SCIS 2007 The 2007 Symposium on Cryptography and Information Security
Sasebo, Japan, Jan. 23-26, 2007
The Institute of Electronics, Information and Communication Engineers

Remarks on Cheon's algorithms for pairing-related problems

Taketeru Kutsuma* Kazuto Matsuo*†

Abstract— In EUROCRYPT 2006, Cheon proposed breakthrough algorithms for pairing-related problems such as the q-weak/strong Diffie-Hellman problem. Using that the exponents of an element in an abelian group G of prime order p form the ring $\mathbb{Z}/p\mathbb{Z}$ structure even if G is a generic group, Cheon's algorithms reduce their complexity by Polig-Hellman like method over $(\mathbb{Z}/p\mathbb{Z})^*$ or its extension. The algorithms are more efficient than solving the relative discrete log. problems in certain cases. This paper shows that Cheon's algorithms are faster than the result obtained by the complexity analysis in Cheon's paper, and also an improvement of one of the algorithms. These two results lead that the q-weak Diffie-Hellman problem on an abelian group of prime order p can be solved within $O(\sqrt{p/d})$ group operations, where $d \leq q$ is a positive divisor of $p-1$. It is faster than the result shown by Cheon that needs $O(\log p(\sqrt{p/d}+\sqrt{d}))$ group operations. Moreover, this paper discusses how one chooses the group order so that the algorithms are inefficient, and shows a condition for the group order.

Keywords: Cheon's algorithms, generic groups, square-root algorithms, the baby-step giant-step algorithm, pairing-based cryptography

1 Introduction

The Weil and Tate pairings have been used to solve the discrete log. problems on (hyper-)elliptic curves [MOV91, FR94]. However, around the end of the past century, positive usages of the Weil/Tate pairing for cryptography have been proposed by Ohgishi, Sakai, and Kasahara [OSK99] and Joux [Jou00] independently. They used the pairing to construct cryptographic protocols with nice properties. Then, Boneh and Franklin [BF01] proposed an IND-ID-CCA secure identity-based cryptography under the Weil Diffie-Hellman assumption in the random oracle model. Its provable security is fundamentally obtained from the properties of the pairing. Afterwards, a lot of provable secure protocols has been proposed by using the pairings. Moreover, many kind of protocols have been carried out by using the pairings.

In 2002, Mitsunari, Sakai, and Kasahara [MSK02] proposed a traitor tracing protocol. It is based on the weak Diffie-Hellman problem that takes a long series of group elements as its input. Then many protocols without random oracles have been proposed based on weak Diffie-Hellman-like problems, e.g. [BB04b, BB04a, BBG05, Oka06]. We call such kind of problems the "pairing-related problems" in this paper.

Any pairing-related problems known can be reduced to a discrete log. problem. Moreover, no efficient algorithm more than solving the relative discrete log. problems had been known for the pairing-related problems.

In 2006, Cheon [Che06] proposed breakthrough algorithms for pairing-related problems. He used that the exponents of an element in an abelian group G of prime order p form the ring $\mathbb{Z}/p\mathbb{Z}$ structure even if G is a generic group [Sho97], and the algorithms were constructed by using a subgroup of $(\mathbb{Z}/p\mathbb{Z})^*$ or its generalization. The algorithms are more efficient than solving the relative discrete log. problems in certain cases.

This paper shows that Cheon's algorithms are faster than the result obtained by the complexity analysis in [Che06], and also an improvement of one of the algorithms. These two results lead that the q-weak Diffie-Hellman problem (strictly defined in Sect. 2) on an abelian group of prime order p can be solved within $O\left(\sqrt{p/d}\right)$ group operations, where $d \leq q$ is a positive divisor of $p-1$. That is faster than the result in [Che06] that needs $O\left(\log p\left(\sqrt{p/d}+\sqrt{d}\right)\right)$ group operations. Moreover, this paper discusses how one chooses the group order so that the algorithms are inefficient, and shows a condition for the group order.

The organization of this paper is as follows: Sect. 2 defines the discrete log. problem and the paring-related problems dealt with in this paper, and summarizes the algorithms for those problems. Sect. 3 briefly reviews Cheon's algorithms and their complexity shown in [Che06]. Then, Sect. 4 shows a better complexity analysis for the algorithms, and Sect. 5 an improvement for the weak Diffie-Hellman problem. Furthermore, Sect. 6 discusses about the group order so that the algorithms are inefficient. Finally, Sect. 7 concludes this paper.

In this paper, we estimate for the time complexity by group operations, and for the space complexity by group elements respectively following from [Che06].

