that of the conventional method. Throughout this paper, p and k denote characteristic and extension degree, respectively. $\mathbb{F}_{p^k}$ denotes k-th extension field over $\mathbb{F}_p$ and $\mathbb{F}^*_{p^k}$ denotes the multiplicative group in $\mathbb{F}_{p^k}$. $X \mid Y$ and $X \nmid Y$ mean that X divides and does not divide Y, respectively. S_i, M_i, and I_i denote the calculation costs of a squaring, multiplication and inversion in $\mathbb{F}_{p^i}$, respectively.

2 COST EVALUATION

In this section, using BN curve whose embedding degree is 12, the cost evaluation of the proposed method [8] is shown.

2.1 Twisted Ate Pairing with BN Curve

Let ϕ be Frobenius endomorphism, i.e.,

$$\phi : E(\mathbb{F}_{p^{12}}) \to E(\mathbb{F}_{p^{12}}) : (x, y) \mapsto (x^p, y^p), \tag{3}$$

Then, let $\mathbb{G}_1$ and $\mathbb{G}_2$ be

$$\mathbb{G}_1 = E[r] \cap \mathrm{Ker}(\phi - [1]), \tag{4a}$$
$$\mathbb{G}_2 = E[r] \cap \mathrm{Ker}([\zeta_6]\phi^2 - [1]), \tag{4b}$$

where ζ_6 is a primitive 6-th root of unity and let $P \in \mathbb{G}_1$ and $Q \in \mathbb{G}_2$, twisted Ate pairing $\alpha(\cdot, \cdot)$ is defined as

$$\alpha(\cdot,\cdot) : \begin{cases} \mathbb{G}_1 \times \mathbb{G}_2 & \to \mathbb{F}^*_{p^{12}}/(\mathbb{F}^*_{p^{12}})^r \\ (P, Q) & \mapsto f_{T^2,P}(Q)^{(p^{12}-1)/r}. \end{cases} \tag{5}$$

In general, $A = f_{T^2,P}(Q)$ is calculated by Miller's algorithm [5], then so–called *final exponentiation* $A^{(p^{12}-1)/r}$ follows. The number of calculation loops of Miller's algorithm of twisted Ate pairing with BN curve is determined by $\lfloor \log_2(T^2) \rfloor$, where T^2 is given by

$$\begin{aligned} T^2 &= (t-1)^2 \bmod r \\ &= 36\chi^3 + 18\chi^2 + 6\chi + 1. \end{aligned} \tag{6}$$

It is said that calculation cost of Miller's Algorithm is about twice of that of final exponentiation.

2.2 Miller's Algorithm

Several improvements for Miller's algorithm have been proposed. Barreto et al. [9] proposed *reduced* Miller's algorithm. **Algorithm 1** shows the calculation flow of *reduced* Miller's algorithm for $f_{s,P}(Q)$. After **Algorithm 1**, T becomes sP. **Algorithm 1** consists of functions FDBL and FADD shown in **Algorithm 2** and **Algorithm 3**, *see also* **Table** 1. In **Algorithm 1**, FDBL denotes the calculation result of $l_{T,T}(Q)$ for which an elliptic curve doubling over $\mathbb{G}_1$ is needed and FADD denotes the calculation result of $l_{T,P}(Q)$ for which an elliptic curve addition over $\mathbb{G}_1$ is needed. Thus, let C_{FDBL} and C_{FADD} be the calculation costs of FDBL and FADD, they are given by

$$C_{\mathrm{FDBL}} = S_1 + 4M_1 + I_1, \tag{7a}$$
$$C_{\mathrm{FADD}} = 4M_1 + I_1. \tag{7b}$$

Algorithm 1 :Miller's Algorithm

MILLER1	
Input:	s, $P \in \mathbb{G}_1$, $Q \in \mathbb{G}_2$, $T \in \mathbb{G}_1$
Output:	$f_{s,P}(Q)$
1.	$f \leftarrow 1$, $T \leftarrow P$.
2.	For $i = \lfloor \log_2(s) \rfloor$ downto 1:
3.	$f \leftarrow f^2 \cdot$FDBL(T,Q).
4.	If $s[i] = 1$,then:
5.	$f \leftarrow f \cdot$FADD(T,P,Q).
6.	return f

