaluation of the Frobenius map at a point is just one bit circular shift of its vector representation over the base field. Therefore, without loss of generality, we can assume that evaluating the Frobenius map is free.

Let E be an elliptic curve defined over $\mathbb{F}_q$, $t = q + 1 - \#E(\mathbb{F}_q)$ and P a point of

$E(\mathbb{F}_{q^n})$. Suppose one wants to compute mP for an integer m.

The first step is to find the remainder $m' = x + y\alpha$ of division of m by $\alpha^n - 1$. To do this, we introduce a division algorithm in $\mathbb{Z}[\alpha]$ corresponding to Lemma 3. The following algorithm inputs the dividend $u+v\alpha$ and divisor $r+s\alpha$ and outputs a quotient $w+z\alpha$ and remainder $x+y\alpha$, Here $\lfloor a \rceil$ means the nearest integer from a.

Algorithm 1 (Division Algorithm)

Input $u+v\alpha$, $r+s\alpha$
1. Set $k \leftarrow ru+tsu+qsv$, $l \leftarrow rv-su$
2. Set $h \leftarrow r^2+trs+qs^2$
3. Set $w \leftarrow \lfloor k/h \rceil$, $z \leftarrow \lfloor l/h \rceil$
4. Set $x \leftarrow u-rw+qsz$, $y \leftarrow v-sw-rz-tsz$
5. Output w, z, x, y

To compute the remainder $x+y\alpha$ of division by α^n-1, one needs to express α^n-1 in the form $r+s\alpha$. This is done via a Lucas sequence, which is similar to the anomalous binary case in [16]. Let $U_0 = 0$, $U_1 = 1$ and

$$U_k = tU_{k-1} - qU_{k-2}$$

for $k \geq 2$. Then it is easy to prove that

$$\alpha^n = U_n\alpha - qU_{n-1}.$$

After we get the remainder $x+y\alpha$ by Algorithm 1, the second step is to find a Frobenius expansion of $x+y\alpha$. The following algorithm inputs $x+y\alpha$ and outputs a set $C =< c(k), c(k-1), \cdots, c(1), c(0) >$ of coefficients of a Frobenius expansion $\sum_{j=0}^{k} c_j\alpha^j$ of $x+y\alpha$. Then m has the Frobenius expansion $m = \sum_{j=0}^{k} c_j\phi^j$.

Algorithm 2 (Frobenius Expansions)

Input x, y, t
1. Set $C \leftarrow <>$
2. While $x \neq 0$ **or** $y \neq 0$
  2.1 Set $u \leftarrow (x \mod q)$, $v \leftarrow (x-u)/q$
  2.2 Set $(x, y) \leftarrow (tv+y, -v)$
  2.3 Prepend u **to** C
3. EndWhile
4. Output

In Step 2.1, we take $\{i \in \mathbb{Z} | -q/2 < i \leq q/2\}$ as the set of residues modulo q.

The third step is to compute mP using the Frobenius expansion of m. First, according to c_j, we rearrange and combine the terms of the expansion of m into $\lfloor q/2 \rfloor$ parts. Let $S_i := \{j | c_j = i, 0 \leq j \leq n+2\}$ and $S_i' := \{j | c_j = -i, 0 \leq j \leq n+2\}$ for $1 \leq i \leq \lfloor q/2 \rfloor$. Then we can write m as follows:

$$m = \sum_{j=0}^{n+2} c_j\phi^j \tag{15}$$

$$= (\sum_{j\in S_1} \phi^j - \sum_{j\in S_1'} \phi^j) + 2(\sum_{j\in S_2} \phi^j - \sum_{j\in S_2'} \phi^j) + \cdots + \lfloor q/2 \rfloor (\sum_{j\in S_{\lfloor q/2 \rfloor}} \phi^j - \sum_{j\in S_{\lfloor q/2 \rfloor}'} \phi^j) \tag{16}$$

$$= \sum_{i=1}^{\lfloor q/2 \rfloor} i(\sum_{j\in S_i} \phi^j - \sum_{j\in S_i'} \phi^j). \tag{17}$$