* Institute of Information Security, 2-14-1, Tsuruya-cho Kanagawa-ku, Yokohama 221-0835, Japan
† RDI at Chuo Univ., 1-13-27, Kasuga Bunkyo-ku, Tokyo 112-8551, Japan

2 DLP and pairing-related problems

In this section, we recall the discrete log. problem and the pairing-related problems which are dealt with in this paper.

Let G be an abelian group whose order is a large prime number p, $g \in G$, and $\alpha \in (\mathbb{Z}/p\mathbb{Z})^*$.

Definition 1 (Discrete log. problem)**.** The discrete log. problem (hereafter DLP) asks α for a pair $(g, [\alpha]g)$.

DLP can be solved within $O\left(\sqrt{p}\right)$ group operations by using Shanks's baby-step giant-step algorithm [Sha71] (hereafter BSGS) and Pollard's algorithm [Pol78]. Those algorithms are usually called "square-root algorithms." For the details of the algorithms, see [Tes01] for example.

This paper deals with the pairing-related problems shown below.

Definition 2 (q-weak Diffie-Hellman problem [MSK02])**.** The q-weak Diffie-Hellman problem (hereafter q-WDHP) asks $[1/\alpha]g$ for a tuple

$$(g, [\alpha]g, [\alpha^2]g, \ldots, [\alpha^q]g).$$

Definition 3 (q-strong Diffie-Hellman problem [BB04b])**.** The q-strong Diffie-Hellman problem (hereafter q-SDHP) asks $([1/(\alpha + c)]g, c)$, where c is any element in $(\mathbb{Z}/p\mathbb{Z})^*$, for a tuple

$$(g_0 \in G_0, g, [\alpha]g, [\alpha^2]g, \ldots, [\alpha^q]g),$$

where G_0 is an abelian group of order p.

If $g_0 = g$ (and $G_0 = G$), the problem is called the "q-simplified strong Diffie-Hellman problem (hereafter q-SSDHP)" [BBG05, Appendix A.4].

The blind and partially blind signatures proposed by [Oka06] are based on a 2 variable variant of q-SDHP.

Definition 4 (q-bilinear Diffie-Hellman inversion problem [BB04a])**.** The q-bilinear Diffie-Hellman inversion problem (hereafter q-BDHIP) asks $e(g, g)^{1/\alpha} \in G_1$ for a tuple

$$(g, [\alpha]g, [\alpha^2]g, \ldots, [\alpha^q]g),$$

where G_1 is a (multiplicative) abelian group of order p and

$$e : G \times G \to G_1$$

a non-degenerate bilinear map.

Definition 5 ($(q+1)$-bilinear Diffie-Hellman exponent problem [BBG05])**.** $(q+1)$-bilinear Diffie-Hellman exponent problem (hereafter $(q+1)$-BDHEP) asks $e(h, g)^{\alpha^{q+1}} \in G_1$ for a tuple

$$(h \in G_0, g, [\alpha]g, \ldots, [\alpha^q]g, [\alpha^{q+2}]g, \ldots, [\alpha^{2(q+1)}]g),$$

where G_0 is an abelian group of order p, G_1 a (multiplicative) abelian group of order p, and

$$e : G_0 \times G \to G_1$$

a non-degenerate bilinear map.

Some reductions are known between the above problems:

- Both of q-WDHP and q-SDHP can be reduced to DLP in polynomial time.
- q-SSDHP can be reduced to both of q-WDHP and q-SDHP in polynomial time.
- Each of q-BDHIP and $(q+1)$-BDHEP can be reduced to q-WDHP in polynomial time.

See [BBG05, Wei05] for the details.

Until recently, no other efficient algorithm had been known for the pairing-related problems than solving DLP obtained by the above reductions.

3 Cheon's algorithms

In EUROCRYPT 2006, Cheon [Che06] proposed breakthrough algorithms for the pairing-related problems. Using that the exponents of an element in an abelian group G of prime order p form the ring $\mathbb{Z}/p\mathbb{Z}$ structure even if G is a generic group [Sho97], Cheon's algorithms reduce their complexity by Pohlig-Hellman [PH78] like method over $(\mathbb{Z}/p\mathbb{Z})^*$ or its extension. The algorithms are more efficient than solving the relative discrete log. problems in certain cases.

Cheon's basic algorithm finds the discrete log. α for given $(g, [\alpha]g, [\alpha^d]g)$, where d is a positive divisor of $p-1$. Therefore, if there exists $d \mid p-1$ with $d \leq q$, the algorithm can be applied to the pairing-related problems.

Algorithm 1 shows an outline of the algorithm. For the correctness of the algorithm, see [Che06].

