PAPER *Special Section on Cryptography and Information Security*

Fast Computation over Elliptic Curves $E(\mathbf{F}_{q^n})$ Based on Optimal Addition Sequences

Yukio TSURUOKA[†] ***and*** **Kenji KOYAMA**[†], ***Regular Members***

SUMMARY A fast method for computing a multiple mP for a point P on elliptic curves is proposed. This new method is based on optimal addition sequences and the Frobenius map. The new method can be effectively applied to elliptic curves $E(\mathbf{F}_{q^n})$, where q is a prime power of medium size (e.g., $q \leq 128$). When we compute mP over curves $E(\mathbf{F}_{q^n})$ with q^n of nearly 160-bits and $11 \leq q \leq 128$, the new method requires less elliptic curve additions than previously proposed methods. In this case, the average number of elliptic curve additions ranges from 40 to 50.

key words: *elliptic curves, scalar multiplications, Frobenius map, addition chains*

1. Introduction

The use of elliptic curves in cryptography was suggested independently by Miller [1] and Koblitz [2] in 1985. Since then, elliptic curve cryptosystems have gained much attention and have been discussed in IEEE P1363, ISO/IEC and ANSI because of their high security and efficiency. In elliptic curve cryptosystems, encryption/decryption involves multiplying a point P on an elliptic curve E by a large integer m. To speed up such scalar multiplications, there are many factors to consider: the choice of underlying field, type of curves [3], addition formulae [4], representation of points, scalar multiplication algorithms [5], and their combinations [6]–[8]. Among these choices, the use of an elliptic curve over an extension field $\mathbf{F}_{q^n}$ denoted $E(\mathbf{F}_{q^n})$ is particularly attractive. This is because, by using the Frobenius map, scalar multiplications can be performed much faster on $E(\mathbf{F}_{q^n})$ compared with curves over a prime field. In 1991, Koblitz [9] proposed the use of curves over $\mathbf{F}_{2^n}$, $\mathbf{F}_{4^n}$, $\mathbf{F}_{8^n}$ and $\mathbf{F}_{16^n}$. Since then, several extensions have been proposed [10]–[13]. The extension [11] in which uses an NAF representation, the curve over $\mathbf{F}_{2^n}$ requires about 1/3 the number of elliptic curve additions compared with curves over prime fields. However, there are only a few curves available over $\mathbf{F}_{2^n}$ for cryptographic use, and using restricted curves may not be appropriate for security. For this reason, we focus our attention on curves over $\mathbf{F}_{q^n}$ with $q > 2$. Curves over $\mathbf{F}_{q^n}$ with $2 < q \leq 32$ have been studied in [12] and [13]. In these schemes, the required number of elliptic curve additions increases when q becomes large (e.g., $q > 16$).

In this paper, we also study scalar multiplications on curves over $\mathbf{F}_{q^n}$ with $q > 2$. Our method based on optimal addition sequences requires less elliptic curve additions than previously proposed methods [12], [13] for $q > 16$. An optimal addition sequence can be found using a search with newly proposed heuristics. When $q \leq 128$, the time required for the search is negligibly small compared with that for a scalar multiplication.

The organization of this paper is as follows. In Sect. 2, we give a brief review of the Frobenius map and scalar multiplications over elliptic curves. In Sect. 3, we show a new method of finding an optimal addition sequence that minimizes the number of required elliptic curve additions. In Sect. 4, comparison of the new method with previously proposed methods is shown. In Sect. 5, we conclude the discussion.

2. Scalar Multiplications

In Sect. 2.1, we briefly review elliptic curves and the Frobenius map. In elliptic curve cryptosystems, there are two types of scalar multiplications: one is to compute mP with a constant m and variable P, and the other is to compute mP with a variable m and constant P. We show how to compute each type of scalar multiplication with the Frobenius map in Sects. 2.2 and 2.3, respectively.

2.1 Frobenius Expansions

Let $\mathbf{F}_q$ be the field with q elements where q is a prime power. Let E be an elliptic curve

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \quad (a_i \in \mathbf{F}_q).$$

We denote the group of $\mathbf{F}_q$-rational points on E together with a point at infinity $\mathcal{O}$ as $E(\mathbf{F}_q)$. Let $\mathbf{F}_{q^n}$ be an extension field of $\mathbf{F}_q$. The group of $\mathbf{F}_{q^n}$-rational points on E is denoted by $E(\mathbf{F}_{q^n})$. Note that the order of $E(\mathbf{F}_{q^n})$ can be easily computed using the Weil conjecture [9]. The Frobenius map $\phi : E(\mathbf{F}_{q^n}) \to E(\mathbf{F}_{q^n})$ is defined as $(x, y) \mapsto (x^q, y^q)$. For all $P \in E(\mathbf{F}_{q^n})$, the Frobenius map ϕ satisfies

$$\phi^2(P) - t\phi(P) + qP = \mathcal{O} \tag{1}$$

where t is the trace of ϕ. In (1), by replacing $\phi()$

Manuscript received March 24, 2000.
Manuscript revised July 7, 2000.

