Memoirs of the Faculty of Engineering, Okayama University, Vol. 43, pp. 108-112, January 2009

Extension Field for Xate Pairing with Freeman Curve

Kenta NEKADO*
Graduate School of Natural Science and Technology, Okayama University
3-1-1, Tsushima-naka, Okayama, Okayama 700-8530, Japan

Hidehiro KATO*
Graduate School of Natural Science and Technology, Okayama University
3-1-1, Tsushima-naka, Okayama, Okayama 700-8530, Japan

Yasuyuki NOGAMI*
Graduate School of Natural Science and Technology, Okayama University
3-1-1, Tsushima-naka, Okayama, Okayama 700-8530, Japan

Yoshitaka MORIKAWA*
Graduate School of Natural Science and Technology, Okayama University
3-1-1, Tsushima-naka, Okayama, Okayama 700-8530, Japan

(Received December 9, 2008)

Recently, pairing-based cryptographies such as ID-based cryptography and group signature have been studied. For fast pairing calculation, not only pairing algorithms but also arithmetic operations in extension field must be efficiently carried out. The authors show efficient arithmetic operations of extension field for Xate pairing especially with Freeman curve.

1 INTRODUCTION

In recent years, pairing-based cryptographies such as ID-based cryptography [1] and group signature [2] have been studied. For their implementations, pairings such as Weil pairing [1], Tate pairing, Ate pairing [3] and Xate pairing [4] have been used. In order to implement these pairings, several kinds of ordinary pairing-friendly curves such as Miyaji-Nakabayashi-Takano (MNT) curve [5], Barreto-Naehrig (BN) curve [6] and Freeman curve [7, 8] have been proposed. As the definition field of these curves, many researchers use optimal extension field (OEF) [9] because OEF carries out arithmetic operations efficiently. However, it is known that OEF is not available for the definition field of Freeman curve due to the condition of OEF. Our previous work namely Type I-X all one polynomial field (AOPF) [11] is available for the definition field of Freeman curve. Type I-X AOPF can carry out arithmetic operations as efficient as OEF.

In this paper, the authors consider how to constructed type I-X AOPF and optimize a multiplication algorithm for Xate pairing with Freeman curve. Additionally, this paper shows some experimental results of Xate pairing with Freeman curve defined over the type I-X AOPF. Then, it is shown that the proposed method works approximately 10 percent faster than the conventional method.

Notation : $\mathbb{F}_p$, $\mathbb{F}_{p^m}$, and $\mathbb{F}^*_{p^m}$ denote a prime field, m-th extension field over $\mathbb{F}_{p^m}$, and the multiplicative group in $\mathbb{F}_{p^m}$. For two integers m and n, $m|n$ means that m divides n. $E(\mathbb{F}_{p^m})$ denotes the elliptic curve defined over $\mathbb{F}_{p^m}$.

2 XATE PAIRING WITH FREEMAN CURVE

This section briefly reviews Ate pairing, Freeman curve, and Xate pairing.

2.1 Ate Pairing

The smallest positive integer d such that $r|(p^d-1)$ is called *embedding degree*, then let $\mathbb{G}_1$ and $\mathbb{G}_2$ be

$$\mathbb{G}_1 = E(\mathbb{F}_{p^d})[r] \cap \mathrm{Ker}(\phi - [1]), \tag{1a}$$
$$\mathbb{G}_2 = E(\mathbb{F}_{p^d})[r] \cap \mathrm{Ker}(\phi - [p]), \tag{1b}$$

where $E(\mathbb{F}_{p^d})[r]$ denotes the set of rational points of order r in $E(\mathbb{F}_{p^d})$. Let $P \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$, Ate pairing e is defined as

$$e : \begin{cases} \mathbb{G}_2 \times \mathbb{G}_1 & \to \mathbb{F}^*_{p^d}/(\mathbb{F}^*_{p^d})^r, \\ (Q,P) & \mapsto f_{T,Q}(P)^{(p^d-1)/r}. \end{cases} \tag{2}$$

*E-mail: { nekado, kato, nogami, morikawa }@trans.cne.okayama-u.ac.jp

It gives a non-degenerate and bilinear map. The parameter T is given as

$$T = (t-1)^i \mod r \quad (1 \le i < d), \tag{3}$$

where t is *Frobenius trace*. Ate pairing consists of two steps, one is $f_{T,Q}(P)$ calculation by Miller's algorithm, and the other is the calculation called *final exponentiation* that $f_{T,Q}(P)$ is raised to the $((p^d-1)/r)$-th power. The number of iterations in Miller's algorithm of Ate pairing is given by $\lfloor \log_2(T) \rfloor$.

