JSIAM Letters Vol.6 (2014) pp.53–56 ©2014 Japan Society for Industrial and Applied Mathematics JSIAM Letters

An experiment of number field sieve for discrete logarithm problem over GF(p^n)

Kenichiro Hayasaka[1], Kazumaro Aoki[2], Tetsutaro Kobayashi[2] and Tsuyoshi Takagi[3]

[1] Graduate School of Mathematics, Kyushu University, 744, Motooka, Nishiku, Fukuoka-shi, Fukuoka 819-0395, Japan
[2] NTT Secure Platform Laboratories, 3-9-11 Midori-cho, Musashino-shi, Tokyo 180-8585, Japan
[3] Institute of Mathematics for Industry, Kyushu University, 744, Motooka, Nishiku, Fukuoka-shi, Fukuoka 819-0395, Japan

E-mail *k-hayasaka@math.kyushu-u.ac.jp*

Received October 7, 2013, Accepted January 19, 2014

Abstract

The security of the optimal Ate pairing using the BN curves is based on the hardness of the DLP over GF(p^{12}). At CRYPTO 2006, Joux et al. proposed the number field sieve over GF(p^n), but the number field sieve needs multi-dimensional sieving. In this paper, we deal with the multi-dimensional sieving, and discuss its parameter sizes such as the dimension of sieving and the size of the sieving region from some experiments of the multi-dimensional sieving. Using efficient parameters, we have solved the DLP over GF(p^{12}) of 203 bits in about 43 hours using a PC of 16 CPU cores.

Keywords pairing, discrete logarithm problem, number field sieve, extension field, lattice sieve

Research Activity Group Algorithmic Number Theory and Its Applications

1. Introduction

Pairing-based cryptography has attracted us due to novel cryptographic protocols such as ID-based cryptography, functional encryption, etc. Many efficient implementations of pairing have been reported, and one of the most efficient algorithms for computing pairing is the optimal Ate pairing [1] using the BN curves [2]. The security of pairing-based cryptography using the BN curves is based on the hardness of the discrete logarithm problem (DLP) over finite fields GF(p^{12}).

The asymptotically fastest algorithm for solving the DLP over prime fields GF(p) is the number field sieve [3]. At CRYPTO 2006, Joux et al. extended the number field sieve to the case of extension fields GF(p^n) of degree n and characteristic p [4]. The complexity of solving the DLP over finite fields GF(p^{12}) of 3072 bits by the number field sieve is estimated to be 2^{128} [2]. There are two experimental reports on the implementation of the number field sieve over extension fields GF(p^n) of degrees $n = 3$ [4] and $n = 6$ [5,6]. However, to the best of our knowledge, there is no experimental report on the hardness of the DLP over finite fields GF(p^{12}) by the number field sieve. In order to correctly estimate the security of the pairing-based cryptography, we need some experimental evaluations of number field sieve over finite field GF(p^{12}).

The number field sieve over extension field GF(p^n) has a substantially different sieving step from that over prime field GF(p). There are two sieving algorithms, called the line sieve and the lattice sieve [7]. The large-scale implementation of the number field sieve over prime fields GF(p) deploys the lattice sieve of dimension two, but we have to construct the lattice sieve of dimension higher than two for the number field sieve over extension fields GF(p^{12}). The currently known reports on the multi-dimensional sieving have discussed only the case of dimension three [5,6].

In this paper, we propose the lattice sieving of dimension higher than two for the number field sieve over extension fields GF(p^{12}) by naturally extending the lattice sieve of dimension two. We implemented the proposed multi-dimensional lattice sieve over an extension field GF(p^{12}) of 203 bits, and we show some experimental data for accelerating the number field sieve by choosing suitable dimensions and sizes of the sieving region. Consequently, we have solved the DLP over the extension field GF(p^{12}) of 203 bits by the number field sieve using a PC of 16 CPU cores in about 43 hours.

2. Number field sieve over GF(p^n) [4]

2.1 DLP over GF(p^n)

We denote by GF(p^n)* the multiplicative group of a finite field of cardinality p^n, where p is a prime number and n is an extension degree. The DLP over a finite field GF(p^n) tries to find the non-negative smallest integer x that satisfies $\gamma^x = \delta$ for given δ, γ in GF(p^n)*. This discrete logarithm x is written as $\log_\gamma \delta$ in this paper.