[†] The authors are with NTT Communication Science Laboratories, Kyoto-fu, 619-0237 Japan.

with T, we create an equation $T^2 - tT + q = 0$, and $\alpha = (t + \sqrt{t^2 - 4q})/2$ is one of its roots. There is a natural homomorphism from the ring $\mathbf{Z}[\alpha]$ to the endomorphism ring of $E(\mathbf{F}_{q^n})$ that maps α to ϕ. This means the following. For a given integer m, if we have an expansion $m = \sum m_i \alpha^i$ in $\mathbf{Z}[\alpha]$, we immediately obtain a corresponding expansion of $mP = \sum m_i \cdot \phi^i(P)$ for $P \in E(\mathbf{F}_{q^n})$ and m. In addition, we can replace a multiple mP with $m'P$ where $m' = m \bmod (\alpha^n - 1)$ since $\alpha^n \cdot P = \phi^n(P) = P$. Moreover, assuming that $P \in E(\mathbf{F}_{q^n}) \setminus E(\mathbf{F}_q)$, we have $\alpha^n - 1 = 0$ and $\alpha - 1 \neq 0$. Thus, $(\alpha^n - 1)/(\alpha - 1) = \alpha^{n-1} + \alpha^{n-2} + \cdots + \alpha + 1 = 0$. Accordingly, $mP = m''P$, where $m'' = m \bmod (\alpha^{n-1} + \alpha^{n-2} + \cdots + \alpha + 1)$.

The way of computing the Frobenius expansion is as follows. First, we compute m'' for a given m as shown above. Then, we obtain an α-ary expansion for m'':

$$m'' = \sum_{i=1}^{s} m_i \alpha^{i-1}, \quad m_i \in \mathbf{Z} \tag{2}$$

with the following algorithm.

```
Algorithm FROBENIUS-EXP
Input: $u$, $v$, $q$, $t$                    /* $m'' = u + v\alpha$ */
Output: $m_1, \cdots, m_s$          /* $\alpha$-ary expansion of $m''$ */
$i := 1$;
while $u \neq 0$ or $v \neq 0$ do
    $m_i := u \bmod q$ where $-q/2 \leq m_i < q/2$;
    $w := (u - m_i)/q$;
    $u := v + wt$;
    $v := -w$;
    $i := i + 1$
endw;
return $m_0, m_1, \cdots, m_s$; end
```

In this algorithm, the expansion length s is bounded by $n + 3$ [12].

2.2 Compute mP with a Variable P

First, we show a conventional method for computing mP. Then, we briefly show how to reduce the number of required elliptic curve additions. The details of the algorithm will be shown in Sect. 3.

Using α-ary expansion $\{m_i\}$ given by (2), we compute

$$\begin{aligned} mP = {} & m_1 P + \phi(m_2 P + \cdots \\ & + \phi(m_{s-1} P + \phi(m_s P)) \cdots). \end{aligned} \tag{3}$$

When each coordinate of P is represented using a normal basis [14], $\phi(P)$ can be immediately obtained with a rotation of the basis for each P coordinate. As a result, we ignore the cost for $\phi(\cdot)$. Thus, mP in (3) can be computed in $s - 1$ additions from the set of points $S = \{m_1 P,\ m_2 P,\ \cdots,\ m_s P\}$. All elements of S can be computed in $m_{max} - 1$ additions by using an ascending chain

$$P, 2P, 3P, \cdots, m_{max} P \tag{4}$$

where $m_{max} = \max\{m_i \mid 1 \leq i \leq s\}$. In this conventional method, an upper bound of the number of required elliptic curve additions for scalar multiplications is

$$m_{max} + s - 2 \qquad \left(\leq \frac{q}{2} + s - 2\right).$$

Note that methods used in [12] and [13] are essentially equivalent to this conventional method.

In some cases, there may be a better way of computing all points in S than (4). A way of computing the set of points $S = \{m_1 P,\ m_2 P,\ \cdots,\ m_s P\}$ from P is formalized as follows.

Definition 1 (addition sequence): An addition sequence for a set of integers $\{m_1, m_2, \ldots, m_s\}$ is a sequence of integers

$$1 = a_0, a_1, a_2, \cdots, a_r,$$

where $a_i = a_j + a_k$ for some j, k $(i > j \geq k)$ for all $i = 1, 2, \ldots, r$, and $m_\ell = a_i$ for some i $(1 \leq i \leq r)$ for all $\ell = 1, 2, \cdots, s$. The length of the addition sequence, defined as r, is equal to the number of required additions.