In what follows, **Algorithm 1** is denoted by MILLER1 and $C_{\mathrm{ML1}}(s)$ denotes its calculation cost. Then, $C_{\mathrm{ML1}}(s)$ is given by C_{FDBL} and C_{FADD} as follows, where $\mathrm{Hw}(s)$ denotes the hamming weight of s.

$$\begin{aligned} C_{\mathrm{ML1}}(s) &= \lfloor \log_2(s) \rfloor \cdot (C_{\mathrm{FDBL}} + S_{12} + M_{12}) \\ &\quad + \mathrm{Hw}(s) \cdot (C_{\mathrm{FADD}} + M_{12}). \end{aligned} \tag{8}$$

In the case of twisted Ate pairing, let $P \in \mathbb{G}_1$, $Q \in \mathbb{G}_2$ and s be given by T^2 as Eq.(6), $f_{s,P}(Q)$ becomes an element in $\mathbb{F}^*_{p^{12}}$. Then, let C_{con} be the calculation costs of the conventional Miller's algorithm of twisted Ate pairing, it is given with C_{FDBL} and C_{FADD} as follows.

$$\begin{aligned} C_{con} &= C_{\mathrm{ML1}}(T^2) \\ &= \lfloor \log_2 T^2 \rfloor \cdot (C_{\mathrm{FDBL}} + S_{12} + M_{12}) \\ &\quad + \mathrm{Hw}(T^2) \cdot (C_{\mathrm{FADD}} + M_{12}). \end{aligned} \tag{9}$$

Algorithm 2 : FDBL of MILLER1

FDBL	
Input :	$T \in \mathbb{G}_1$, $Q \in \mathbb{G}_2$
Output :	$l_{T,T}(Q)$
1. $\lambda_{T,T} \leftarrow (3x_T^2)/(2y_T)$	$S_1+M_1+I_1$
2. $l_{T,T}(Q) \leftarrow (x_Q - x_T)\lambda_{T,T} - (y_Q - y_T)$	M_1
3. $x_{2T} \leftarrow \lambda_{T,T}^2 - 2x_T$	M_1
4. $y_{2T} \leftarrow (x_T - x_{2T})\lambda_{T,T} - y_T$	M_1
5. $T \leftarrow 2T$	
6. return $l_{T,T}(Q)$	

Algorithm 3 : FADD of Algorithm1

FADD	
Input :	$T, P \in \mathbb{G}_1, Q \in \mathbb{G}_2$
Output :	$l_{T,P}(Q)$
1. $\lambda_{T,P} \leftarrow (y_P - y_T)/(x_P - x_T)$	$M_1 + I_1$
2. $l_{T,P}(Q) \leftarrow (x_Q - x_P)\lambda_{T,P} - (y_Q - y_P)$	M_1
3. $x_{T+P} \leftarrow \lambda_{T,P}^2 - x_T - x_P$	M_1
4. $y_{T+P} \leftarrow (x_P - x_{T+P})\lambda_{T,P} - y_P$	M_1
5. $T \leftarrow T + P$	
6. return $l_{T,P}(Q)$	

Table 1: Meaning of notations

s_i :	i–th bit of the binary representation of s from the lower.
$l_{T,T}$:	the tangent line at T.
$l_{T,P}$:	the line passing through T and P.
$\lambda_{T,T}$:	the slope of the tangent line $l_{T,T}$.
$\lambda_{T,P}$:	the slope of the line $l_{T,P}$.

2.3 Proposed Method

The proposed method [8] calculates $f_{T^2,P}$ by using $f_{\chi,P}$, $f_{\chi^2,P}$ and $f_{\chi^3,P}$, where $f_{\chi,P}$, $f_{\chi^2,P}$ and $f_{\chi^3,P}$ are the rational functions for χD, $\chi^2 D$ and $\chi^3 D$, respectively. They hold the following relations.

$$\chi D = (\chi P) - (\mathcal{O}) + div(f_{\chi,P}), \quad (10)$$
$$\chi^2 D = (\chi^2 P) - (\mathcal{O}) + div(f_{\chi^2,P}), \quad (11)$$
$$\chi^3 D = (\chi^3 P) - (\mathcal{O}) + div(f_{\chi^3,P}). \quad (12)$$

Algorithm 4 shows the Miller's algorithm whose initial value of f is f'. After **Algorithm 4**, T becomes sP. In what follows, **Algorithm 4** is denoted by MILLER2. MILLER1 is equal to MILLER2 when f' is equal to 1. Let $C_{\mathrm{ML2}}(s)$ be the calculation costs of MILLER2, it is given by $C_{\mathrm{ML1}}(s)$ as follows.