That is, mP is of the form $mP = \sum_{i=1}^{\lfloor q/2 \rfloor} iP_i$ with $P_i = \sum_{j\in S_i} \phi^j(P) - \sum_{j\in S_i'} \phi^j(P)$. Hence we can calculate mP by

$$P_{\lfloor q/2 \rfloor} + (P_{\lfloor q/2 \rfloor} + P_{\lfloor q/2 \rfloor - 1}) + \cdots + (P_{\lfloor q/2 \rfloor} + P_{\lfloor q/2 \rfloor - 1} + \cdots + P_2 + P_1)$$

Algorithm 3 (Scalar Multiplications)

Input $m = \sum_{j=0}^{k} c_j \phi^j$, P

1. Set $T \leftarrow O$

2. Set $Q \leftarrow O$

3. For $i = \lfloor q/2 \rfloor$ **to 1 by** -1

3.1 For each j **such that** $c_j = i$,
set $T \leftarrow T + \phi^j(P)$

3.2 For each j **such that** $c_j = -i$,
set $T \leftarrow T - \phi^j(P)$

3.3 Set $Q \leftarrow Q + T$

4. Output Q

Step 3.1 and Step 3.2 require k elliptic additions and Step 3 requires $\lfloor q/2 \rfloor - 1$ elliptic additions. Totally Algorithm 3 requires at most $\lfloor q/2 \rfloor + k - 1$ elliptic additions.

Remember an arbitrary element $s = x + y\alpha$ in $\mathbb{Z}[\alpha]$ is divisible by α if and only if $x \equiv 0 \mod q$. Hence with probability $1/q$ this element has a reduction of the form $s = q\alpha$, i.e., with $c = 0$. Hence we guess that in the expansion $m = \sum_{j=0}^{k} c_j \phi^j$ a q-th of the coefficients c_j are expected to be zero. Hence Algorithm 3 requires at average $\lfloor q/2 \rfloor - 1 + k(q-1)/q$ elliptic additions.

Now we compare the maximal numbers of elliptic additions in the proposed method to those of the binary method. We assume that an elliptic curve E is defined over $\mathbb{F}_q$ and consider a point P in $E(\mathbb{F}_{q^n})$. Let $l = \lceil \log_2 q^n \rceil$ be the smallest integer not less than $\log_2 q^n$. In Table 1, the second column gives the upper bound of the expansion length of any multiplier m when we determine the expansion of m by Theorem 1. The third and fourth column give the maximal numbers of elliptic additions required for scalar multiplications, when we apply the proposed method and the binary method, respectively.

Table 1. The comparison of the maximal complexity.

q	Exp. Leng.	Proposed	Binary
2^2	$l/2+3$	$l/2+3$	$2l$
2^3	$l/3+3$	$l/3+5$	$2l$
2^4	$l/4+2$	$l/4+8$	$2l$
2^5	$l/5+2$	$l/5+16$	$2l$
2^6	$l/6+2$	$l/6+32$	$2l$
2^7	$l/7+2$	$l/7+64$	$2l$

Table 1 shows the complexity of the proposed and the binary method for various defining fields which are suitable for elliptic curve cryptosystems. For example, if E is defined over $\mathbb{F}_{2^5}$ and consider the points in $E(\mathbb{F}_{(2^5)^{31}})$, the proposed method requires at most 47 elliptic additions for any integer m. On the other hand, the binary method requires at average 233 and at maximum 310 elliptic additions.

In Table 2, we see that the proposed method is almost four times (or more) faster than the binary method. Since Muller's method is two times faster than the binary method, we see that the proposed method is two times (or more) faster than Muller's. Moreover, the method does not require additional memory while Muller's requires much memory to store $q/2$ points.