To compute all points in S in the least number of additions, we must find the shortest addition sequence for a set of digits $\{m_1, m_2, \cdots, m_s\}$. Section 3 will describe how we find an optimal (or the shortest) addition sequence.

2.3 Compute mP with a Constant P

Using expansion (2), mP can be expressed as a linear combination

$$m_1 P_1 + m_2 P_2 + \cdots + m_s P_s \tag{5}$$

with $P_i = \alpha^{i-1} \cdot P = \phi^{i-1}(P)$. Since P is constant, each P_i can be computed in advance. In this case, normal basis representation is not required for coordinates of P because there is no need to compute ϕ quickly. A sum of multiples (5) can be represented by a vector $[m_1, m_2, \cdots, m_s]$, and points $P_1,\ P_2,\ \cdots,\ P_s$ are represented by the basis vectors $[1, 0, \cdots, 0]$, $[0, 1, 0, \cdots, 0]$, $\cdots$, $[0, \cdots, 0, 1]$, respectively. A way of computing (5) from $P_1,\ P_2,\ \cdots,\ P_s$ is formalized as follows.

Definition 2 (vector addition chain): A vector addition chain for a vector $v_r = [m_1, m_2, \ldots, m_s]$ is a sequence of vectors $v_{-s+1}, v_{-s+2}, \ldots, v_0, v_1, v_2, \cdots, v_r$, where $v_{-s+1} = [1, 0, 0, \ldots, 0]$, $v_{-s+2} = [0, 1, 0, \ldots, 0], \ldots$, $v_0 = [0, 0, 0, \ldots, 1]$, and $v_i = v_j + v_k$ for some j, k $(i > j \geq k)$ for $i = 1, 2, \ldots, r$. The length of the vector addition chain, defined as r, is equal to the number of required additions.

There is a one-to-one correspondence between vector addition chains and addition sequences.

Fact 1 ([15]): For any addition sequence of length L for $\{m_1, m_2, \cdots, m_s\}$, there exists a corresponding vector addition chain of length $L + s - 1$ for $[m_1, m_2, \cdots, m_s]$, and vice versa.

From Fact 1, the problem of computing $m_1P_1 + m_2P_2 + \cdots + m_sP_s$ from $P_1, P_2, \cdots, P_s$ is essentially equivalent to the problem of finding an addition sequence for $\{m_1, m_2, \cdots, m_s\}$. We can easily construct a vector addition chain from a corresponding addition sequence.

3. Searching for an Optimal Addition Sequence

In general, finding an optimal addition sequence is an NP-hard problem [16]. In fact, an exhaustive search for finding optimal addition sequences takes a very long time. However, we can efficiently solve the instances we are interested in (e.g., $q < 512$) by using the search algorithm described in this section. This is because we employ two useful propositions to prune irrelevant branches in the search tree, and the size of the instances is small enough to be handled. Note that, for elliptic curves over the prime field $\mathbf{F}_p$, finding an optimal addition chain for a full 160-bit multiplier is still intractable. In this sense, $E(\mathbf{F}_{q^n})$ is suitable for elliptic curve cryptosystems compared with $E(\mathbf{F}_p)$.

For a given set of integers $\{m_1, m_2, \cdots, m_s\}$, there exists an optimal addition sequence $a_0, a_1, \cdots, a_\ell$ such that $a_i < a_j$ for $i < j$. Hereafter, we assume without a loss of generality that the optimal addition sequence is ascending. We denote an ascending sequence of integers $a_0 < a_1 < \cdots < a_\ell$ by $(a_0, a_1, \cdots, a_\ell)$. For $M = \{m_1, m_2, \cdots, m_s\}$, we also assume $m_i < m_j$ for $i < j$, and write set M as $(m_1, m_2, \cdots, m_s)$ instead of $\{m_1, m_2, \cdots, m_s\}$. An optimal addition sequence for M can be found by searching over all possible sequences. The following two propositions can be used to check whether or not a given sequence $(a_0, \cdots, a_i)$ can be a prefix of some optimal addition sequence for M.

Proposition 1 (length estimation): If an addition sequence $(a_0, a_1, \ldots, a_\ell)$ for $(m_1, m_2, \cdots, m_s)$ is optimal, then the following holds for $i = 0, 1 \ldots, \ell$.

$$\sum_{j=1}^{s} d(m_j, \max\{a_i, m_{j-1}\}) \leq \ell - i$$

where

$$d(x, y) = \begin{cases} \lceil \log_2(x/y) \rceil & (x > y) \\ 0 & (x \leq y) \end{cases}$$

and $m_0 = 1$.

In order to prove Proposition 1, we start by proving the following lemma.