if $f' = 1$,
$$C_{\mathrm{ML2}}(s) = C_{\mathrm{ML1}}(s), \quad (13a)$$
else
$$C_{\mathrm{ML2}}(s) = C_{\mathrm{ML1}}(s) + \mathrm{Hw}(s)\ M_{12}. \quad (13b)$$

According to the addition law for divisors, $f_{\chi^2,P}$ is calculated by **Algorithm 4** whose f' and P are given $f_{\chi,P}$ and χP, respectively. In this calculation, we need χP; however, we can previously obtain χP in the calculation of $f_{\chi,P}$. Similarly, $f_{\chi^3,P}$ is also calculated by **Algorithm 4**. Then $f_{T^2,P}$ is calculated by combining $f_{\chi,P}$, $f_{\chi^2,P}$ and $f_{\chi^3,P}$ because T^2 is decomposed as

$$\begin{aligned} T^2 &= 3(\ 12\chi^3 + 6\chi^2 + 2\chi\ ) + 1 \\ &= 6(\ 3(\ 2\chi^3 + \chi^2\ ) + \chi\ ) + 1. \end{aligned} \quad (14)$$

The detailed procedure has been shown in [8]. The proposed algorithm is given by **Algorithm 5**. In **Algorithm 5**, T_1, T_2, and T_3 are given χP, $\chi^2 P$, and $\chi^3 P$ by MILLER1 and MILLER2, respectively. Then, let the calculation costs of the proposed method be denoted by C_{pro}, it is given with $C_{\mathrm{ML1}}(s)$, $C_{\mathrm{ML2}}(s)$, C_{FDBL} and C_{FADD} as follows.

Algorithm 4 :Miller's Algorithm whose initial value of f is f'.

MILLER2

Input: s, $P \in \mathbb{G}_1$, $Q \in \mathbb{G}_2$, $f' \in \mathbb{F}_{p^{12}}$, $T \in \mathbb{G}_1$
Output: $f_{\chi,P'}(Q)$

1. $f \leftarrow f'$, $T \leftarrow P$.
2. For $i = \lfloor \log_2(s) \rfloor$ downto 1:
3. $\quad f \leftarrow f^2 \cdot \mathrm{FDBL}(T,Q)$.
4. $\quad$ If $s[i] = 1$,then:
5. $\quad\quad f \leftarrow f \cdot \mathrm{FADD}(T,P,Q)$.
6. $\quad\quad f \leftarrow f \cdot f'$.
7. return f

Algorithm 5 :Proposed Miller's Algorithm

Input: T^2, $P \in \mathbb{G}_1$, $Q \in E(\mathbb{F}_{p^{12}})$
Output: $f_{T^2,P}(Q)$

1. $f_1 \leftarrow$MILLER1(χ, P, Q, T_1)	$C_{\mathrm{ML1}}(\chi)$
2. $f_2 \leftarrow$MILLER2(χ, T_1, Q, f_1, T_2)	$C_{\mathrm{ML2}}(\chi)$
3. $f_3 \leftarrow$MILLER2(χ, T_2, Q, f_2, T_3)	$C_{\mathrm{ML2}}(\chi)$
4. $T \leftarrow T_3$	
5. $f \leftarrow f_3^2 \cdot \mathrm{FDBL}(T,Q)$.	$S_{12} + M_{12} + C_{\mathrm{FDBL}}$
6. $f \leftarrow f \cdot f_2 \cdot \mathrm{FADD}(T,T_2,Q)$.	$2M_{12} + C_{\mathrm{FADD}}$
7. $T' \leftarrow T$.	
8. $f \leftarrow f^3 \cdot \mathrm{FDBL}(T,Q)$.	$S_{12} + 2M_{12} + C_{\mathrm{FDBL}}$
9. $f \leftarrow f \cdot \mathrm{FADD}(T,T',Q)$.	$M_{12} + C_{\mathrm{FADD}}$
10. $f \leftarrow f \cdot f_1 \cdot \mathrm{FADD}(T,T_1,Q)$.	$2M_{12} + C_{\mathrm{FADD}}$
11. $T' \leftarrow T$.	
12. $f \leftarrow f^3 \cdot \mathrm{FDBL}(T,Q)$.	$S_{12} + 2M_{12} + C_{\mathrm{FDBL}}$
13. $f \leftarrow f \cdot \mathrm{FADD}(T,T',Q)$.	$M_{12} + C_{\mathrm{FADD}}$
14. $f \leftarrow f^2 \cdot \mathrm{FDBL}(T,Q)$.	$S_{12} + M_{12} + C_{\mathrm{FDBL}}$
15. $f \leftarrow f \cdot \mathrm{FADD}(T,P,Q)$.	$M_{12} + C_{\mathrm{FADD}}$
16. Return f	