Devegili et al. have improved Ate pairing by using subfield-twisted curve [6]. However, this technique can not be applied to Ate pairing with Freeman curve because in order to apply this technique we need to use OEF as the definition field. On the other hand, Nogami et al. have also improved Ate pairing by using the similar technique [10]. It is called *cross-twisted* Ate (Xt-Ate) paring and can be used even in the case of Freeman curve.

2.2 Freeman Curve

Freeman curve is a class of ordinary pairing-friendly curves of embedding degree $d = 10$ [7, 8]. The parameters of Freeman curve $E(\mathbb{F}_p)$ are given as

$$p(\chi) = 25\chi^4 + 25\chi^3 + 25\chi^2 + 10\chi + 3, \tag{4a}$$
$$r(\chi) = 25\chi^4 + 25\chi^3 + 15\chi^2 + 5\chi + 1, \tag{4b}$$
$$T(\chi) = (t(\chi)-1)^2 = 5\chi^2, \tag{4c}$$

where χ is an integer such that $p(\chi)$ becomes a prime number. Xt-Ate pairing with Freeman curve $E(\mathbb{F}_{p^{10}})$ uses its *quadratic twisted* curve $E'(\mathbb{F}_{p^5})$. Thus, we need to prepare subfield $\mathbb{F}_{p^5}$ besides the definition field $\mathbb{F}_{p^{10}}$.

2.3 Xate Pairing

Nogami et al. have proposed *integer variable χ-based* Ate (Xate) pairing [4]. In the case of Freeman curve, Xate paring e' is given as

$$e' : \begin{cases} \mathbb{G}_2 \times \mathbb{G}_1 & \to \ \mathbb{F}^*_{p^d}/(\mathbb{F}^*_{p^d})^r, \\ (Q,P) & \mapsto \ \hat{f}_{\chi,Q}(P)^{(p^d-1)/r}. \end{cases} \tag{5}$$

It also gives a non-degenerate and bilinear map. $\hat{f}_{\chi,Q}(P)$ can be calculated by Miller's algorithm, and the number of iterations becomes only $\lfloor \log_2(\chi) \rfloor$. Moreover, we can apply Xt-Ate paring technique to Xate pairing, namely Xt-Xate pairing.

3 EXTENSION FIELD FOR FREEMAN CURVE

Bailey et al. have proposed optimal extension field (OEF) [9]. It needs to satisfy the condition that each prime factor of m divides $p-1$, for example. OEF carries out arithmetic operations efficiently. However, according to Eq.(4a), OEF technique can not be applied for constructing $\mathbb{F}_{p^5}$ of Eq.(4a) because 5 does not divides $p(\chi)-1$.

The authors have proposed type I-X all one polynomial field (AOPF) [11]. Type I-X AOPF technique can be applied for constructing $\mathbb{F}_{p^5}$ of Eq.(4a) because it has been proven that type I-X AOPF is prepared for every pair of characteristic p and extension degree m when $p > m$ [11].

First, we review type I-X AOPF and then consider how to construct type I-X AOPF for Freeman curve.

3.1 Type-$\langle k,m \rangle$ Gauss Period Normal Basis

Type I-X AOPF $\mathbb{F}_{(p^n)^m}$ is constructed by m-th towering over $\mathbb{F}_{p^n}$ with type-$\langle k,m \rangle$ Gauss period normal basis (GNB) [12] when $\gcd(m,n) = 1$. Type-$\langle k,m \rangle$ GNB is defined with a certain integer k as follows.

Define 1 : Let $km+1$ be a prime number not equal to p and suppose that $\gcd(km/e, m) = 1$, where e is the order of p modulo $km+1$. Then, for any primitive k-th root θ of unity in $\mathbb{F}_{km+1}$,

$$\gamma = \sum_{i=0}^{k-1} \beta^{\theta^i} \tag{6}$$

generates a normal basis $\{\gamma, \gamma^p, \cdots, \gamma^{p^{m-1}}\}$ in $\mathbb{F}_{p^m}$, where β is a $(km+1)$-st root of unity that belongs to $\mathbb{F}_{p^e}$. This normal basis is called type-$\langle k,m \rangle$ GNB. ■

Type-$\langle k,m \rangle$ GNB is available when $4p$ does not divide $m(p-1)$ [13]. Therefore, type I-X AOPF $\mathbb{F}_{(p^n)^m}$ can be constructed for every pair of characteristic p and extension degree m when $p > m$, for instance.