Table 2. The comparison of examples.

q	n	$\lceil \log_2 q^n \rceil$	Maximal			Average		
			Prop.	Bin.	Bin./Prop.	Prop.	Bin.	Bin./Prop.
2^2	79	158	82	316	3.9	62	237	3.8
2^3	53	159	58	318	5.5	51	239	4.7
2^4	41	164	49	328	6.7	46	246	5.3
2^5	31	155	47	310	6.6	46	233	5.1
2^6	29	174	61	348	5.7	61	261	4.3
2^7	23	161	87	322	3.7	87	242	2.8
$2^2+1=5$	67	156	70	312	4.5	56.6	234	4.1
$2^3-1=7$	59	166	63	332	5.3	53.2	249	4.7
$2^4+1=17$	41	168	49	336	6.9	46.4	252	5.4
$2^5-1=31$	31	156	46	312	6.8	45.0	234	5.2
$2^6-3=61$	29	172	59	344	5.8	58.5	258	4.4

V. ELLIPTIC CURVES FOR PUBLIC KEY CRYPTOSYSTEMS

Note that in applying the proposed method, there is no restriction in taking elliptic curves except that the defining field is small. Furthermore we can also take an elliptic curve defined over a field with odd characteristic if we use a normal basis representation for elements of a field.

We present a table of elliptic curves defined over a small field whose orders over an extension field contain large primes.

The Table 2 shows elliptic curves whose orders are divided by a large prime and permit efficient multiplication by the Algorithms 1, 2 and 3. Note that we considered odd q of the form $2^n \pm$

a

eyHheuGkydhsuayKhKkymdduW&qkKGKydaklRiheuGK-hwuiiym-

Table 3. Elliptic curves suitable for cryptosystems.

q	n	t	Order of $E(\mathbb{F}_{q^n})$
2^4	47	7	$2\cdot 5\cdot 392318858461667547739736838942997715128064667934031507 29$
2^5	31	-3	$2^2\cdot 3^2\cdot 1268664615738631005385132475256827463554251723$
17	59	-7	$5^2\cdot 821\cdot 136811501951409793933337194553359822602756842 21$
31	31	-8	$2^3\cdot 5\cdot 1427\cdot 299039490727456831790657263017714497486693$
31	31	-7	$3\cdot 13\cdot 437671131557006050220780395909037989200200701$
31	31	-4	$2^2\cdot 3^2\cdot 373\cdot 127116280389657700019443582484912132361078 7$
31	31	5	$3\cdot 3\cdot 63219163447123096143003306616429635914864178 1$

the value of this polynomial at P. The proposed method is efficient because the Frobenius map is easy to evaluate. In particular, it is just one-bit circular shift of the vector representation of a point over the base field when a normal basis representation is used.

Using the proposed method (Algorithm 1, 2 and 3), we can multiply a point $P \in E(\mathbb{F}_{2^{4n}})$ by any integer m in at most (not at average) $n+8$ elliptic additions when the elliptic curve E is defined over $\mathbb{F}_{2^4}$. For the other curves with small defining fields, any multiplication of a point $P \in E(\mathbb{F}_{q^n})$ just requires elliptic additions for $l = \lceil \log_2 q^n \rceil$. Hence we can see that this method is four (or more) times faster than the binary method since the binary method requires $2l$ elliptic additions to multiply a point by any l-bit integer m. Consequently, the proposed method is two times faster than Muller's. Moreover, the proposed method can be applied for all non-supersingular elliptic curves with small defining field of even or odd characteristic, and does not require any precomputation and additional memory.

The proposed method is useful when one implements elliptic curve cryptosystems in small hardware such as a smart card because it provides high computational speed and requires small size of memory.

REFERENCES

[1] Jung Hee Cheon, Sungmo Park, Sangwoo Park, and Daeho Kim, "Two Efficient Algorithms for Arithmetic of Elliptic Curves Using Frobenius Map," *Proc. of PKC'98*, 1998, pp. 160–168.

