Extension Field for Ate Pairing with Freeman Curve

Kenta Nekado¹, Hidehiro Kato¹, Masataka Akane¹,
Yasuuki Nogami² and Yoshitaka Morikawa¹

¹Graduate School of Natural Science and Technology, Okayama University
3-1-1, Tsushima-naka, Okayama, Okayama 700-8530, Japan
E-mail: { nekado, kato, akane, nogami, morikawa }@trans.cse.okayama-u.ac.jp

Abstract: 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 consider efficient arithmetic operations of extension field for Ate 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 implementation of these cryptographies, pairings such as Weil pairing [1], Tate pairing [1], and Ate pairing [3] have been used. In order to implement these pairings, several kinds of pairing-friendly curves such as Miyaji-Nakabayashi-Takano (MNT) curve [4], Barreto-Naehrig (BN) curve [5] and Freeman curve [6] have been proposed. As the definition field of these curves, many researchers use optimal extension field (OEF) [7] because OEF carries out arithmetic operations efficiently. However, OEF cannot be the definition field of Freeman curve due to the condition of OEF.

Type I-X all polynomial field (AOPF) [8] which the authors have proposed can be the definition field of Freeman curve. Type I-X AOPF technique can carry out arithmetic operations as efficient as OEF. In this paper, the authors consider how to construct type I-X AOPF and optimize a multiplication algorithm for Ate pairing with Freeman curve. 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 $\mathbb{F}_p$ and the multiplicative group in $\mathbb{F}_p$. For two integers $m$ and $n$, $m|n$ means that $m$ divides $n$. $E(\mathbb{F}_{p^m})$ denotes the elliptic curve over $\mathbb{F}_{p^m}$.

2. Ate pairing with Freeman curve

This section briefly reviews Ate pairing and Freeman curve.

2.1 Ate pairing

The smallest positive integer $d$ such that $r|(p^d - 1)$ is called embedding degree, then Ate pairing $\epsilon$ is defined as

$$G_1 = E[r](\mathbb{F}_{p^d}) \cap \text{Ker}(\phi - [1]),$$
$$G_2 = E[r](\mathbb{F}_{p^d}) \cap \text{Ker}(\phi - [p]),$$
$$\epsilon : G_1 \times G_2 \rightarrow G_3 = \mathbb{F}_{p^d}^*/(E(\mathbb{F}_{p^d})^d),$$

where $E[r](\mathbb{F}_{p^d})$ denotes the set of rational points of order $r$ in $E(\mathbb{F}_{p^d})$. It gives a non-degenerate and bilinear map.

2.2 Freeman curve

Freeman curve is a class of ordinary pairing-friendly curves of embedding degree $d = 10$ [6]. Characteristic $p$ of Freeman curve $E(\mathbb{F}_p)$ is given as

$$p(\chi) = 25\chi^4 + 25\chi^3 + 25\chi^2 + 10\chi + 3,$$

where $\chi$ is an integer such that $p(\chi)$ becomes a prime number. Moreover, Ate pairing with Freeman curve $E(\mathbb{F}_{p^{10}})$ becomes more efficient by using its quadratic twisted curve $E'(\mathbb{F}_{p^5})$. Thus, we need to prepare subfield $\mathbb{F}_{p^5}$ besides an extension field $\mathbb{F}_{p^{10}}$.

3. Extension field for Freeman curve

Bailey et al. have proposed optimal extension field (OEF) [7]. 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.(2), OEF technique cannot prepare $\mathbb{F}_{p^5}$ of Eq.(2) because 5 does not divides $p(\chi) - 1$ or $p(\chi) + 1$.

The authors have proposed type I-X all one polynomial field (AOPF) [8]. Type I-X AOPF technique can prepare $\mathbb{F}_{p^5}$ of Eq.(2) because it has been proven that type I-X AOPF is prepared for every pair of characteristic $p$ and extension degree $m$ when $p > m$ [8].

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

3.1 Type-(k, m) Gauss period normal basis

Type I-X AOPF $\mathbb{F}_{(p^m)}$ is constructed by $m$-th tower over $\mathbb{F}_p$ with type-(k, m) Gauss period normal basis (GNB) [9] when $gcd(m, n) = 1$. Type-(k, m) GNB is defined with a certain integer $k$ as follows.

Define 1: Let $km + 1$ be a primitive 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 of unity in $\mathbb{F}_{km+1}$, $\gamma = \sum_{i=0}^{k-1} \beta^{p^i}$ generates a normal basis $\{\gamma, \gamma^p, \cdots, \gamma^{p^{m-1}}\}$ in $\mathbb{F}_{p^m}$, where $\beta$ is a $(km + 1)$-st root of unity that belongs to $\mathbb{F}_{p^m}$. This normal basis is called type-(k, m) GNB.
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) [10]. Fig. 1 shows CVMA in $\mathbb{F}_{p}^n$. Note that $n$ means $x \mod km + 1$.