Algorithm 1 Cheon's Algorithm

Input: G: an abelian group of prime order p, $g, g_1, g_d \in G$, and $d \in \mathbb{N}$ such that $d \mid p-1$
Output: $\alpha \in \mathbb{Z}/p\mathbb{Z}$ such that $g_1 = [\alpha]g$ and $g_d = [\alpha^d]g$

1: Find a generator $\zeta_0 \in (\mathbb{Z}/p\mathbb{Z})^*$
2: $\zeta \leftarrow \zeta_0^d$
3: $d_1 \leftarrow \left\lceil \sqrt{(p-1)/d} \right\rceil$
4: Find $0 \leq u_1, v_1 < d_1$ such that $[\zeta^{-u_1}]g_d = [\zeta^{d_1 v_1}]g$ by BSGS
5: $k_0 \leftarrow d_1 v_1 + u_1$
6: $d_2 \leftarrow \left\lceil \sqrt{d} \right\rceil$
7: Find $0 \leq u_2, v_2 < d_2$ such that $[\zeta_0^{-u_2(p-1)/d}]g_1 = [\zeta_0^{k_0 + d_2 v_2 (p-1)/d}]g$ by BSGS
8: **return** $\zeta_0^{k_0 + (d_2 v_2 + u_2)(p-1)/d}$

Cheon [Che06] showed Theorem 1 below corresponding to the complexity of Algorithm 1.

Theorem 1 (Cheon)**.** *Let g be an element of prime order p in an abelian group. Suppose that d is a positive divisor of $p-1$. If g, $[\alpha]g$ and $[\alpha^d]g$ are given, α can be computed within $O\left(\log p \left(\sqrt{p/d} + \sqrt{d}\right)\right)$ group operations using space for $O\left(\max\left(\sqrt{p/d}, \sqrt{d}\right)\right)$ group elements.*

Furthermore, [Che06] showed a $p+1$ variant of Algorithm 1, i.e. an algorithm using a positive divisor d of $p+1$ instead of $p-1$, by applying the similar manner in Algorithm 1 on $(\mathbb{F}_{p^2})^*/(\mathbb{Z}/p\mathbb{Z})^*$ instead of $(\mathbb{Z}/p\mathbb{Z})^*$. [Che06] showed Theorem 2 below corresponding to the $p+1$ variant.

Theorem 2 (Cheon)**.** *Let g be an element of prime order p in an abelian group. Suppose that d is a positive divisor of $p+1$ and $[\alpha^i]g$ for $i = 1, 2, \ldots, 2d$ are given. Then α can be computed within $O\left(\log p\left(\sqrt{p/d}+d\right)\right)$ group operations using space for $O\left(\max\left(\sqrt{p/d}, \sqrt{d}\right)\right)$ group elements.*

4 Better complexity analysis

This section shows that the upper bound of group operations required in Algorithm 1 is lower than the bound in Theorem 1. i.e. Algorithm 1 is faster than the result shown in Theorem 1.

The complexity of Algorithm 1 is dominated by BSGS in Steps 4 and 7. In Step 4, one needs to compute $[\zeta^{-u_1}]g$ for $u_1 = 0, 1, \ldots, d_1$, where $d_1 = \left\lceil \sqrt{(p-1)/d} \right\rceil$, from given $\zeta^{-1} \in (\mathbb{Z}/p\mathbb{Z})^*$ and g_d, and also to compute $[\zeta^{v_1 d_1}]g$ for $v_1 = 0, 1, \ldots, d_1$ from given $\zeta^d \in (\mathbb{Z}/p\mathbb{Z})^*$ and g. Similarly, in Step 7, one needs to compute $[\zeta^{-u_1}]g$ for $u_2 = 0, 1, \ldots, d_2$, where $d_2 = \left\lceil \sqrt{d} \right\rceil$, from given $\zeta^{-1} \in (\mathbb{Z}/p\mathbb{Z})^*$ and g_d, and also to compute $[\zeta^{v_1 d_1}]g$ for $v_2 = 0, 1, \ldots, d_2$ from given $\zeta^d \in (\mathbb{Z}/p\mathbb{Z})^*$ and g.