Lemma 1: For an ascending addition sequence $(a_0, a_1, \ldots, a_\ell)$, $d(a_{i'}, a_i) \leq i' - i$ holds for i, i'; $(0 \leq i < i' \leq \ell)$.

Proof: Let i be any integer that satisfies $0 \leq i \leq \ell - 1$. Since $a_0, a_1, \cdots, a_{i+1}$ is an addition sequence, there exist a_j and a_k such that $a_{i+1} = a_j + a_k$ $(i \geq j$, $i \geq k$, $a_j \geq a_k)$. Since $i \geq j$ and $(a_0, a_1, \cdots, a_{i+1})$ is ascending, $a_i \geq a_j$. It follows that $a_{i+1} = a_j + a_k \leq 2a_j \leq 2a_i$. By using the relation $a_{i+1} \leq 2a_i$ repeatedly, $a_{i'} \leq 2^{i'-i}a_i$ for $i < i'$. Since $i' - i$ is an integer, we have $i' - i \geq \lceil \log_2(a_{i'}/a_i) \rceil = d(a_{i'}, a_i)$. □

Lemma 1 means that if a_i and $a_{i'}$ are elements in $(a_0, a_1, \ldots, a_\ell)$, there are at least $d(a_{i'}, a_i) - 1$ other elements between a_i and $a_{i'}$.

Proof of Proposition 1: For all m_j; $(1 \leq j \leq s)$, there exists $a_{\pi(j)}$ such that $a_{\pi(j)} = m_j$. In particular, $m_s = a_{\pi(s)} = a_\ell$.

First, we show the case for $i = \ell$. Since $a_i = a_\ell = m_s$, $d(m_j, \max\{a_i, m_{j-1}\}) = d(m_j, a_\ell) = 0$ for all j; $(1 \leq j \leq s)$. $\sum_{j=1}^{s} d(m_j, \max\{a_i, m_{j-1}\}) = 0 = \ell - i$. Thus, Proposition 1 holds for $i = \ell$.

Second, for the case of $i < \ell$, there exists an integer k such that $m_{k-1} \leq a_i < m_k$. As a result,

$$d(m_j, \max\{a_i, m_{j-1}\}) = \begin{cases} 0 & (1 \leq j \leq k-1) \\ d(m_k, a_i) & (j = k) \\ d(m_j, m_{j-1}) & (k+1 \leq j \leq s). \end{cases}$$

By using this relation and lemma 1, we have $\ell - i = \pi(s) - i = (\pi(k) - i) + \sum_{j=k+1}^{s}(\pi(j) - \pi(j-1)) \geq d(a_{\pi(k)}, a_i) + \sum_{j=k+1}^{s} d(a_{\pi(j)}, a_{\pi(j-1)}) = d(m_k, a_i) + \sum_{j=k+1}^{s} d(m_j, m_{j-1}) = \sum_{j=1}^{s} d(m_j, \max\{a_i, m_{j-1}\})$. □

Example 1: Suppose that we want to find an optimal addition sequence for $M = (4, 9, 19)$. We assume the length ℓ of optimal addition sequences is 6. Consider an ascending sequence $\gamma = (1, 2, 3)$. If γ is a prefix of some optimal addition sequence, Proposition 1 must hold for $i = 2$. However, $\sum_{j=1}^{s} d(m_j, \max\{a_i, m_{j-1}\}) = d(4, 3) + d(9, 4) + d(19, 9) = 1 + 2 + 2 = 5$ and $\ell - i = 6 - 2 = 4$, so Proposition 1 does not hold. Thus, γ can not be a prefix of any optimal addition sequence for M. Therefore, we do not need to check all sequences that contain 3.

We define *extended addition sequences* as follows. An addition sequence $(a_0, a_1, \cdots, a_\ell)$ can be extended to $(\langle a_0, b_0, c_0 \rangle, \langle a_1, b_1, c_1 \rangle, \cdots, \langle a_\ell, b_\ell, c_\ell \rangle)$ where $a_i = b_i + c_i$, $b_i \geq c_i$, $\{b_i, c_i\} \subseteq \cup_{j=0}^{i-1}\{a_j\}$ for i; $(1 \leq i \leq \ell)$, and $a_0 = b_0 = 1$, $c_0 = 0$. Note that this extension is not unique. For example, $(1, 2, 3, 4)$ can be extended to $(\langle 1, 1, 0 \rangle, \langle 2, 1, 1 \rangle, \langle 3, 2, 1 \rangle, \langle 4, 3, 1 \rangle)$ or

$(\langle 1,1,0\rangle, \langle 2,1,1\rangle, \langle 3,2,1\rangle, \langle 4,2,2\rangle)$.