2.2 Polynomial selection

We generate two irreducible polynomials $f_1, f_2 \in \mathbb{Z}[X] \setminus \{0\}$ that satisfy the following conditions: $f_1 \neq f_2$, $\deg f_1 = n$, f_1 is irreducible in GF(p)[X], $f_1 \mid f_2 \bmod p$. From the conditions, there exists $v \in$ GF(p^n) such that $f_1(v) = f_2(v) = 0$ in GF(p^n). Let α_1 and $\alpha_2 \in \mathbb{C}$ be roots of $f_1(X) = 0$ and $f_2(X) = 0$, respectively. There

are homomorphism maps $\phi_1 : \mathbb{Z}[\alpha_1] \to \mathrm{GF}(p^n),\ \alpha_1 \mapsto v, \phi_2 : \mathbb{Z}[\alpha_2] \to \mathrm{GF}(p^n),\ \alpha_2 \mapsto v$.

2.3 Searching relations

In the step of searching relations, we try to find many relations of certain polynomials of degree $t \geq 1$. Let $B_1, B_2 \in \mathbb{R}_{>0}$ be smoothness bounds associated with polynomials f_1, f_2 in Section 2.2. We define the factor bases $\mathcal{B}_1, \mathcal{B}_2$ by

$$\mathcal{B}_i = \{(q,g) \mid q : \text{prime},\ q \leq B_i,\ g : \text{irreducible monic polynomial in GF(q)[X]},\ g | f_i \bmod q,\ \deg g \leq t\}.$$

In this paper, we represent a polynomial $h_a(X) = \sum_{j=0}^{t} a_j X^j \in \mathbb{Z}[X]$ as a vector $a = (a_0, a_1, \ldots, a_t)^{\mathrm{T}} \in \mathbb{Z}^{t+1}$. For a given $H = (H_0, H_1, \ldots, H_t) \in \mathbb{R}_{>0}^{t+1}$, we define a $(t+1)$-dimensional region $\mathcal{H}_a(H)$ as

$$\mathcal{H}_a(H) = \{(a_0, a_1, \ldots, a_t)^{\mathrm{T}} \in \mathbb{Z}^{t+1} \mid |a_i| \leq H_i\ (0 \leq i \leq t),\ a_t \geq 0\}.$$

Here H and $\mathcal{H}_a$ are called a sieving interval and a sieving region, respectively. Next, the norm of $h_a(\alpha_i)$ is defined by $N(h_a(\alpha_i)) = |\mathrm{Res}(h_a, f_i)|$, where $\mathrm{Res}(h_a, f_i)$ is the resultant of $h_a(X)$ and $f_i(X)$ for $i = 1, 2$. In the step of searching relations, for the given sieving interval H and the smoothness bound B_1, B_2, we try to find $a \in \mathbb{Z}^{t+1}$ (called a hit tuple) that satisfies the following conditions: $N(h_a(\alpha_1))$ is B_1-smooth, $N(h_a(\alpha_2))$ is B_2-smooth, $\gcd(a_0, a_1, \ldots, a_t) = 1$, where an integer is B-smooth if and only if its prime factors are at most B. We denote by S the set of all hit tuples gathered in searching relations. In order to solve the correct discrete logarithm, the size of S is chosen as

$$\sharp S \geq \sharp\mathcal{B}_1 + \sharp\mathcal{B}_2 + 2n. \tag{1}$$

From $\phi_1(h_a(\alpha_1)) = \phi_2(h_a(\alpha_2))$, using $a \in S$ and the homomorphism maps in Section 2.2, we obtain relations of discrete logarithms. Consequently, we can compute the discrete logarithms of $\mathfrak{q} \in \mathcal{B}_i$ by solving the linear equations obtained from the relations.