Now, we let

$$\begin{aligned}(g_t, \beta, n) \quad \in \quad & \{(g_d, \zeta^{-1}, d_1), (g, \zeta^{d_1}, d_1), \\ & (g_1, \zeta_0^{-(p-1)/d}, d_2), (\zeta_0^{k_0} g, \zeta_0^{d_2(p-1)/d}, d_2)\},\end{aligned}$$

then computing the repeated integer multiplications of an element in G appeared in Steps 4 and 7 can be concluded by a problem given as follows:

Problem 1. asks

$$(g_t, [\beta]g_t, [\beta^2]g_t, \ldots, [\beta^{n-1}]g_t)$$

for given

$$g_t \in G, \beta \in (\mathbb{Z}/p\mathbb{Z})^*, \text{and } n \in \mathbb{Z}.$$

Problem 1 corresponds to "find

$$(g_t, [2]g_t, [3]g_t, \ldots, [n-1]g_t)$$

for given $g_t \in G$ and $n \in \mathbb{Z}$" in ordinary BSGS. The problem in ordinary BSGS can be solved by using the recurrent sequence

$$[i]g_t = [i-1]g_t + g_t, i = 1, \ldots, n-1.$$

In this computation, $[i]g_t$ can be obtained by a group operation from $[i-1]g_t$. Thus, the problem can be solved within $O(n)$ group operations.

In the analysis in [Che06], using the similar manner in ordinary BSGS, one computes $[\beta^i]g_t$ from $[\beta^{i-1}]g_t$ by the recurrence

$$[\beta^i]g_t = [\beta][\beta^{i-1}]g_t.$$

It needs $O(\log p)$ group operations to compute $[\beta^i]g_t$ from $[\beta^{i-1}]g_t$ by the binary method. Therefore, one needs $O(n \log p)$ group operations to solve Problem 1 by the manner in the analysis.

In the following, we discuss to use an on-line precomputation table in the above computation. While on-line precomputation tables are usually used for practical implementation of integer multiplication, this section uses a table in order to reduce the asymptotic complexity.

A scenario of the computation with a table is to use a table spanned just $\{g_t, [2]g_t, [3]g_t, \ldots, [n-1]g_t\}$. However, a table in this scenario seems difficult to construct because the elements in $\{g_t, [2]g_t, [3]g_t, \ldots, [n-1]g_t\}$ are discretely distributed in $\mathbb{Z}/p\mathbb{Z}$ in general. Another scenario is to use a table spanned all elements in $\mathbb{Z}/p\mathbb{Z}$. However, the efficiency of this scenario is not so clear because the number of elements asked by the problem is smaller than the cardinality of $\mathbb{Z}/p\mathbb{Z}$, i.e. p, in general. Below shows that the latter scenario is actually efficient.

Let c be a small positive integer with regarded as a constant and $b = \lceil \sqrt[c]{p} \rceil$. Then, any $\delta \in \mathbb{Z}/p\mathbb{Z}$ can be represented by a b-adic expansion as

$$\delta \equiv \delta_0 + \delta_1 b + \delta_2 b^2 + \cdots + \delta_{c-1} b^{c-1} \bmod p, \quad (1)$$

where $0 \le \delta_i < b$ for $i = 0, \ldots, c-1$. We consider the computation with a precomputation table by using this representation.

One can construct the on-line precomputation table $T = \{T_i | i = 0, \ldots, c-1\}$ as

$$\begin{aligned} T_i &= (T_{(i,0)}, T_{(i,1)}, \ldots, T_{(i,b-1)}) \\ &= (1, [b^i]g_t, [2b^i]g_t, \ldots, [(b-1)b^i]g_t). \end{aligned}$$

In fact, T can be obtained by the following manner.

First, one puts $T_{(0,1)} = g_t$, then computes $T_{(i,1)}$ for $i = 0, \ldots, c-1$ by the recurrence

$$T_{(i,1)} = [b]T_{(i-1,1)}.$$

Each recurrence can be computed within $O(\log p)$ group operations by the binary method. Thus, it needs $(c-1)O(\log p)$ group operations to obtain $T_{(i,1)}$ for $i = 0, \ldots, c-1$.

Second, one computes T_i for each $i = 0, \ldots, c-1$. It can be done by the recurrence

$$T_{(i,j)} = T_{(i,j-1)} + T_{(i,1)}.$$

Each recurrence can be computed a group operation, so that T_i can be obtained by $b-2$ group operations for each i.

Consequently, the table T can be obtained within

$$\begin{aligned} O\left((c-1)\log p + c(b-2)\right) &= O\left(c\left(\log p + b\right)\right) \\ &= O\left(\sqrt[c]{p}\right) \end{aligned}$$

group operations. In addition, T needs the space for $O(cb) = O\left(\sqrt[c]{p}\right)$ group elements.