$$\begin{aligned} C_{pro} &= C_{\mathrm{ML1}}(\chi) + 2\ C_{\mathrm{ML2}}(\chi) + 4\ C_{\mathrm{FDBL}} \\ &\quad + 5\ C_{\mathrm{FADD}} + 4S_{12} + 13M_{12} \\ &= 3\ C_{\mathrm{ML1}}(\chi) + 2\ \mathrm{Hw}(\chi)\ M_{12} \\ &\quad + 4\ C_{\mathrm{FDBL}} + 5\ C_{\mathrm{FADD}} + 4S_{12} + 13M_{12} \\ &= 3\ (\ \lfloor \log_2(\chi) \rfloor\ (\ C_{\mathrm{FDBL}} + S_{12} + M_{12}\ ) \\ &\quad\quad + \mathrm{Hw}(\chi) \cdot (C_{\mathrm{FADD}} + M_{12}\ )\ ) \\ &\quad + 4\ C_{\mathrm{FDBL}} + 5\ C_{\mathrm{FADD}} \\ &\quad + 4S_{12} + (\ 2\ \mathrm{Hw}(\chi) + 13\ )M_{12}. \end{aligned} \quad (15)$$

From **Algorithm 5**, the proposed method calculates $f_{\chi,P}$, $f_{\chi^2,P}$ and $f_{\chi^3,P}$ by MILLER1 and MILLER2 whose parameter s are equal to χ, respectively. It is known that the calculation cost of Miller's algorithm can be reduced by using s of small hamming weight. Thus, the proposed method can reduce the calculation costs of Miller's algorithm by using χ of small hamming weight.

Table 2: Timing of each operation in extension field[μs]

p, r		158 bit	254 bit
$\mathbb{F}_p$	mul	0.41	0.65
	inv	4.93	8.43
$\mathbb{F}_{p^2}$	mul	1.08	1.65
	inv	6.82	11.4
$\mathbb{F}_{p^4}$	mul	2.92	4.39
	inv	12.4	19.6
$\mathbb{F}_{p^6}$	mul	5.98	7.78
	inv	22.8	32.4
$\mathbb{F}_{p^{12}}$	mul	14.7	21.6
	inv	54.2	80.3
	sqr	13.2	19.7

2.4 Cost Evaluation and Comparison

This section compares the calculation costs of the conventional and proposed methods. In order to make the cost evaluation simple, we only take the calculation costs for squaring, multiplication, and inversion in finite field into acount. **Table** 2 shows the timing of each operation in extension field. According to **Table** 2, let $S_{12} = 32M_1$, $M_{12} = 36M_1$, and $I_1 = 12M_1$. Then, supposing that roughly $S_1 = M_1$, C_{con} and C_{pro} become as follows from Eq.(9), Eq.(15), Eq.(7a) and Eq.(7b).

$$\begin{aligned} C_{con} &= \lfloor \log_2(T^2) \rfloor \cdot (17M_1 + 32M_1 + 36M_1) \\ &\quad + \mathrm{Hw}(T^2) \cdot (16M_1 + 36M_1) \\ &= \lfloor \log_2(T^2) \rfloor \, 85M_1 + \mathrm{Hw}(T^2) \, 52M_1 \qquad (16a) \\ C_{pro} &= 3 \, (\lfloor \log_2(\chi) \rfloor \cdot (17M_1 + 32M_1 + 36M_1) \\ &\quad + \mathrm{Hw}(\chi) \cdot (16M_1 + 36M_1)) + 4 \cdot 17M_1 \\ &\quad + 5 \cdot 16M_1 + 4 \cdot 32M_1 + (2\mathrm{Hw}(\chi) + 13) \cdot 36M_1 \\ &= 3 \, \lfloor \log_2(\chi) \rfloor \cdot 85M_1 + \mathrm{Hw}(\chi) \cdot 228M_1 \\ &\quad + 744M_1 \qquad (16b) \end{aligned}$$