3. Searching relations by multi-dimensional sieving

In the following, we describe the line sieve presented by Zajac [6]. If $q \mid N(h_a(\alpha_i))$ holds for a prime $q < B_i$ and $i = 1, 2$, then $q \mid N(h(\alpha_i))$ holds for any polynomial $h(X) = h_a(X) + kq$ where k is any integer. From this fact, we can search a hit tuple a divisible by q in the sieving region without performing the division of integers. Similarly, for $\mathfrak{q} = (q, g) \in \mathcal{B}_i (i = 1, 2)$, we have the property

$$g \mid h_a \bmod q \ \Rightarrow\ q^{\deg g} \mid N(h_a(\alpha_i)). \tag{2}$$

For $\forall \mathfrak{q} = (q, g) \in \mathcal{B}_i$ where $i \in 1, 2$ is fixed, we accumulate $\deg g \log q$ in a variable $L[a]$ if the sufficient condition in (2) for (q, g) and a is satisfied. Then, we can find a candidate of $a \in \mathcal{H}_a$ whose norm $N(h_a(\alpha_i))$ is B_i-smooth by checking $\log N(h_a(\alpha_i)) - L[a]$ is sufficiently small.

Let I_d be the identity matrix of size $d \times d$. For $\forall \mathfrak{q} = (q, g) \in \mathcal{B}_i (i = 1, 2)$ where $g = \sum_{j=0}^{\deg g} g_j X^j$, the set of all polynomials in $\mathbb{Z}[X]$ of degree less than $t+1$ that satisfy the sufficient condition in (2) is generated by the integer linear combinations of the columns of the following $(t+1) \times (t+1)$ matrix:

$$\left(\begin{array}{c|cccc} & g_0 & & 0 \\ qI_{\deg g} & g_1 & \ddots & \\ & \vdots & \ddots & g_0 \\ & g_{\deg g} & & g_1 \\ 0 & & \ddots & \vdots \\ & 0 & & g_{\deg g} \end{array}\right). \tag{3}$$

Since g is a monic polynomial (see Section 2.3), we can convert the i-th column $(\deg g + 1 \leq i \leq t+1)$ columns of this matrix (3) by integer linear combinations of columns as follows:

$$M_{\mathfrak{q}} = \left(\begin{array}{c|c} qI_{\deg g} & T_{\mathfrak{q}} \\ \hline 0 & I_{t-\deg g+1} \end{array}\right), \tag{4}$$

where $T_{\mathfrak{q}}$ is a $\deg g \times (t - \deg g + 1)$ integer matrix. Conversely, for any $c = (c_0, c_1, \ldots, c_t)^{\mathrm{T}} \in \mathbb{Z}^{t+1}$ the polynomial vector $a = M_{\mathfrak{q}}\, c$ satisfies the sufficient condition in (2). Therefore, for the matrix $T_{\mathfrak{q}}$ and $c \in \mathbb{Z}^{t+1}$, we can represent $M_{\mathfrak{q}}$ as follows:

$$\begin{aligned}&(a_0, a_1, \ldots, a_{\deg g - 1})^{\mathrm{T}} \\ &\quad = q(c_0, c_1, \ldots, c_{\deg g - 1})^{\mathrm{T}} + T_{\mathfrak{q}}(a_{\deg g}, a_{\deg g + 1}, \ldots, a_t)^{\mathrm{T}}.\end{aligned} \tag{5}$$

For $T_{\mathfrak{q}}$ and $(a_{\deg g}, a_{\deg_g + 1}, \ldots, a_t)^{\mathrm{T}} \in \mathbb{Z}^{t - \deg_g + 1}$, we set $(u_0, u_1, \ldots, u_{\deg g - 1}) = T_{\mathfrak{q}}\,(a_{\deg g}, a_{\deg_g + 1}, \ldots, a_t)^{\mathrm{T}}$. Then, we can search a that satisfies the sufficient condition in (2) by repeatedly adding $u_0, u_1, \ldots, u_{\deg g - 1}$ to q in the sieving region $\mathcal{H}_a$ for $(u_0, u_1, \ldots, u_{\deg g - 1}, a_{\deg g}, a_{\deg g + 1}, \ldots, a_t)$.