For δ given by (1), $[\delta]g_t$ can be computed as follows:

$$\begin{aligned} [\delta]g_t &= [\delta_0 + \delta_1 b + \delta_2 b^2 + \cdots + \delta_{c-1} b^{c-1}]g_t \\ &= [\delta_0]g_t + [\delta_1 b]g_t + [\delta_2 b^2]g_t + \cdots + [\delta_{c-1} b^{c-1}]g_t \\ &= T_{(0,\delta_0)} + T_{(1,\delta_1)} + T_{(2,\delta_2)} + \cdots + T_{(c-1,\delta_{c-1})}. \end{aligned}$$

Therefore, it can be obtained by $c-1$ group operations by using the table T. Computing $[\delta]g_t$ for all $\delta \in \{\beta^i \mid i = 0, \ldots, n-1\}$, one can obtain $(g_t, [\beta]g_t, [\beta^2]g_t, \ldots, [\beta^{n-1}]g_t)$. It can be done within $(n-1)(c-1) = O(n)$ group operations.

Summarizing, we see that Problem 1 can be solved within $O\left(\sqrt[c]{p}+n\right)$ group operations using the table with $O\left(\sqrt[c]{p}\right)$ group elements.

Algorithm 1 needs to solve Problem 1 for $n = d_1$ 2 times and also for $n = d_2$ 2 times. Therefore, it needs

$$O\left(\sqrt[c]{p}+d_1+d_2\right) = O\left(\sqrt[c]{p}+\sqrt{p/d}+\sqrt{d}\right)$$

group operations. Besides,

$$\sqrt[c]{p} \leq \sqrt{p/d}+\sqrt{d}.$$

can be obtained by setting $c \geq 4$. Similarly, Algorithm 1 needs the space for

$$O\left(\sqrt[c]{p}+\sqrt{p/d}+\sqrt{d}\right) = O\left(\sqrt{p/d}+\sqrt{d}\right)$$

group elements if $c \geq 4$.

From Theorem 1 and the above discussion, we have Theorem 1' below.

Theorem 1'. *Let g be an element of prime order p in an abelian group. Suppose that d is a positive divisor of $p-1$. If g, $[\alpha]g$ and $[\alpha^d]g$ are given, α can be computed within $O\left(\sqrt{p/d}+\sqrt{d}\right)$ group operations using space for $O\left(\max\left(\sqrt{p/d}, \sqrt{d}\right)\right)$ group elements.*

Theorem 1' shows that Algorithm 1 is $O(\log p)$ times faster than the result shown by [Che06].

By the similar discussion for the $p+1$ variant, Theorem 2' shown below can be obtained from Theorem 2.

Theorem 2'. *Let g be an element of prime order p in an abelian group. Suppose that d is a positive divisor of $p+1$ and $[\alpha^i]g$ for $i = 1, 2, \ldots, 2d$ are given. Then α can be computed within $O\left(\sqrt{p/d}+d\right)$ group operations using space for $O\left(\max\left(\sqrt{p/d}, \sqrt{d}\right)\right)$ group elements.*

Note that the number of group operations shown in Theorem 2' can be obtained if $c \geq 3$. However, the space needs for $O\left(\max\left(\sqrt{p/d}, d\right)\right)$ group elements in the case of $c = 3$. Therefore, $c \geq 4$ is necessary for the discussion for the $p+1$ variant as well as Algorithm 1.

5 Improvement for large d

The complexity of Algorithm 1 achieves its lowest as $O\left(\sqrt[4]{p}\right)$ group operations when $d = O\left(\sqrt{p}\right)$. Besides, it increases monotonically as d increases for $O(d) > O\left(\sqrt{p}\right)$.

This section proposes an improvement of Algorithm 1 for q-WDHP. It is more efficient than Algorithm 1 if $O(d) > O\left(\sqrt{p}\right)$. The improvement can be applied to the problems that can be reduced to q-WDHP obviously.

This section deals with Problem 2 shown below.

Problem 2. asks $[\alpha^k]g$ for given $k \in \mathbb{Z}$, $[\alpha^r]g$, $[\alpha^d]g$ such that

$$k \equiv r \bmod d, \tag{2}$$

where d is a positive divisor of $p-1$, and $0 \leq r < d$.

q-WDHP can be reduced to Problem 2 by setting $k = p-2$, and moreover, q-SSDHP can be directly reduced to the problem by $k = q+1$.

Corresponding to Problem 2, we have:

Theorem 3. *Let g be an element of prime order p in an abelian group and k a positive integer. Suppose that d is a positive divisor of $p-1$, and g, $[\alpha^d]g$ and $[\alpha^r]$ are given, where $r \equiv k \bmod d$, $0 \leq r < d$. Then $[\alpha^k]g$ can be computed within $O\left(\sqrt{p/d}\right)$ group operations using space for $O\left(\sqrt{p/d}\right)$ group elements.*

Proof. First, ζ and k_0 in Algorithm 1 can be obtained by executing Steps 1-5 of the algorithm. They need $O\left(\sqrt{p/d}\right)$ group operations and space for $O\left(\sqrt{p/d}\right)$ group elements. Then α^d can be obtained as

$$\alpha^d = \zeta^{k_0}.$$

Next,

$$s = \frac{k-r}{d}$$

can be obtained as as an integer since (2).