Proposition 2 (reference check): If an extended addition sequence $\gamma = (\langle a_0,b_0,c_0\rangle,\ \langle a_1,b_1,c_1\rangle,\ \cdots,\ \langle a_\ell,b_\ell,c_\ell\rangle)$ for $M = (m_1,m_2,\cdots,m_s)$ is optimal, then

$$\#|I(i)| \le \ell - i + \#|M \setminus A(i)|$$

for all i; $(0 \le i \le \ell)$ where $I(i) = A(i) \setminus \{B(i) \cup M\}$, $A(i) = \cup_{j=0}^{i}\{a_j\}$, $B(i) = \cup_{j=1}^{i}\{b_j,c_j\}$ and $\#|\cdot|$ represents the number of elements of a set. Note that $I(i)$ is a subset of un-referred elements among $(a_0,a_1,\cdots,a_i) \setminus (m_1,\cdots,m_s)$.

Proof: First, we prove the case for $i = \ell$. Suppose to the contrary that $I(\ell) \ne \{\}$, then there exists $a_j \in I(\ell)$. Since a_j is an unreferred element, a sequence $(a_0,\cdots,a_{j-1},a_{j+1},\cdots,a_\ell)$ (i.e., the same sequence as $(a_0,\cdots,a_\ell)$ except for a_j) is also an addition sequence for M of length $\ell - 1$. This contradicts the claim that γ is optimal. Therefore, $I(\ell) = \{\}$.

Second, by the definition of $I(i)$, we have $I(i) = A(i) \setminus \{B(i) \cup M\} = \{A(i-1) \cup \{a_i\}\} \setminus \{B(i-1) \cup \{b_i,c_i\} \cup M\} = (I(i-1) \cup (\{a_i\} \setminus M)) \setminus \{b_i,c_i\}$. So $\#|I(i)| \ge \#|I(i-1)| + \#|\{a_i\} \setminus M| - 2$. It follows that $\#|I(i-1)| \le \#|I(i)| + 1 + \delta(a_i,M)$ where

$$\delta(a_i,M) = \begin{cases} 1 & (a_i \in M) \\ 0 & (a_i \notin M). \end{cases}$$

By induction on i, we have $\#|I(i)| \le (\ell - i) + \#|M \setminus A(i)|$. []

Example 2: Suppose that we want to find an optimal addition sequence for $M = (127)$. We assume the length ℓ of optimal addition sequences is 10. Consider an extended addition sequence $\gamma = (\langle 1,1,0\rangle,\ \langle 2,1,1\rangle,\ \langle 4,2,2\rangle,\ \langle 8,4,4\rangle,\ \langle 12,8,4\rangle,\ \langle 16,8,8\rangle,\ \langle 18,16,2\rangle,\ \langle 24,16,8\rangle,\ \langle 32,16,16\rangle)$. If γ is a prefix of some optimal addition sequence, Proposition 2 must hold for $i = 8$. However, $\#|I(8)| = \#|\{12,18,24,32\}| = 4$, and $\ell - i + \#|M \setminus A(i)| = 10 - 8 + \#|\{127\} \setminus \{1,2,4,8,12,16,18,24,32\}| = 3$, so Proposition 2 does not hold. Thus, γ can not be a prefix of any optimal addition sequence for M. Therefore, we do not need to check all sequences starting with γ.

In order to use Propositions 1 and 2, we have to know the length ℓ of optimal addition sequences *a priori*. However, finding ℓ is as hard as finding an optimal addition sequence itself. Therefore, we use an estimated upper bound $\hat{\ell}$ instead of ℓ. If $\hat{\ell} < \ell$, which is a wrong estimation, the search algorithm stops and outputs "no solution"; otherwise, the search algorithm outputs an addition sequence, and the above propositions work to prune irrelevant branches in the search tree. Let $i = 0$ in Proposition 1, then we have a lower bound ℓ_0 for ℓ as

$$\ell_0 = \sum_{j=1}^{s} d(m_j, m_{j-1}) \le \ell.$$

We use ℓ_0 as an initial value for $\hat{\ell}$. If the following algorithm returns "no solution" with $\hat{\ell} = \ell_0$, then we increment $\hat{\ell}$ and try again.

We define some notations to describe the algorithm. Suppose that we want to find an optimal addition sequence for $M = (m_1,\cdots,m_s)$. Let $\gamma_i = (\langle a_0,b_0,c_0\rangle,\cdots,\langle a_i,b_i,c_i\rangle)$ be a prefix of some addition sequence for M. We denote a pair of γ_i and N by $[\![\gamma_i, N]\!]$, where $N = (m_j,m_{j+1},\cdots,m_s)$ is a sequence of integers to be obtained (i.e., $m_{j-1} \le a_i < m_j$). To search an optimal addition sequence for M, we start from $[\![(\langle 1,1,0\rangle), M]\!]$ and end with $[\![\gamma_\ell, ()]\!]$, where $\gamma_\ell = (\langle 1,1,0\rangle,\cdots,\langle a_\ell,b_\ell,c_\ell\rangle)$ is an optimal addition sequence for M.