4. Proposed multi-dimensional lattice sieve

The lattice sieve tries to find candidates of hit tuples in the lattice whose elements are divisible by $\mathfrak{q} \in \mathcal{B}_i$ (called special-$\mathfrak{q}$). For a special-$\mathfrak{q} = (q, g) \in \mathcal{B}_i$, let $M_{\mathfrak{q}}$ be the matrix of equation (4), and let $M_{\mathfrak{q}}^{\mathrm{LLL}}$ be the matrix generated by the LLL algorithm [8] from $M_{\mathfrak{q}}$.

In this paper, we call the search space of dimension $t+1$ for hit tuple $a \in \mathcal{H}_a$ the a-space. On the other hand, the $(t+1)$-dimensional lattice $M_{\mathfrak{q}}^{\mathrm{LLL}}$, which is generated by $M_{\mathfrak{q}}^{\mathrm{LLL}}\, c$ for $c \in \mathbb{Z}^{t+1}$, is called the c-space. Moreover, for a sieving interval $H_c \in \mathbb{R}_{>0}$, we define the sieving region over the c-space by

$$\mathcal{H}_c(H_c) = \{(c_0, c_1, \ldots, c_t)^{\mathrm{T}} \in \mathbb{Z}^{t+1} \mid |c_i| \leq H_c\ (0 \leq i \leq t),\ c_t \geq 0\}.$$

The lattice sieve for the special-$\mathfrak{q}$ searches candidates of hit tuples in the sieving region $\mathcal{H}_c$ in the c-space.

Next, we construct the matrix $M_{\mathfrak{r}}$ from an element $\mathfrak{r} = (r, h) \in \mathcal{B}_i$ that is different from $\mathfrak{q}$ in the factor base. By the same method for generating $M_{\mathfrak{q}}$ from $\mathfrak{q}$, we can obtain equation (5) corresponding to $M_{\mathfrak{r}}$, and by reducing vector $r(c_0, c_1, \ldots, c_{\deg h - 1})^{\mathrm{T}}$ modulo r, we can

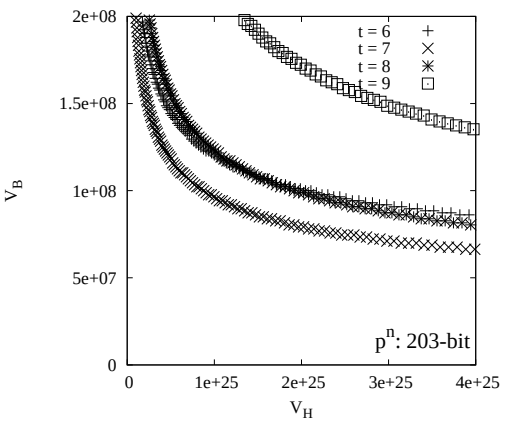

Fig. 1. Here V_H is the size of the sieving region and V_B is that of the factor bases of the multi-dimensional lattice sieve for the number field sieve over the extension field $\mathrm{GF}(p^{12})$ of 203 bits.

yield the following equation

$$(a_0, a_1, \ldots, a_{\deg h-1})^{\mathrm{T}} \equiv T_{\mathfrak{r}}(a_{\deg h}, a_{\deg h+1}, .. a_t)^{\mathrm{T}} \pmod{r}. \quad (6)$$

Here, we decompose the $(t+1)\times(t+1)$ matrix $M_{\mathfrak{q}}^{\mathrm{LLL}}$ into the $\deg h \times (t+1)$ matrix $M_{\mathfrak{q},1}^{\mathrm{LLL}}$ and the $(t-\deg h+1)\times(t+1)$ matrix $M_{\mathfrak{q},2}^{\mathrm{LLL}}$ as $M_{\mathfrak{q}}^{\mathrm{LLL}} = \begin{pmatrix} M_{\mathfrak{q},1}^{\mathrm{LLL}} \\ M_{\mathfrak{q},2}^{\mathrm{LLL}} \end{pmatrix}$. The set of all elements a divisible by $\mathfrak{q}$ is represented by $a = M_{\mathfrak{q}}^{\mathrm{LLL}} c$ for $c \in \mathbb{Z}^{t+1}$, namely

$$(a_0, a_1, \ldots, a_{\deg h-1})^{\mathrm{T}} = M_{\mathfrak{q},1}^{\mathrm{LLL}} c, \quad (7)$$

$$(a_{\deg h}, a_{\deg h+1}, \ldots, a_t)^{\mathrm{T}} = M_{\mathfrak{q},2}^{\mathrm{LLL}} c. \quad (8)$$