3.2 Cyclic Vector Multiplication Algorithm

As an efficient multiplication algorithm in type I-X AOPF, the authors have proposed *cyclic vector multiplication algorithm* (CVMA) [13]. **Fig**.1 shows CVMA in $\mathbb{F}_{(p^n)^m}$.

In the algorithm of **Fig**.1, $q[0]$ becomes 0 when k is even [13]. Then, the calculation cost of CVMA is explictly given as follows. Note that A_n and M_n denote the computational costs of an addition and a multiplication in $\mathbb{F}_{p^n}$, respectively.

$$M_{mn} = \frac{m(m+1)}{2} M_n + \begin{cases} \left(\left(\frac{m(m-1)(k+2)}{2} - 1 \right) + k - 1 + m \right) A_n & \text{when } k \text{ is odd}, \\ \left(\frac{m(m-1)(k+2)}{2} \right) A_n & \text{when } k \text{ is even}. \end{cases} \tag{7}$$

As shown in Eq.(7), CVMA needs more additions in $\mathbb{F}_{p^n}$ as k becomes larger. Usually, A_n is much smaller than M_n. However, if the number of additions in $\mathbb{F}_{p^n}$ is much more than that of multiplications in $\mathbb{F}_{p^n}$, it will not be negligible.

Input: $X = \sum_{i=0}^{m-1} x_i \gamma^{p^i},\ Y = \sum_{i=0}^{m-1} y_i \gamma^{p^i} \quad (x_i, y_i \in \mathbb{F}_{p^n})$.

Output: $Z = XY = \sum_{i=0}^{m-1} z_i \gamma^{p^i} \quad (z_i \in \mathbb{F}_{p^n})$.

Preparation:

1. Determine k that satisfies the conditions in **Def**.1.
2. For $0 \le i \le m,\ q[i] \leftarrow 0$.
3. For $0 \le t < m$ and $0 \le h < k,\ g[\langle p^{t+hm} \rangle] \leftarrow t+1$.
4. $g[0] \leftarrow 0$.

Procedure:

1. For $0 \le i < m,\ q[i+1] \leftarrow x_t y_t$.
2. For $0 \le i < j \le m-1$, {
3. $\quad R_{ij} \leftarrow (x_i - x_j)(y_i - y_j)$,
4. $\quad$ For $0 \le h \le k-1$, {
5. $\quad\quad q\left[g[\langle p^i + p^{j+hm} \rangle]\right] \leftarrow q\left[g[\langle p^i + p^{j+hm} \rangle]\right] + R_{ij}$.
6. $\quad$ }
7. }
8. For $0 \le i < m,\ z_i \leftarrow kq[0] - q[i+1]$.

(End of algorithm)

† $\langle x \rangle$ means x mod $km+1$.

Figure 1: CVMA in $\mathbb{F}_{(p^n)^m}$

3.3 Itoh-Tsujii Algorithm

As an inversion algorithm in type I-X AOPF $\mathbb{F}_{(p^n)^m}$, Itoh-Tsujii algorithm (ITA) [14] that uses Frobenius map is available. Consider a non-zero element X that belongs to $\mathbb{F}_{(p^n)^m}$, its inverse element is given as

$$X^{-1} = \frac{X^{p^n} \cdots X^{(p^n)^{m-1}}}{X X^{p^n} \cdots X^{(p^n)^{m-1}}}. \tag{8}$$

Then, $X X^{p^n} \cdots X^{(p^n)^{m-1}}$ becomes a non-zero element that belongs to $\mathbb{F}_{p^n}$ because it is the norm of X with respect to $\mathbb{F}_{p^n}$. In the case of type I-X AOPF, Frobenius map $X \to X^{p^n}$ dose not need any algebraic calculations such as additions and multiplications because the coefficients of X^{p^n} is just a cyclic shift of the coefficients of X [12].