- **Input:** $X = \sum_{i=0}^{m-1} x_0 \gamma^i$, $Y = \sum_{i=0}^{m-1} y_0 \gamma^i$ ($x_i, y_i \in \mathbb{F}_p$).
- **Output:** $Z = XY = \sum_{i=0}^{m-1} z_i \gamma^i$ ($z_i \in \mathbb{F}_p$).

**Preparation:**
1. Determine $k$ that satisfies the conditions in Def.1.
2. For $0 \leq i \leq m$, $q[i] \leftarrow 0$.
3. For $0 \leq i < m$ and $0 \leq h < k$, $g[(p^{i+h}m)] \leftarrow t + 1$.
4. $g[0] \leftarrow 0$.

**Procedure:**
1. For $0 \leq i < m$, $q[i + 1] \leftarrow q[i] - z_i y_i$.
2. For $0 \leq i < j \leq m - 1$, \(\{\)
   3. $M_{ij} \leftarrow (x_i - x_j)(y_j - y_i)$.
   4. For $0 \leq h \leq k - 1$, \(\{\)
   5. $q \leftarrow q[(p^i + p^{i+h}m)] + M_{ij}$.
   6. \(\}\)
   7. \(\}\)
   8. For $0 \leq i < m$, $z_i \leftarrow kq[i] - q[i + 1]$.

   **(End of algorithm)**

**Figure 1.** CVMA in $\mathbb{F}_{p}^n$

In the algorithm of Fig.1, $q[0]$ becomes 0 when $k$ is even [10]. Thus, the calculation cost of CVMA is 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.

\[
M_{mn} = \frac{m(m+1)}{2} M_n + \begin{cases} \frac{(m(m-1)(k+2)}{2} A_n + k - 1 + m \text{ when } k \text{ is odd,} \\ \frac{(m(m-1)(k+2)}{2} A_n \text{ when } k \text{ is even.} \end{cases}
\]

As shown in Eq. (4), 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.

3.3 Itoh-Tsujii algorithm

As an inversion algorithm in type I-X AOPF $\mathbb{F}_{(p^r)^m}$, Itoh-Tsujii algorithm (ITA) [11] where it uses Frobenius map is available. Consider a non-zero element $X$ that belongs to $\mathbb{F}_{(p^r)^m}$, its inverse element is given as

\[
X^{-1} = \frac{X^{p \cdot \ldots \cdot p^{m-1}}}{XX \ldots X^{p^{m-1}}}. \tag{5}
\]

Then, $XX \ldots X^{p^{m-1}}$ becomes a non-zero element that belongs to $\mathbb{F}_{p^r}$ because it is the norm of $X$ in $\mathbb{F}_{p^r}$. Frobenius map in type I-X AOPF does not need algebraic calculations such as additions and multiplications.

4. Type I-X AOPF for Freeman curve

Freeman has shown four kinds of curves that have the following characteristic $p$ [6].

\[
p = 503189899097385532598615948567975432740967203 \quad (149-\text{bit}) \tag{6a}
\]
\[
p = 610999632710831287460737695679448703542706164 \quad (196-\text{bit}) \tag{6b}
\]
\[
p = 182116508039694720644932643473759500459342546 \quad (234-\text{bit}) \tag{6c}
\]
\[
p = 646231009734881696220312491050525208267333884 \quad (252-\text{bit}) \tag{6d}
\]

In all these cases, type I-X AOPF $\mathbb{F}_{(p^r)^2}$ can be prepared by 2nd towering over $\mathbb{F}_{p^r}$ with type-(2, 2) GNB as Fig. 2. There is a problem that the minimal $k$ such that type I-X AOPF $\mathbb{F}_{p^r}$ can be constructed is $6$ in all these cases. With type-(6, 5) GNB as Fig. 2, the computation amount of a multiplication in $\mathbb{F}_{p^r}$ is given as

\[
M_5 = 15M_1 + 80A_1. \tag{7}
\]

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

\[
M_5 = 15M_1 + 40A_1. \tag{8}
\]

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

![Figure 2. Type I-X AOPF $\mathbb{F}_{(p^r)^2}$](image-url)
4.1 Improvement of CVMA

For example, in type I-X AOPF $F_{p^5}$ of Eq.(6b), CVMA needs to calculate the following equations.

\[ q[0] = 0, \]
\[ q[1] = x_{1}y_{1} + 2M_{01} + M_{03} + M_{12} + 2M_{13}, \]  
\[ q[2] = x_{2}y_{2} + M_{02} + 2M_{03} + M_{04} + 2M_{12}, \]
\[ q[3] = x_{3}y_{3} + M_{01} + M_{02} + 2M_{03} + 2M_{04}, \]
\[ q[4] = x_{4}y_{4} + 2M_{01} + 2M_{02} + M_{04} + M_{12}, \]
\[ q[5] = x_{5}y_{5} + M_{01} + M_{02} + M_{03} + 2M_{04} + 2M_{12} + M_{13} + 2M_{14} + 2M_{23} + M_{34}, \]