Therefore, from equations (6) and (8), we obtain

$$(M_{\mathfrak{q},1}^{\mathrm{LLL}} - T_{\mathfrak{r}}\, M_{\mathfrak{q},2}^{\mathrm{LLL}})\, c \equiv 0 \pmod{r}. \quad (9)$$

Next, let $M_{\mathfrak{q},\mathfrak{r}}$ be the lattice generated by c from equation (9), namely $M_{\mathfrak{q},\mathfrak{r}}$ is the kernel of the linear map $(M_{\mathfrak{q},1}^{\mathrm{LLL}} - T_{\mathfrak{r}}\, M_{\mathfrak{q},2}^{\mathrm{LLL}})$. Note that $a = M_{\mathfrak{q}}^{\mathrm{LLL}} M_{\mathfrak{q},\mathfrak{r}}\, e$ for any $e = (e_0, e_1, \ldots, e_t) \in \mathbb{Z}^{t+1}$ satisfies the sufficient condition in (2) for both $\mathfrak{q}$ and $\mathfrak{r}$. We can compute $M_{\mathfrak{q},\mathfrak{r}}$ from $M_{\mathfrak{q},1}^{\mathrm{LLL}} - T_{\mathfrak{r}}\, M_{\mathfrak{q},2}^{\mathrm{LLL}}$ corresponding to equation (9).

5. How to select parameters t, H, B_1, B_2

In this section, we explain how to select the parameters of the lattice sieve in Section 2.3 for given two polynomials f_1, f_2 in the polynomial selection in Section 2.2. In particular, we discuss suitable size of the dimension $t+1$, the sieving interval H, and the smoothness bounds B_1, B_2 that satisfy inequality (1) for the number field sieve over extension fields $\mathrm{GF}(p^{12})$. If we select the parameters that accelerate both the searching relation step and the linear algebra step simultaneously, then the total running time of the number field sieve becomes faster.

5.1 Selection of t

We denote by V_H the size of the sieving region $\mathcal{H}_a(H)$, namely $V_H = 2^t \prod_{j=0}^{t} H_j$. We extend the estimation of the average norm in the two-dimensional lattice sieve [9] to our multi-dimensional case. The average norm $N_{\mathrm{ave}}(h_a(\alpha_i))$ of the polynomial $f_i\,(i=1,2)$ in the lattice sieve of dimension $t+1$ is evaluated by the formula

$$N_{\mathrm{ave}}(h_a(\alpha_i)) = \frac{\sqrt{\int_0^{H_t}\int_{-H_{t-1}}^{H_{t-1}} \cdots \int_{-H_0}^{H_0} (\mathrm{Res}(h_a, f_i))^2\, da_0 \ldots da_t}}{V_H}.$$

Moreover, we approximate the probability $\rho(x,y)$ that the integers smaller than x are y-smooth to be $(\log_y x)^{-\log_y x}$, and we assume that the total size of the factor bases $\mathcal{B}_1, \mathcal{B}_2$ is $V_B = \pi(B_1) + \pi(B_2)$ where $\pi(B_i)$ is the number of primes smaller than or equal to $B_i\,(i=1,2)$. Let R be the number of hit tuples in the sieving region $\mathcal{H}_a(H)$. Then R is calculated by

$$R = \rho(N_{\mathrm{ave}}(h_a(\alpha_1)), B_1)\rho(N_{\mathrm{ave}}(h_a(\alpha_2)), B_2)V_H. \quad (10)$$

Here, we have to find parameters that satisfy (1), namely $R > V_B$. Fig. 1 shows the minimal V_B that satisfies $R > V_B$ for V_H in the lattice sieve of dimension $t+1$ in the extension field $\mathrm{GF}(p^{12})$ of 203 bits. In order to reduce the time of searching such a bound V_B, we set $H_0 = H_1 = \cdots = H_t$ and $B_1 = B_2$. From Fig. 1, we can select smaller sizes V_H of the sieving region and V_B of the factor bases that satisfy inequality (1) using dimension 8 for the extension field $\mathrm{GF}(p^{12})$ of 203 bits.