Algorithm OPT-ASEQ
Input: $M = (m_1,m_2,\cdots,m_s)$,
$\hat{\ell}$ /* an upper bound of ℓ */
Output: an optimal addition sequence γ for M

```
$S := \{[\![(\langle 1,1,0\rangle), M]\!]\}$; $i := 1$;
while $S \ne \{\}$ do
  $U := \{\}$;
  for all pairs $[\![\gamma, N]\!] \in S$ do
    if $N = ()$ then return $\gamma$ endif;
    let $(\langle a_0,b_0,c_0\rangle,\cdots,\langle a_{i-1},b_{i-1},c_{i-1}\rangle) = \gamma$;
    let $(m_j,m_{j+1},\ldots,m_s) = N$;
    for all possible triples $\langle a_i,b_i,c_i\rangle$ where $a_i = b_i + c_i$,
    $\{b_i,c_i\} \subseteq \cup_{j=0}^{i-1}\{a_j\}$, $a_{i-1} < a_i \le m_j$ do
      $\gamma' := \gamma \cup \{\langle a_i,b_i,c_i\rangle\}$
        $= (\langle a_0,b_0,c_0\rangle,\cdots,\langle a_{i-1},b_{i-1},c_{i-1}\rangle,\langle a_i,b_i,c_i\rangle)$;
      if $\gamma'$ satisfies Propositions 1 and 2 with $\hat{\ell}$ and $i$
        then $N' := N \setminus \{a_i\}$; $U := U \cup \{[\![\gamma', N']\!]\}$
      endif
    endf
  endf;
  $S := U$; $i := i + 1$
endw
return "no solution"; end.
```

With an optimal addition sequence, we can compute scalar multiplications mP in less elliptic curve additions.

Example 3: Suppose that $q = 128$ and the Frobenius expansion of a multiplier m is $M = \{-58,\ 24,\ 19,\ -45,\ -40,\ 61,\ -56,\ -64,\ -29,\ 21,\ 38,\ -34,\ -29,\ 48,\ 37,\ 42,\ -21,\ 3,\ 38,\ 37,\ 24,\ -51,\ 8,\ -58,\ -29\}$. We ignore the sign of each digit $m_i \in M$ because $-m_iP = m_i(-P)$ and $-P$ can be obtained quickly from P. We also ignore the duplication of elements in M. As a result, we have sorted elements as $M' = \{3,8,19,21,24,29,34,37,38,40,42,45,48,51,56,58,61,64\}$.

In the proposed method, algorithm OPT-ASEQ finds an optimal addition sequence (1, 2, 3, 5, 8, 16, 19, 21, 24, 29, 34, 37, 38, 40, 42, 45, 48, 51, 56, 58, 61, 64) of length 21 for M'. To compute a scalar multiplication mP, the total number of elliptic curve additions is $\ell + \#|M| - 1 = 21 + 25 - 1 = 45$.

However, in the conventional algorithm, which is described in (4) of Sect. 2.2., uses an addition sequence (1, 2, 3, 4, $\cdots$, 63, 64) of length 63. Thus, the total number of elliptic curve additions is $63 + 25 - 1 = 87$. Therefore, by using the proposed method, we can reduce 48% of the elliptic curve additions.

4. Comparison

We compare the number of elliptic curve additions required for scalar multiplications for both the new method and the method in [12] (Müller method). The method in [13] is essentially equivalent to the Müller method. The new method uses the same Frobenius expansions as the Müller method, but the difference between it and the Müller method is in the addition sequence (i.e., how to compute all points in S). The new method uses an optimal addition sequence, while the Müller method uses the conventional addition sequence described in (4) of 2.2.

In scalar multiplications mP on $E(\mathbf{F}_{q^n})$, we assume that the multiplier m is a random integer up to the order of the subgroup generated by P. We choose q^n to be approximately 160-bits for security. Figure 1 shows the results of the simulation. The x-axis indicates the subfield size q, and the y-axis indicates the average number of elliptic curve additions required for mP. From Fig. 1, we can observe that the new method requires less additions than the Müller method for $q > 16$. For $3 \leq q \leq 16$, both methods require the same number of additions. The optimal subfield minimizing the number of additions is F_{19} for $q^n \approx 2^{160}$.