4 TYPE I-X AOPF FOR FREEMAN CURVE

Freeman has shown four kinds of pairing-friendly curves [7, 8]. Their characteristic p's are given as follows.

$$p = 5031898990973855325986159485679754327409672 03 \quad (149-\text{bit}) \tag{9a}$$

$$p = 6109996327108312874607376956794487035427061646150914794603 \quad (196-\text{bit}) \tag{9b}$$

$$p = 1821165080396947206449326434737595004593425469665709042072623004320 3803 \quad (234-\text{bit}) \tag{9c}$$

$$p = 6462310997348816962203124910505252082673338846966431201635262694402825461643 \quad (252-\text{bit}) \tag{9d}$$

In these cases, each minimal k such that type I-X AOPF $\mathbb{F}_{p^5}$ can be constructed is given as **Table** 1, and type I-X AOPF $\mathbb{F}_{(p^5)^2}$ can be prepared by 2nd towering over $\mathbb{F}_{p^5}$ with type-$\langle 2,2 \rangle$ GNB.

Table 1: The minimal k

p	Eq.(9a)	Eq.(9b)	Eq.(9c)	Eq.(9d)
k	2	6	8	6

In order to make pairing-based cryptography practical, the characteristic p needs to be larger than or equal to 160-bit. The k of Eqs.(9b), (9c), or (9d) that satisfies $p \ge 160$ are larger than or equal to 6 as shown **Table** 1.

For example, with type-$\langle 6,5 \rangle$ GNB as **Fig**.2, the computation amount of a multiplication in $\mathbb{F}_{p^5}$ is given as

$$M_5 = 15M_1 + 80A_1. \tag{10}$$

If we can apply type-$\langle 2,5 \rangle$ GNB, the computation amount of a multiplication in $\mathbb{F}_{p^5}$ is given as

$$M_5 = 15M_1 + 40A_1. \tag{11}$$

Thus, a multiplication in $\mathbb{F}_{p^5}$ with type-$\langle 6,5 \rangle$ GNB needs twice as many additions in $\mathbb{F}_p$ as that with type-$\langle 2,5 \rangle$ GNB. In what follows, we consider how to decrease the number of additions in $\mathbb{F}_p$ of a multiplication in $\mathbb{F}_{p^5}$ with type-$\langle 6,5 \rangle$ GNB.

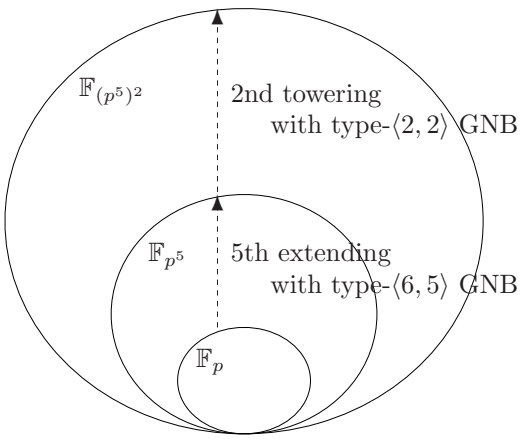

Figure 2: Type I-X AOPF $\mathbb{F}_{(p^5)^2}$

4.1 Improvement Of CVMA For Freeman Curve

For example, in type I-X AOPF $\mathbb{F}_{p^5}$ of Eq.(9b), CVMA needs to calculate the following equations.

$$q[0] = 0, \tag{12a}$$
$$q[1] = x_0y_0 + 2R_{01} + R_{03} + R_{12} + 2R_{13} + R_{14} + 2R_{23} + 2R_{24} + R_{34}, \tag{12b}$$
$$q[2] = x_1y_1 + R_{02} + 2R_{03} + R_{04} + 2R_{12} + R_{14} + R_{23} + 2R_{24} + 2R_{34}, \tag{12c}$$
$$q[3] = x_2y_2 + R_{01} + R_{02} + 2R_{03} + 2R_{04} + R_{13} + 2R_{14} + 2R_{23} + R_{34}, \tag{12d}$$
$$q[4] = x_3y_3 + 2R_{01} + 2R_{02} + R_{04} + R_{12} + R_{13} + 2R_{14} + R_{24} + 2R_{34}, \tag{12e}$$
$$q[5] = x_4y_4 + R_{01} + 2R_{02} + R_{03} + 2R_{04} + 2R_{12} + 2R_{13} + R_{23} + R_{24}. \tag{12f}$$

Eqs.(12) except for x_iy_i can be expressed with the matrix shown as Eq.(13). In this matrix, the lows correspond to $q[1]$, $q[2]$, $q[3]$, $q[4]$ and $q[5]$, and the columns correspond to the coefficients of R_{01}, R_{02}, R_{03}, R_{04}, R_{12}, ..., R_{24} and R_{34}.