5.2 Selection of H and B_1, B_2

For fixed sizes V_H of the sieving region and V_B of the factor bases, we first select a sieving interval H and then smoothness bounds B_1, B_2. The sieving interval H is chosen so that the probability $\rho(N_{\mathrm{ave}}(h_a(\alpha_1)), B_1)\rho(N_{\mathrm{ave}}(h_a(\alpha_2)), B_2)$ of a hit tuple in equation (10) is maximum for fixed B_1, B_2 with $B_1 = B_2$. For the above H, we then select B_1, B_2 so that the number of hit tuples in equation (10) become maximum.

6. Our experiment on number field sieve over $\mathrm{GF}(p^{12})$

In this section, we report our experiment on solving the DLP over the extension field $\mathrm{GF}(p^{12})$ of 203 bits using the number field sieve in Section 2. We chose the characteristic $p = 122663$ of 17 bits, namely the cardinality of the extension field $\mathrm{GF}(p^{12})$ is

$$p^{12} = 116028047901493489912893641612 4 \backslash$$
$$526007290958514026649130779408 1.$$

The computational environment in our experiment is as follows. We used one PC equipped with four CPUs (Intel Xeon X7350 2.93 GHz; Core2 microarchitecture; 16 cores in total) and 64 GBytes of RAM. We utilize gmp-5.0.5 for the arithmetic of multi-precision integers, openmpi-1.6 for parallel implementation between processes, pari-2.5.1 for the decomposition of ideals in the number fields, and ntl-5.5.2 for the computation of lattice reduction using the LLL algorithm. We use C++ with compiler gcc-4.7.1 on Linux OS (64 bits).

Table 1 presents the experimental data in our implementation and the previous ones of the number field sieve over extension fields $\mathrm{GF}(p^n)$.

Table 1. Comparison of known experiments of the number field sieve over extension field $\mathrm{GF}(p^n)$.

Finite Field	$\mathrm{GF}(p^3)$	$\mathrm{GF}(p^6)$	$\mathrm{GF}(p^{12})$
Authors	Joux et al. [4]	Zajac [5]	This paper
Year	2006	2008	2012
CPU	Alpha (1.15GHz) × 8	Sempron (2.01GHz) × 8	Xeon (2.93GHz) × 4
Days	19 days	5 days	2 days
Bit Length	394	242	203
Sieving	2-dim. lattice sieve	3-dim. line sieve	**7-dim. lattice sieve**

6.1 Polynomial selection

In order to select two polynomials f_1, f_2 in Section 2.2, we use the polynomial selection similar to the previous experiments [4] and [5]. At first, an irreducible polynomial $f_1 \in \mathbb{Z}[X]$ of degree 12 with small coefficients is chosen, and then we set $f_2 = f_1 + p$ or $f_2 = f_1 - p$.

In this paper, Murphy's α function [9] is used for selecting a more suitable pair of polynomials f_1, f_2. If Murphy's α function $f_i\,(i = 1, 2)$ is smaller, then the norm $N(h_a(\alpha_i))\,(i = 1, 2)$ is expected to become smoother, namely it is divisible by small prime divisors with higher probability. The coefficients of the polynomial f_1 are searched in the range of ± 10, and then the sum of Murphy's α of the following polynomials f_1, f_2 is the smallest among the range of our search: $f_1(X) = X^{12} - 3X^4 + 9X^3 - 9X^2 - 9X + 2$, $f_2(X) = X^{12} - 3X^4 + 9X^3 - 9X^2 - 9X - 122661$.

6.2 Searching relations

In the estimation of Section 5.1, the suitable dimension of the lattice sieve for the extension field $\mathrm{GF}(p^{12})$ of 203 bits was estimated to be eight. We perform some experiments of the lattice sieve of dimensions 6, 7 and 8 for a random special-$\mathfrak{q}$ with fixed V_H and V_B. From these experiments, the lattice sieve of dimension 7 yields the largest number of hit tuples for one special-$\mathfrak{q}$, and then we select $H = (443, 427, 304, 140, 70, 24, 9)$ and smoothness bounds $B_1 = 114547$ and, $B_2 = 148859$.