Then, let $\lfloor \log_2(T^2) \rfloor = 3 \lfloor \log_2(\chi) \rfloor$ from Eq.(6). Supposing that $\mathrm{Hw}(\chi)$=3, the condition that $\mathrm{Hw}(T^2)$ satisfies $C_{pro} > C_{con}$ is obtained as

$$\begin{aligned} 52M_1 \cdot \mathrm{Hw}(T^2) &< 228M_1 \cdot \mathrm{Hw}(\chi) + 744M_1 \\ 52M_1 \cdot \mathrm{Hw}(T^2) &< 1428M_1 \\ \mathrm{Hw}(T^2) &< 28. \qquad (17) \end{aligned}$$

Table 3 shows all $\chi's$ of Hamming weight 3 that gives 158–bit and 254–bit prime order BN curve. According to **Table** 3, $\mathrm{Hw}(T^2)$ that satisfies Eq.(17) does not exist when the hamming weight of χ is equal to 3. Then, using every χ that give 158–bit and 254–bit prime order BN curves, the authors calculated $\mathrm{Hw}(T^2)$; however, every $\mathrm{Hw}(T^2)$ did not satisfy Eq.(17). Thus, in the case of 158–bit and 254–bit prime order BN curves, the calculation cost of the proposed method is much less than that of the conventional method.

3 CONCLUSION

The authors have proposed an improvement of twisted Ate pairing with BN curve in ITC–CSCC2008 [8]. In the proceeding [8], its cost evaluation has not been explicitly shown. In this paper, the detail of the cost evaluation was shown. Then, in the case of χ with small hamming weight, this paper showed that the calculation cost of the proposed method was less than that of the conventional method.

Table 3: χ of small Hamming weight that gives 158–bit and 254–bit prime order BN curve

$p(\chi)$	χ	$\mathrm{Hw}(T^2)$
158 bit	$2^{38} + 2^{15} + 2^{14}$	65
	$2^{38} + 2^{27} + 2^{16}$	62
	$2^{38} + 2^{28} + 1$	36
	$-2^{38} - 2^{32} - 2^5$	47
254 bit	$2^{62} + 2^{46} + 2^{29}$	83
	$2^{62} + 2^{35} + 2^{24}$	82
	$2^{62} + 2^{55} + 1$	36
	$-2^{62} - 2^{41} - 2^{23}$	43

References

[1] D.Boneh, B.Lynn, and H.Shacham, "Short signatures from the Weil pairing," Asiacrypt2001, LNCS 2248, pp. 514–532, 2001.

[2] T. Nakanishi, and N. Funabiki, "Verifier-Local Revocation Group Signature Schemes with Backward Unlinkability from Bilinear Maps," ASIACRYPT2005, LNCS, Vol. 3788, pp. 533–548, 2005.

[3] H. Cohen and G. Frey, *Handbook of Elliptic and Hyperelliptic Curve Cryptography*, *Discrete Mathematics and Its Applications*, Chapman & Hall CRC, 2005.

[4] S. Matsuda, N. Kanayama, F. Hess, and E. Okamoto, "Optimised versions of the Ate and Twisted Ate Pairings," IMA2007, LNCS, Vol. 4887, pp. 302–312, 2007.

[5] A. J. Devegili, M. Scott, and R. Dahab, "Implementing Cryptographic Pairings over Barreto-Naehrig Curves," LNCS, Vol. 4575, pp. 197–207, 2007.

[6] M. Akane, H. Kato, T. Okimoto, Y. Nogami, and Y. Morikawa, "An Improvement of Miller's Algorithm in Ate Pairing with Barreto–Naehrig Curve," Proc. of Computer Security Symposium 2007 (CSS2007), pp. 489–494, 2007.

[7] P. S. L. M. Barreto, and M. Naehrig, "Pairing-Friendly Elliptic Curves of Prime Order," SAC2005, LNCS, Vol. 3897, pp. 319–331, 2006.

[8] Y. Sakemi, H. Kato, Y. Nogami, and Y. Morikawa, "An Improvement of Twisted Ate Pairing Using Integer Variable with Small Hamming Weight," Proc. of the 23rd International Technical Conference on Circuits/Systems, Computers and Communications 2008 (ITC–CSCC2008), pp. 269–272, 2008.

[9] P. S. L. M. Barreto, B. Lynn, and M. Scott, "Efficient Implementation of Pairing-Based Cryptosystems," J. Cryptology 2004, Vol.17, pp. 321–334, 2004.