$$\begin{pmatrix} 2 & 0 & 1 & 0 & 1 & 2 & 1 & 2 & 2 & 1 \\ 0 & 1 & 2 & 1 & 2 & 0 & 1 & 1 & 2 & 2 \\ 1 & 1 & 2 & 2 & 0 & 1 & 2 & 2 & 0 & 1 \\ 2 & 2 & 0 & 1 & 1 & 1 & 2 & 0 & 1 & 2 \\ 1 & 2 & 1 & 2 & 2 & 2 & 0 & 1 & 1 & 0 \end{pmatrix} \tag{13}$$

Then, it can be decomposed as

$$\begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} + \begin{pmatrix} 2 & 0 & 0 & 0 & 0 & 2 & 0 & 2 & 2 & 0 \\ 0 & 0 & 2 & 0 & 2 & 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 2 & 2 & 0 & 0 & 2 & 2 & 0 & 0 \\ 2 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 2 \\ 0 & 2 & 0 & 2 & 2 & 2 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{14}$$

Then, consider the following components C_j whose suffixes j means the column vector of the former matrix of Eq.(14).

$$C_{00101} = R_{01} + 2R_{04}, \quad C_{01100} = R_{02} + 2R_{03}, \tag{15a}$$
$$C_{10001} = R_{03} + 2R_{13}, \quad C_{01010} = R_{04} + 2R_{34}, \tag{15b}$$
$$C_{10010} = R_{12} + 2R_{01}, \quad C_{00110} = R_{13} + 2R_{14}, \tag{15c}$$
$$C_{11000} = R_{14} + 2R_{24}, \quad C_{01001} = R_{23} + 2R_{12}, \tag{15d}$$
$$C_{00011} = R_{24} + 2R_{02}, \quad C_{10100} = R_{34} + 2R_{23}. \tag{15e}$$

Then, Eq.(12) can be recomposed as

$$q[0] = 0, \tag{16a}$$
$$q[1] = x_0y_0 + C_{11000} + C_{10100} + C_{10010} + C_{10001}, \tag{16b}$$
$$q[2] = x_1y_1 + C_{11000} + C_{01100} + C_{01010} + C_{01001}, \tag{16c}$$
$$q[3] = x_2y_2 + C_{10100} + C_{01100} + C_{00110} + C_{00101}, \tag{16d}$$
$$q[4] = x_3y_3 + C_{10010} + C_{01010} + C_{00110} + C_{00011}, \tag{16e}$$
$$q[5] = x_4y_4 + C_{10001} + C_{01001} + C_{00101} + C_{00011}. \tag{16f}$$

In this case, the computation amount of a multiplication in $\mathbb{F}_{p^5}$ is given as follows. Note that D_n denotes the computational cost of a doubling in $\mathbb{F}_{p^n}$.

$$M_5 = 15M_1 + 50A_1 + 10D_1. \tag{17}$$

Table 2 shows the computation amounts of a multiplication in each $\mathbb{F}_{p^5}$ and $\mathbb{F}_{p^{10}}$.

Table 2: The computation amounts of a multiplication

	with original CVMA	with improved CVMA
$\mathbb{F}_{p^5}$	(15, 80, 0) †	(15, 50, 10) †
$\mathbb{F}_{(p^5)^2}$	(45, 260, 0) †	(45, 185, 30) †

† For example, (15, 50, 10) denotes $15M_1 + 50A_1 + 10D_1$.

4.2 Reduction Of Modulo p Operations

A multiplication in extension field needs a lot of multiplications in prime field $\mathbb{F}_p$, thus we need a lot of modulo p operations. However, the calculation time of modulo p operation is much larger than that of other operations such as addition and multiplication. If we have enough memory, we do not need to carry out modulo p operation for every multiplication in $\mathbb{F}_p$. Therefore, the authors carried out modulo p operation only at step 8.

4.3 Improvement Of The Inversion Algorithm

As previously introduced, we use ITA as the inversion algorithm in type I-X AOPF. In ITA, we calculate an inversion in $\mathbb{F}_{(p^5)^2}$ with a norm in $\mathbb{F}_p$. In general, we directly calculate this norm. However, by calculating this norm after the calculation of the norm with respect to the subfield $\mathbb{F}_{p^5}$, we can carry out an inversion more efficiently. **Table** 3 shows the computation amounts of a inversion in each $\mathbb{F}_{p^5}$ and $\mathbb{F}_{(p^5)^2}$ with the subfield. Note that I_n means the computational cost of an inversion in $\mathbb{F}_{p^n}$.