[2] J. Guajardo and C. Paar, "Efficient Algorithms for Elliptic Curve Cryptosystems," *Proc. Crypto '97*, Springer-Verlag, 1997, pp. 342–356.

[3] K. Koyama and Y. Tsuruoka, "Speeding up Elliptic Cryptosystems by Using a Signed Binary Window Method," *Proc. Crypto'92*, Springer-Verlag, 1993, pp. 43–56.

[4] N. Koblitz, *A Course in Number Theory and Cryptography*, Springer-Verlag, 1991.

[5] N. Koblitz, "CM Curves with Good Cryptographic Properties," *Proc. Crypto '91*, Springer-Verlag, 1992, pp. 279–287.

[6] N. Koblitz, "Elliptic Curve Cryptosystems," *Math. of Comp.*, Vol. 49, 1987, pp. 203−209.

[7] W. Meier and O. Staffelbach, "Efficient Multiplication on Certain Non-Supersingular Elliptic Curves," *Proc. Crypto '92*, Springer-Verlag, 1993, pp. 333−344.

[8] A. Menezes, *Elliptic Curve Public Key Cryptosystems*, Kluwer Academic Publishers, 1993.

[9] F. Morain and J. Olivos, "Speeding up the Computations on an Elliptic Curve Using Additions-Subtraction Chains," *Inform. Theory. Appl.*, Vol. 24, 1990, pp. 531−543.

[10] V. Muller, "Fast Multiplication on Elliptic Curve over Small Fields of Characteristic Two," *J. of Cryptology*, Vol. 11, 1998, pp. 219−234.

[11] T. Satoh and K. Araki, "Fermat Quotient and the Polynomial Time Discrete Log Algorithm for Anomalous Elliptic Curves," *preprint*, 1997.

[12] R. Schoof, "Elliptic Curves over Finite Fields and the Computation of Square Roots mod p," *Math. Comp.*, Vol. 44, 1985, pp. 483−494.

[13] I. Semaev, "Evaluation of Discrete Logarithms in a Group of p-Torsion Points of an Elliptic Curve in Characteristic p," *Math. of Comp.*, Vol. 67, 1998, pp. 353−356.

[14] J. Silverman, *The Arithmetic of Elliptic Curves*, Springer-Verlag, 1992.

[15] N. Smart, "The Discrete Logarithm Problem on Elliptic Curves of Trace One," *preprint*, 1997.

[16] J. Solinas, "An Improved Algorithm for Arithmetic on a Family of Elliptic Curves," *Proc. Crypto '97*, Springer-Verlag, 1997, pp. 357−371.

Jung Hee Cheon received his B.S., M.S., and Ph.D. degrees in mathematics from Korea Advaced Institute of Science and Technology (KAIST) in 1991,1993, and 1997, respectively. He joined ETRI in 1997, and has been a senior member of engineering staff in Coding Technology Department. His research interests are elliptic curves, number theory and cryptography.

Sangjoon Park received the B.S. and M.S. degrees in Mathematics from Hanyang University, Korea, in 1984 and 1986. He joined ETRI in 1986 and has been with Coding Technology Department. His research interests are cryptography and information security.

Choonsik Park received the B.S. degree from Kwangwoon University and the M.S. degree from Hanyang University, Seoul, Korea in 1981 and 1983, respectively, and the Dr. Eng. degree in electronic engineering from Tokyo Institute of Technology, Tokyo, Japan in 1995. Since joining Coding Technology and Research Section of ETRI in 1982, he has been engaged in research and development on information security. His research interests are information security and cryptographic protocols.

Sang Geun Hahn received B.S. from Seoul Natinal Univiversity (1979), M.S. from New Mexico State University (1982) and Ph.D. from Ohio State University (1987) in mathematics. He has been a professor of mathematics in Korea Advaced Institute of Science and Technology (KAIST) since 1989. His research interests are number theory and its applications.