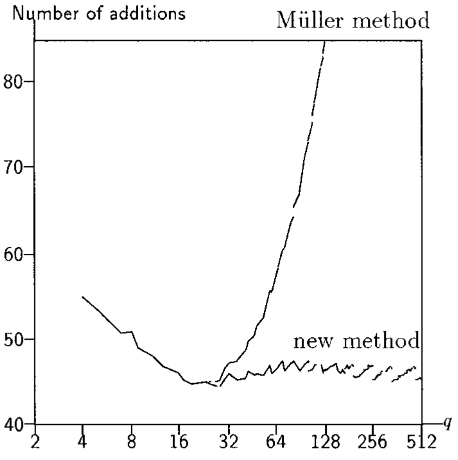


Fig. 1 Number of additions vs. q.

Solinas [11] and Kobayashi et al. [7] proposed methods for $E(\mathbf{F}_{2^n})$ and $E(\mathbf{F}_{q^n})$ for large q, respectively. The Solinas method uses the NAF representation to reduce the number of elliptic curve additions. However, the NAF representation seems not applicable to $E(\mathbf{F}_{q^n})$ with $q > 2$. In [7], two conventional algorithms are used. If we use these algorithms with $E(\mathbf{F}_{q^n})$ of medium sized q (e.g., $q \leq 128$), these require more additions than the new method. As a result, when we compute mP over curves $E(\mathbf{F}_{q^n})$ with q^n of nearly 160-bits, $11 \leq q \leq 128$ and $m \approx q^n$, the new method requires less elliptic curve additions than previously proposed methods. In this case, the average number of elliptic curve additions ranges from 40 to 50.

For both OPT-ASEQ and a conventional exhaustive search method, the times required for finding an optimal addition sequence are listed, for various subfield sizes, in Table 1. Because of combinatorial explosions, the conventional method requires much time for $q \geq 32$, while OPT-ASEQ requires far less time. When $q \leq 128$, the time OPT-ASEQ needs for finding an optimal addition sequence is negligible compared with the time needed for scalar multiplications, however, when q is large (e.g., $q > 512$), OPT-ASEQ also requires much time for finding an optimal addition sequence. We therefore recommend that a medium size q be used (e.g., $2 < q \leq 128$).

We compare the proposed method and a naive method for computing mP with constant P. Using formula (5) we compute mP as $m_1P_1 + \cdots + m_sP_s$ where $P_i = \alpha^{i-1}P$, m_i $(1 \leq i \leq s)$ is a Frobenius expansion of m. We assume the same m_i as in Example 3 in Sect. 3. In the naive method we compute each m_iP_i separately, which requires 111 elliptic curve additions, and then add all m_iP_i to obtain mP. The total number of elliptic curve additions is thus $111 + 25 - 1 = 135$. Using the proposed method, we first obtain an optimal addition sequence for $\{m_i\}$, and then convert it to the corresponding optimal vector addition chain. Since this conversion is performed quickly, the time required for finding optimal vector addition chain is the same as that for finding optimal addition sequence (in this example, 120 μsec). The length of the optimal vector addition chain, which is the number of required elliptic curve additions, is $21+25-1 = 45$ by Fact 1. There-

Table 1 Time needed for finding an optimal addition sequence.

q (subfield size)	16	24	32	48	64	128
Conventinal method	18 μsec	362 μsec	14.7 msec	10.87 sec	1287 sec	–
OPT-ASEQ	8 μsec	17 μsec	27 μsec	49 μsec	69 μsec	141 μsec

CPU: 500 MHz Celeron processor

fore, when using the proposed method, we need only one third as many elliptic curve additions as we need when using the naive method. When the time required for one elliptic curve addition is greater than $1.33\,\mu$sec ($120\,\mu$sec$/(135-45)$), we can speed up the computation mP for a constant P by using the proposed method.

5. Conclusion

We have proposed a fast method for computing a multiple mP for a point P on elliptic curves $E(\mathbf{F}_{q^n})$. The new method is based on optimal addition sequences and the Frobenius map. The new method can be effectively applied to elliptic curves $E(\mathbf{F}_{q^n})$ where q is a prime power of medium size (e.g., $q \leq 128$). When we compute mP over curves $E(\mathbf{F}_{q^n})$ with q^n of nearly 160-bits, $11 \leq q \leq 128$ and $m \approx q^n$, the new method requires less elliptic curve additions than previously proposed methods. In this case, the average number of elliptic curve additions ranges from 40 to 50.

The security of elliptic curve cryptosystems depends on the order of the largest subgroup of $E(\mathbf{F}_{q^n})$. Since the order of $E(\mathbf{F}_{q^n})$ is divisible by that of $E(\mathbf{F}_q)$, the security margin of $E(\mathbf{F}_{q^n})$ may be reduced by about $\log q$-bits. However, this amount is relatively small compared with the gain in speed when q is of a medium size ($q \leq 128$).

A benefit of constructing curves $E(\mathbf{F}_{q^n})$ with a medium sized q is the increase in the number of curve candidates while having a high security margin.

Acknowledgement

The authors would like to thank the anonymous referees for their helpful comments.