Table 3: The computation amounts of an inversion

	with original CVMA	with improved CVMA
$\mathbb{F}_{p^5}$	(44, 188, 0, 1) †	(44, 124, 24, 1) †
$\mathbb{F}_{(p^5)^2}$	(104, 523, 0, 1) †	(104, 339, 64, 1) †

† For example, (44, 124, 24, 1) denotes $44M_1+114A_1+24D_1+I_1$.

5 SIMULATION

This section shows the implementation result of the proposed method. In this implementation, we use characteristic p of Eq.(9b). **Table** 4 shows the calculation timings of a multiplication and an inversion in each $\mathbb{F}_{p^5}$ and $\mathbb{F}_{(p^5)^2}$ with the computational environment **Table** 5. **Table** 6 shows the implementation result of Xt-Xate pairing with the extension field proposed in this paper.

Table 4: Timing of each operation

		with original CVMA	with improved CVMA
$\mathbb{F}_{p^5}$	mul	8.03 μs	7.31 μs
	inv	30.0 μs	28.4 μs
$\mathbb{F}_{(p^5)^2}$	mul	21.6 μs	19.2 μs
	inv	59.4 μs	55.5 μs

Table 5: Computational environment

CPU	Pentium4 3.00GHz
Cache Size	512 KB
OS	Linux 2.6.27
Language	C
Compiler	gcc 4.3.2
Library	GNU MP 4.2.4 [15]

Table 6: Timing of Xt-Xate pairing

	with original CVMA	with improved CVMA
Miller's algorithm	7.74 ms	6.94 ms
Final exponentiation	4.28 ms	3.79 ms
Total	12.0 ms	10.7 ms

Table 6 shows that the extension field $\mathbb{F}_{p^5}$ and $\mathbb{F}_{(p^5)^2}$ proposed in this paper are more efficient for Xt-Xate pairing with Freeman.

6 CONCLUSION

In this paper, we have considered how to constructed type I-X AOPF and optimized CVMA for Ate pairing with Freeman curve. Additionally, we showed the implementation result of Xt-Xate pairing with Freeman curve defined over type I-X AOPF. Then, it was shown that the proposed method works approximately 10 percent faster than the conventional method.

ACKNOWLEDGMENT

This research is supported by "Strategic Information and Communications R&D Promotion Programme (SCOPE)" from Ministry of Internal Affairs and Communications (MIC), Japan.

We thank A. J. Devegili for reading our paper and notifying some typos.

REFERENCES

[1] R. Sakai, K. Ohgishi and M. Kasahara: SCIS 2000, Jun. 2000, Okinawa, 26-28.

[2] D. Boneh, X. Boyan and H. Shacham: Proc. of Crypto2004, Lect. Notes Comput. Sci., 3152 (2004), 41-55.

[3] H. Hess, N. P. Smart and F. Vercauteren: IEEE Trans. Inf. Theory, 52 (2006), 4595-4602 .

[4] Y.Nogami, M.Akane, Y.Sakemi, H.Kato and Y.Morikawa: Pairing 2008, LNCS, 5209 (Aug. 2008), 178-191.

[5] A. Miyaji, M. Nakabayashi and S. Takano: IEICE Trans. Fundam., E84-A(5) (2001), 1234-1243.

[6] A. J. Devegili, M. Scott and R. Dahab: LNCS, 4575 (2007), 197-207.

[7] D. Freeman: ePrint (2006).

[8] D. Freeman: LNCS, Springer Berlin/Heidelberg, 4076 (Oct. 2006), 452-465.

[9] H. Cohen and G. Frey: Chapman & Hall/CRC (2005).

[10] Y.Nogami, M.Akane, Y.Sakemi and Y.Morikawa: ICCIT2008, 2 (Nov. 2008), Busan, 430-439.

[11] T. Yoshida, H. Kato, Y. Nogami and Y. Morikawa: The 2nd Joint Workshop on Information Security, Aug. 2007, Tokyo, Japan, 469-483.

[12] S. Gao: Doctoral thesis, 1993, Waterloo, Ontario, Canada.

[13] Hidehiro Kato, Yasuyuki Nogami, Tomoki Yoshida and Yoshitaka Morikawa: ETRI Journal, 29-6 (Dec. 2007).

[14] T. Itoh and S. Tsujii: Inf. and Comp., 78 (1988), 171-177.

[15] GNU Multiple Precision Arithmetic Library, available at "http://gmplib.org".