Then, $[\alpha^k]g$ can be computed as follows:

$$[\alpha^k]g = [\alpha^{sd+r}]g = [(\alpha^d)^s\alpha^r]g = [(\alpha^d)^s][\alpha^r]g$$

from α^d, s, and given $[\alpha^r]g$. In fact, one can compute $\tau \in \mathbb{Z}$ such that $\tau \equiv (\alpha^d)^s \bmod p$ in $[0, p-1]$. Therefore, $[\alpha^k]g$ can be computed by using τ as follows:

$$[\alpha^k]g = [\tau][\alpha^r]g.$$

It can be done within $O(\log p)$ group operations by the binary method.

□

Corollary 1 below immediately follows from Theorem 3.

Corollary 1. *q-WDHP can be solved within $O\left(\sqrt{p/d}\right)$ group operations using space for $O\left(\sqrt{p/d}\right)$ group elements.*

Unlike Algorithm 1, the complexity of the improvement decreases monotonically as d increases.

Remark 1. In the present protocols based on the pairing-related problem, the problem size is $O(q \log p)$, Therefore, q should be a polynomial of $\log p$. Moreover, d should also be a polynomial of $\log p$ because $d \leq q$. Consequently, it seems that the effect of the improvement shown in this section is not so large on the protocols existed, whereas the improvement is slightly faster than Algorithm 1 even for such small d. However, we need to pay attention to the improvement if we makes a new problem for coming protocols. Because the improvement needs only constant numbers of group elements as its input as well as Algorithm 1.

6 Discussion

This section discusses how one chooses the group order so that both of Algorithm 1 and the $p+1$ variant are inefficient for the pairing-related problems. Moreover, this section shows a condition for the group order and the probability of the order satisfied the condition.

In this section, we assume that d is small enough, i.e. a polynomial of $\log p$.

From Theorem 1 (Theorem 2), it seems that Algorithm 1 (resp. the $p+1$ variant) is more efficient than the ordinary square-root algorithms if

$$d = \Omega\left(\log^2 p\right),$$

which can be obtained from

$$O\left(\log p\sqrt{p/d}\right) \leq O\left(\sqrt{p}\right).$$

In order to avoid the ability of Algorithm 1 and the $p+1$ variant, [Che06] recommended to increase the key size or to choose p so that both of $p+1$ and $p-1$ have no small divisor greater than $\log^2 p$ due to this result.

However, from Theorem 1' (Theorem 2') newly obtained in Sect. 4, we see that Algorithm 1 (resp. the $p+1$ variant) is more efficient than the ordinary square-root algorithms if

$$d = \Omega\left(1\right).$$

i.e. there is asymptotically no lower bound of d for the efficiency of the algorithms.

On the other hand, Algorithm 1 needs $[\alpha^d]g$ as its input, and the $p+1$ variant needs $[\alpha^i]g$ for $i = 1, \ldots, 2d$. Therefore, Algorithm 1 (the $p+1$ variant) becomes the same complexity as ordinary square-root algorithms if $p-1$ (resp. $p+1$) has no divisor $1 < d \leq q$ (resp. $1 < d \leq q/2$).

Unfortunately, $p \pm 1$ always has small divisors if p is cryptographically interesting size. i.e. $2 \mid p \pm 1$,

$$4 \mid p-1 \text{ if } p \equiv 1 \bmod 4,$$
$$4 \mid p+1 \text{ if } p \equiv 3 \bmod 4,$$

and moreover,

$$3 \mid p-1 \text{ if } p \equiv 1 \bmod 3,$$
$$3 \mid p+1 \text{ if } p \equiv 2 \bmod 3.$$

However, the effect of these divisors on the algorithms seems to be small. Accordingly, the simplest strategy to avoid the ability of the algorithms is to choose p depending on q so that $d > q$ for any prime $d \mid p_-$ and $d > q/2$ for any prime $d \mid p_+$, where $p_\pm$ is given by the following table.

$p \bmod 12$	1	5	7	11
p_-	$\frac{p-1}{12}$	$\frac{p-1}{4}$	$\frac{p-1}{6}$	$\frac{p-1}{2}$
p_+	$\frac{p+1}{2}$	$\frac{p+1}{6}$	$\frac{p+1}{4}$	$\frac{p+1}{12}$

Now, suppose p_- be a random positive integer. Then the probability P_- that p_- has no prime divisor less than or equal to q is given as follows:

$$P_- = \prod_{l \leq q}\left(1 - \frac{1}{l}\right),$$

where l is in prime numbers. Therefore, we see directly from Mertens's theorem [HW79, Theorem 429] that

$$P_- \approx \frac{e^{-\gamma}}{\log_e q} = \Theta\left(\frac{1}{\log q}\right),$$

where γ denotes Euler's constant.