References

[1] V.S. Miller, "Use of elliptic curves in cryptography," Proc. CRYPTO'85, LNCS218, pp.417–426, Springer-Verlag, 1986.

[2] N. Koblitz, "Elliptic curve cryptosystems," Math. Comp., vol.48, pp.203–209, 1997.

[3] K. Okeya, H. Kurumatani, and K. Sakurai, "Elliptic curves with the Montgomery-form and their cryptographic applications," Advances in Cryptology–Proc. PKC'2000, LNCS 1751, pp.238–257, Springer-Verlag, 2000.

[4] E.W. Knudsen, "Elliptic scalar multiplication using point halving," Advances in Cryptology–Proc. ASIACRYPT'99, LNCS 1716, pp.135–149, Springer-Verlag, 1999.

[5] N. Kunihiro and H. Yamamoto, "Window and extended window methods for addition chain and addition-subtraction chain," IEICE Trans. Fundamentals, vol.E81-A, no.1, pp.72–81, Jan. 1998.

[6] A. Miyaji, T. Ono, and H. Cohen, "Efficient elliptic curve exponentiation," Advances in Cryptology–Proc. ICICS'97, LNCS 1334, pp.282–290, Springer-Verlag, 1997.

[7] T. Kobayashi, H. Morita, K. Kobayashi, and F. Hoshino, "Fast elliptic curve algorithm combining frobenius map and table reference to adapt to higher characteristic," Advances in Cryptology–Proc. EUROCRYPT'99, LNCS 1592, pp.176–189, Springer-Verlag, 1999.

[8] C.H. Lim and H.S. Hwang, "Fast implementation of elliptic curve arithmetic in GF(p^n)," Advances in Cryptology–Proc. PKC'2000, LNCS 1751, pp.405–421, Springer-Verlag, 2000.

[9] N. Koblitz, "CM-curve with good cryptographic properties," Proc. CRYPTO'91, pp.279–287, Springer-Verlag, 1992.

[10] W. Meier and O. Staffelbach, "Efficient multiplication on certain nonsupersingular elliptic curves," Proc. CRYPTO'92, pp.333–344, Springer-Verlag, 1993.

[11] J.A. Solinas, "An improved algorithm for arithmetic on a family of elliptic curves," Proc. CRYPTO'97, pp.357–371, Springer-Verlag, 1997.

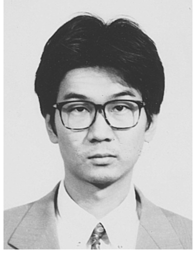
[12] V. Müller, "Fast multiplication on elliptic curve over small fields of characteristic two," J. Cryptology, vol.11, no.4, pp.219–234, 1998.

[13] J.H. Cheon, S. Park, C. Park, and S.G. Hahn, "Scalar multiplications of elliptic curves by frobenius expansions," ETRI J., vol.21, no.1, March 1999.

[14] A.J. Menezes, I.F. Blake, X.H. Gao, R.C. Mullin, S.A. Vanstone, and T. Yaghoobian, Applications of finite field, Kluwer Academic Pub., 1993.

[15] D.E. Knuth, Seminumerical algorithm (arithmetic), The Art of Computer Programming vol.2, Addison Wesley, 1969.

[16] P. Downey, B. Leong, and R. Sethi, "Computing sequences with addition chains," SIAM J. Comput., vol.3, pp.638–696, 1981.



Yukio Tsuruoka was born in Chiba Prefecture, Japan on November 25, 1962. He received the B.E. and M.E. degrees from University of Electro-Communications in 1985 and 1987, respectively. He joined Basic Research Laboratories, Nippon Telegraph and Telephone Corporation (NTT), in 1987. He is presently a senior research scientist in NTT Communication Science Laboratories. His current research interests include cryptography and information security. He is a member of the International Association for Cryptologic Research and the Information Processing Society of Japan.

The Late Kenji Koyama was born in Hyogo, Japan on June 24, 1949. He received the B.E. and M.E. degrees in electrical engineering, and Ph.D. degree in applied mathematics from Kyoto University, Kyoto, Japan, in 1972, 1974, and 1983 respectively. In 1974, He joined the Musashino Electrical Communication Laboratories, Nippon Telegraph and Telephone Corporation (present NTT), Tokyo, Japan. From 1985 to 1986, he was a visiting associate Professor at the University of Waterloo, Canada. He received the Yonezawa Medal Award for the paper "A Master Key for the RSA Public-key Cryptosystem" and the Excellent Book Award for "Modern Cryptography Theory" from the Institute of Electronics and Communication Engineers (present IEICE) of Japan in 1982 and 1988, respectively. He received the Research Achievement Award from the Science and Technology Agency of Japanese Government in 1984. Dr. Koyama passed away on March 27, 2000.