Consider Eqs.(9) except for $x_{1}y_{1}$, they can be expressed with the matrix shown as Eq.(10). In this matrix, the rows correspond to the coefficients of $M_{01}, M_{02}, M_{03}, M_{04}, M_{12}, \ldots, M_{24}$ and $M_{34}$.

\[
\begin{pmatrix}
2 & 0 & 1 & 0 & 1 & 2 & 1 & 2 & 2 & 1 \\
0 & 1 & 2 & 1 & 2 & 0 & 1 & 1 & 2 & 2 \\
1 & 1 & 2 & 2 & 0 & 1 & 2 & 2 & 0 & 1 \\
2 & 2 & 0 & 1 & 1 & 2 & 0 & 1 & 2 \\
1 & 2 & 1 & 2 & 2 & 0 & 1 & 1 & 0
\end{pmatrix}
\]

In this case, the computation amount of a multiplication in $F_{p^5}$ is given as follows. Note that $S$ denotes the computational cost of a bit-shift in $F_{p}$.

\[ M_{5} = 15M_{1} + 45A_{1} + 10S. \]  

\[ \text{Table 1} \] shows the computation amounts of a multiplication in each $F_{p^5}$ and $F_{p^{10}}$.

\begin{table}[h]
\begin{center}
\begin{tabular}{|c|c|c|}
\hline
$F_{p^5}$ & with original CVMA & with improved CVMA \\
\hline
$(15, 80, 0)$ & $(15, 45, 10)$ & \\
$(45, 260, 0)$ & $(45, 155, 30)$ & \\
\hline
\end{tabular}
\end{center}
\end{table}

\[ ^{\dagger} \text{For example, (15, 45, 10) denotes } 15M_{1} + 45A_{1} + 10S. \]

4.2 Reduction of modulo $p$ operations

A multiplication in extension field needs a lot of multiplications in prime field $F_p$, 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 $F_{p^5}$. 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 $F_{p^{10}}$ with a norm in $F_{p}$. In general, we directly calculate this norm. However, by calculating this norm after the calculation of a norm in the subfield $F_{p^5}$, we can carry out an inversion more efficiently. \[ \text{Table 2} \] shows the computation amounts of a norm in each $F_{p^5}$ and $F_{p^{10}}$ with the subfield. Note that $I_{n}$ means the computational cost of an inversion in $F_{p^n}$.

\begin{table}[h]
\begin{center}
\begin{tabular}{|c|c|c|}
\hline
$F_{p^5}$ & with original CVMA & with improved CVMA \\
\hline
$(104, 523, 0, 1)$ & $(104, 309, 64, 1)$ & \\
\hline
\end{tabular}
\end{center}
\end{table}

\[ ^{\dagger} \text{For example, (44, 114, 24, 1) denotes } 44M_{1} + 114A_{1} + 24S + I_{1}. \]

5. Simulation

This section shows the implementation result of the proposed method. In this implementation, we use characteristic $p$ of Eq.(6b). \[ \text{Table 3} \] shows the calculation timings of a multiplication and an inversion in each $F_{p^5}$ and $F_{p^{10}}$ with the computational environment \[ \text{Table 4}. \]

655
Table 3. Timing of each operation

<table>
<thead>
<tr>
<th></th>
<th>with original CVMA</th>
<th>with improved CVMA</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\mathbb{F}_{p^5})</td>
<td>mul 7.94 (\mu)s</td>
<td>mul 7.17 (\mu)s</td>
</tr>
<tr>
<td></td>
<td>inv 30.8 (\mu)s</td>
<td>inv 28.2 (\mu)s</td>
</tr>
<tr>
<td>(\mathbb{F}_{(p^5)^2})</td>
<td>mul 22.2 (\mu)s</td>
<td>mul 19.2 (\mu)s</td>
</tr>
<tr>
<td></td>
<td>inv 60.9 (\mu)s</td>
<td>inv 58.6 (\mu)s</td>
</tr>
</tbody>
</table>

Table 4. Computational environment

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>CPU</td>
<td>Pentium 3.00GHz</td>
</tr>
<tr>
<td>Cache Size</td>
<td>512 KB</td>
</tr>
<tr>
<td>OS</td>
<td>Linux 2.6.22</td>
</tr>
<tr>
<td>Language</td>
<td>C</td>
</tr>
<tr>
<td>Compiler</td>
<td>gcc 4.2.1</td>
</tr>
<tr>
<td>Library</td>
<td>GNU MP 4.2.2 [12]</td>
</tr>
</tbody>
</table>

Table 3 shows that extension field \(\mathbb{F}_{p^5}\) and \(\mathbb{F}_{(p^5)^2}\) with type-\((k, m)\) GNB for Freeman curve are efficient enough for practical use.

6. Conclusion

In this paper, we have considered how to constructed type I-X AOPF and optimized CVMA for Ate pairing with Freeman curve. 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.

References


656