By the similar discussion, the probability P_+ that p_+ has no prime divisor less than or equal to $q/2$ can be obtained as

$$P_+ = \Theta\left(\frac{1}{\log q}\right).$$

Assuming p_- and p_+ are independent of each other, we see that there exists the suitable p, i.e. a prime number p such that $d > q$ for any prime $d \mid p_-$ and $d > q/2$ for any prime $d \mid p_+$, with the probability $\Theta\left(1/\log^2 q\right)$. Consequently, a randomly selected prime order p generally does not satisfy the condition in cryptographically settings, even though the order can be obtained by cut-and-try in many time.

On the other hand, the studies of constructing a "pairing-friendly curve" are still ongoing [MNT01, BLS04b, BLS04a, GMV04, BW05, BN06, SB06, CKT06]. Therefore, construction of a pairing-friendly curve whose order satisfies the condition given in this section is an interesting further research subject.

Remark 2. The condition shown in this section is not sufficient in certain cases, i.e. there are the cases that Algorithm 1 (the $p+1$ variant) efficiently works even if any prime divisor of p_- (resp. p_+) is larger than q (resp. $q/2$). Because the efficiency of algorithms is higher in these cases, we should take care of the cases.

$(q+1)$-BDHEP takes $[\alpha^i]g$ for $i = 0, \ldots, q, q+2, 2q+2$ as its input. Therefore, in order to avoid the ability of Algorithm 1 on $(q+1)$-BDHEP, we should be choose p so that p_- has no prime divisor $d \leq 2q+2$ except $d = q+1$.

q-BDHIP takes $[\alpha^i]g$ for $i = 0, \ldots, q$ as its input, and moreover, it takes

$$e : G \times G \to G_1,$$

so that, for $k = 0, \ldots, 2q$, $e(g,g)^{\alpha^k}$ can be obtained by

$$e(g,g)^{\alpha^k} = e([\alpha^i]g, [\alpha^j]g), 0 \leq i, j \leq q, k = i + j.$$

Executing the algorithms on G_1 with the selected elements in $\{e(g,g)^{\alpha^k} \mid k = 0, \ldots, 2q\}$ as their input, one can obtain $e(g,g)^{1/\alpha}$. Therefore, in order to avoid the ability of Algorithm 1 (the $p+1$ variant) on q-BDHIP, we should be choose p so that p_- (resp. p_+) has no prime divisor $d \leq 2q$ (resp. $d \leq q$).

7 Conclusion

This paper showed that Cheon's algorithms shown in [Che06] are faster than the result obtained by the complexity analysis in [Che06] Moreover, This showed an improvement of one of the algorithms. These two results lead that the q-weak Diffie-Hellman problem on an abelian group of prime order p can be solved within

$O\left(\sqrt{p/d}\right)$ group operations, where $d \leq q$ is a positive divisor of $p-1$. It is faster than the result shown in [Che06] that needs $O\left(\log p\left(\sqrt{p/d}+\sqrt{d}\right)\right)$ group operations. Moreover, this paper discusses how one chooses the group order so that the algorithms are inefficient, and showed a condition for the group order and the probability of the order satisfied the condition.

Acknowledgment

This research was partially supported by "The Research on Security and Reliability in Electronic Society," Chuo University 21st Century COE Program.

References

[BB04a] D. Boneh and X. Boyen, *Efficient selective-ID secure identity-based encryption without random oracles*, Advances in Cryptology - EUROCRYPT 2004 (C. Cachin and J. Camenisch, eds.), LNCS 3027, Springer-Verlag, 2004, pp. 223–238.

[BB04b] _____, *Short signatures without random oracles*, Advances in Cryptology - EUROCRYPT 2004 (C. Cachin and J. Camenisch, eds.), LNCS 3027, Springer-Verlag, 2004, pp. 56–73.

[BBG05] D. Boneh, X. Boyen, and E.-J. Goh, *Hierarchical identity based encryption with constant size ciphertext*, Cryptology ePrint Archive, Report 2005/015, 2005, An extended abstract appears in Advances in Cryptology - EUROCRYPT 2005 (R. Cramer, ed.), LNCS 3494, Springer-Verlag, 2005, pp. 440–456.