We run the lattice sieve using the above polynomials f_1, f_2 and the above parameters t, H, B_1, B_2. Our experiment has generated 32,241 hit tuples in about 42 hours using only 6 cores in our computational environment. This is about 1.3 times larger than the sufficient number $\sharp\mathcal{B}_1 + \sharp\mathcal{B}_2 + 2n$ of hit tuples.

6.3 Linear algebra

From the hit tuples in the searching relations, we construct a matrix of linear equations modulo $\ell = 6118607636866573789$ (63 bits) that is the maximum prime divisor of $p^{12} - 1$. The size of the matrix is 32241×24463, and it is shrunk to 16579×15073 by the filter process such as eliminating duplicated hit tuples. Then, we solve it by the Lanczos method.

We found the solutions of the linear equations in about 25 minutes using the 16 cores in our computational environment, and the logarithms of $\mathfrak{q} \in \mathcal{B}_i$ was obtained.

Finally, we present an example of the discrete logarithm. Let $\gamma = x^2 + x - 7$ be a generator of $\mathrm{GF}(p^{12})^* = (\mathrm{GF}(p)[X]/f_1(X))^*$. Let $\delta = x^2 - 5x + 7$ be a target element of solving the discrete logarithm $\log_\gamma \delta$ in $\mathrm{GF}(p^{12})$. Note that both γ and δ are B_1-smooth. The above linear equations modulo ℓ yields $\log \delta = 3540036734608022534$ and $\log \gamma = 3897708711757659596$, and thus the discrete logarithm $\log_g \delta$ in $\mathrm{GF}(p^{12})$ is computed by $\log \delta / \log \gamma = 3161374319443177763 \bmod \ell$.

7. Conclusion

In this paper, we presented an implementation of the number field sieve for solving the DLP over extension fields $\mathrm{GF}(p^n)$ that underpinned the security of pairing-based cryptography. Especially, we proposed an implementation of the lattice sieve of dimension higher than two. In our experiment, we discussed the dimension and the size of the sieving region suitable for the number field sieve over extension fields $\mathrm{GF}(p^{12})$. Finally, we have solved the DLP over an extension field $\mathrm{GF}(p^{12})$ of 203 bits using a PC of 16 CPU core in about 43 hours.

In the future, we discuss how to select the sieving region for the DLP over extension fields $\mathrm{GF}(p^{12})$ of larger bits. We also extend the efficient lattice sieve of dimension two to the lattice sieve of dimension higher than two.

References

[1] F. Vercauteren, Optimal pairings, IEEE Trans. Inform. Theory, **56** (2010), 455–461.

[2] P. S. L. M. Barreto and M. Naehrig, Pairing-friendly elliptic curves of prime order, in: Proc. of SAC 2005, LNCS, Vol. 3897, pp. 319–331, Springer-Verlag, Berlin, 2006.

[3] A. Joux and R. Lercier, Improvements to the general number field sieve for discrete logarithms in prime fields. A comparison with the Gaussian integer method, Math. Comput., **72** (2003), 953–967.

[4] A. Joux, R. Lercier, N. P. Smart and F. Vercauteren, The number field sieve in the medium prime case, in: Proc. of CRYPTO 2006, LNCS, Vol. 4117, pp. 326–344, Springer-Verlag, Berlin, 2006.

[5] P. Zajac, Discrete logarithm problem in degree six finite fields, Ph.D. thesis, Slovak Univ. of Technology, 2008.

[6] P. Zajac, On the use of the lattice sieve in the 3D NFS, Tatra Mt. Math. Publ., **45** (2010), 161–172.

[7] J. M. Pollard, The lattice sieve, in: Lecture Notes in Math., A. K. Lenstra and H. W. Lenstra eds., Vol. 1554, pp. 43–49, Springer-Verlag, Berlin, 1993.

[8] A. K. Lenstra, H. W. Lenstra and L. Lovász, Factoring polynomials with rational coefficients, Math. Ann., **261** (1982), 515–534.

[9] B. Murphy, Polynomial selection for the number field sieve integer factorisation algorithm, Ph.D. thesis, The Australian National Univ., 1999.