[BF01] D. Boneh and M. Franklin, *Identity-based encription from the Weil pairing*, Advances in Cryptology - CRYPTO 2001 (J. Kilian, ed.), LNCS 2139, Springer-Verlag, 2001, pp. 213–229.

[BLS04a] P. S.L.M. Barreto, B. Lynn, and M. Scott, *Efficient implementation of pairing-based cryptosystems*, Journal of Cryptology **17** (2004), 321–334.

[BLS04b] _____, *On the selection of pairing-friendly groups*, Selected Areas in Cryptography, 10th Annual International Workshop, SAC 2003 (H. Handschuh and M.A. Hasan, eds.), LNCS 3006, Springer-Verlag, 2004, pp. 17–25.

[BN06] P. S.L.M. Barreto and M. Naehrig, *Pairing-friendly elliptic curves of prime order*, Selected Areas in Cryptography: 12th International Workshop, SAC 2005 (B. Preneel and S. Tavares, eds.), LNCS 3897, Springer-Verlag, 2006, pp. 319–331.

[BW05] F. Brezing and A. Weng, *Elliptic curves suitable for pairing based cryptography*, Design, Codes and Cryptography **37** (2005), 133–141.

[Che06] J. H. Cheon, *Security analysis of strong Diffie-Hellman problem*, Advances in Cryptology - EUROCRYPT 2006 (S. Vaudenay, ed.), LNCS 4004, Springer-Verlag, 2006, pp. 1–11.

[CKT06] A. Comuta, M. Kawazoe, and T. Takahashi, *How to construct pairing-friendly curves for the embedding degree $k = 2n$, n is an odd prime*, Cryptology ePrint Archive, Report 2006/427, 2006.

[FR94] G. Frey and H.-G. Rück, *A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves*, Math. Comp. **62** (1994), 865–874.

[GMV04] S. D. Galbraith, J. McKee, and P. Valença, *Ordinary abelian varieties having small embedding degree*, Cryptology ePrint Archive, Report 2004/365, 2004.

[HW79] G. H. Hardy and E. M. Wright, *An introduction to the theory of numbers*, 5th ed., Oxford U. P., 1979.

[Jou00] A. Joux, *One round protocol for tripartite Diffie-Hellman*, ANTS-IV (W. Bosma, ed.), LNCS 1838, Springer-Verlag, 2000, pp. 385–393.

[MNT01] A. Miyaji, M. Nakabayashi, and S. Takano, *New explicit conditions of elliptic curve traces for FR-reduction*, IEICE Trans. Fundamentals **E84-A** (2001), no. 5, 1234–1243.

[MOV91] A. Menezes, T. Okamoto, and S. Vanstone, *Reducing elliptic curve logarithms to logarithms in a finite fields*, Proc. of STOC, 1991, pp. 80–89.

[MSK02] S. Mitsunari, R. Sakai, and M. Kasahara, *A new traitor tracing*, IEICE Trans. Fundamentals **E85-A** (2002), no. 2, 481–484.

[Oka06] T. Okamoto, *Efficient blind and partially blind signatures without random oracles*, TCC 2006 (S. Halevi and T. Rabin, eds.), LNCS 3876, Springer-Verlag, 2006, pp. 80–99.

[OSK99] K. Ohgishi, R. Sakai, and M Kasahara, *Notes on ID-based key sharing systems over elliptic cureve*, Tech. Report ISEC99-57, IEICE , 1999, in Japanese.

[PH78] G. C. Pohlig and M. E. Hellman, *An improved algorithm for comuting logarithms over* GF(p) *and its cryptographic significance*, IEEE Trans. on Info. Theory **IT-24** (1978), 106–110.

[Pol78] J. M. Pollard, *Monte Carlo methods for index computation* (mod p), Math. Comp. **32** (1978), 918–924.

[SB06] M. Scott and P. S.L.M. Barreto, *Generating more MNT elliptic curves*, Designs, Codes and Cryptography **38** (2006), 209–217.

[Sha71] D. Shanks, *Class number, a theory of factrization, and genera*, Proc. of Symp. Math. Soc. 20, 1971, pp. 415–440.

[Sho97] V. Shoup, *Lower bounds for discrete logarithms and related problems*, Advances in Cryptology - EUROCRYPT '97 (W. Fumy, ed.), LNCS 1233, Springer-Verlag, 1997, pp. 256–266.

[Tes01] E. Teske, *Square-root algorithms for the discrete logarithm problem (A survey)*, Public-Key Cryptography and Computational Number Theory, Walter de Gruyter, Berlin-New York, 2001, pp. 283–301.

[Wei05] V. K. Wei, *Tight reductions among strong Diffie-Hellman assumptions*, Cryptology ePrint Archive, Report 2005/057, 2005.