BKLS Algorithm に Window 法を用いた Tate ペアリングの高速化に関する考察

Fast Computation of Tate Pairing with BKLS Algorithm and Window Method

15D8101012B　増渕 佳輝
中央大学理工学部情報工学科 趙研究室
2019 年 3 月

要約　本研究ではペアリング暗号の演算で使用する Miller Algorithm を改良した BKLS Algorihtm に Window 法を適用し，Tate ペアリングの高速化を行った．

キーワード　ペアリング暗号, Tate ペアリング, Miller Algorithm, BKLS Algorithm,

1　序論

楕円曲線暗号とは有限体上の楕円曲線を用いた暗号で, これに対する攻撃方法としてペアリングが用いられた．その後, ペアリングを用いた暗号である ID ベース暗号への応用などに使われ, 近年では，ペアリングを用いたプロトコルが数多く提案されている．楕円曲線上のペアリングとして, Weil ペアリングや Tate ペアリングがあるが, 通常の楕円演算に比べて演算量が多いことが問題となっている．したがって, ペアリングの高速化が課題となっている．

本研究ではペアリング暗号の演算で使用する Miller Algorithm を改良した BKLS Algorihtm に, に Window 法を適用しの Tate ペアリングの高速化を行い, 計算コストと計算時間の比較を行った．

2　楕円曲線の定義

楕円曲線とは, 一般的に

$$E : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6$$

で与えられる. 素数 m, 有限体 $\mathbb{F}_q$ $(q = p^m)$ 上の楕円曲線とは, この方程式を満たす有理点 (x, y) に無限遠点 $\mathcal{O}$ を加えた集合のことであり, $E(\mathbb{F}_q)$ と表す．また, 定義体 $\mathbb{F}_q$ の標数が 3 より大きい場合は変数変換により, $y^2 = x^3 + ax + b$ と一般化できる．

3　ペアリング

3.1　Tate ペアリング

有限体 $\mathbb{F}_q$ 上の楕円曲線を $y^2 = x^3 + ax + b$ とし, 素数 n, 埋め込み次数 k を $n|q^k - 1$ を満たす最小の整数とする．楕円曲線上の点 P, Q を $P \in E(\mathbb{F}_q)[n]$, $Q \in E(\mathbb{F}_{q^k})$ と定め, Tate ペアリングを次に定義する．

$$E(\mathbb{F}_q)[n] \times E(\mathbb{F}_{q^k})/nE(\mathbb{F}_{q^k}) \to \mathbb{F}^*_{q^k}/(\mathbb{F}^*_{q^k})^n$$
$$e(P, Q) = f_n(Q)^{(q^k-1)/n} = (f_P(Q+S)/f_P(S))^{(q^k-1)/n}$$

3.2　Reduced Tate ペアリング

Tate ペアリングの値は剰余類全体の集合 $\mathbb{F}^*_{q^k}/(\mathbb{F}^*_{q^k})^n$ に属しており, 一意に定まらないので, $(q^k - 1)/n$ 乗することで, 一意な値を得られる．最終べき乗した Reduced Tate ペアリングを次に定義する．

$$P \in E(\mathbb{F}_q)[n],\ Q \in E(\mathbb{F}_{q^k}),\ \mu_n = \left\{x \in \mathbb{F}^*_{q^k} | x^n = 1\right\}$$
$$\tau\langle P, Q\rangle = \langle P, Q\rangle_n^{(q^k-1)/n} = f_{n,P}(Q)^{(q^k-1)/n} \in \mu_n$$

さらに, $N = hn$ に対して次の式が成立する．

$$\tau(P, Q) = \langle P, Q\rangle_n^{(q^k-1)/n}$$

3.3　Miller Algorithm

ペアリングの計算手法として Miller Algorithm がある．$\mathbb{F}_q$ 上の楕円曲線の Reduced Tate ペアリングにおける Miller Algorithm を次に示す．

Algorithm 1: Miller Algorithm
Input: $n,\ l = \log n,\ P \in E(\mathbb{F}_q)[n],\ Q \in E(\mathbb{F}_{q^k})$
Output: $f \in \mathbb{F}_{q^k}$
1: $V \leftarrow P,\ f \leftarrow 1,\ n = \sum_{i=0}^{l-1} n_i 2^i,\ n_i \in \{0, 1\}$
2: for $j \leftarrow l - 1$ down do 0
3: $f \leftarrow f^2 \cdot \frac{g_{V,V}(Q)}{g_{2V}(Q)}$. $V \leftarrow 2V$
4: if $n_j = 1$ then
5: $f \leftarrow f \cdot \frac{g_{V,P}(Q)}{g_{V+P}(Q)},\ V \leftarrow V + P$
6: return f

3.4　BKLS Algorithm

supersingular curve の distortion map ψ を利用して分母消去の手法を適用した BKLS Algorithm [?] を次に示す．ordinary curve の場合, $Q' \in E'(K)$ として, distortion map ではなく twist の同型写像 ψ_d を用いる．

Input: $P,\ Q \in E(K_0)[n]$
Output: $f \in K$
1: $f \leftarrow 1,\ V \leftarrow P$
2: $n = \sum_{l-1}^{i=0} n_i 2^i,\ n_i \in \{0, 1\}$
3: for $j \leftarrow l - 1$ down 0 do 0
4: $f \leftarrow f^2 \cdot g_{V,\ V}(\psi(Q))$
5: $V \leftarrow 2V$
6: if $n_j = 1$ then
7: $f \leftarrow f \cdot g_{V,\ P}(\psi(Q))$
8: $V \leftarrow V + P$
9: return f

3.5 Window Miller Algorithm

Input: $n,\ P \in E(\mathbb{F}_q)[n],\ Q \in E(\mathbb{F}_{q^k})\ S \in E(\mathbb{F}_{q^k})$

Output: $f \in \mathbb{F}_{q^k}$

(online computation)

1: $P_1 = P, f_1' = 1$

2: for $i \leftarrow i$ up to do $2^w - 1$

3: $P_i \leftarrow iP_i$

4: $f \leftarrow f \cdot \frac{g_{P,-P_i}(S)g_{\mathcal{O},P_i}(Q+S)}{g_{P,-P_i}(S)g_{\mathcal{O},P_i}(Q+S)}$

(main computation)

5: $T \leftarrow P_i, f \leftarrow 1$

6: $n = \sum_{i=0}^{l-1} n_i 2^i,\ n_i \in \{0,1\},\ n_0 = 1$

7: for $n-1 \leftarrow i$ down to 0 step w

8: step 8-1 から 8-2 を w 回繰り返す

8-1: $T \leftarrow 2T$

8-2: $f \leftarrow f^2 \cdot \frac{g_{T,-2T}(Q+S)g_{\mathcal{O},2T}(S)}{g_{T,-2T}(Q+S)g_{\mathcal{O},2T}(S)}$

9: $m' \leftarrow= \sum_{i}^{j=i-w+1} m[j]2^{j-i+w-1}$

10: if $m' \neq 0$ then

10-1: $T \leftarrow T + P_{m'}$

10-2: $f \leftarrow f^2 \cdot \frac{g_{T,-2T}(Q+S)g_{\mathcal{O},2T}(S)}{g_{T,-2T}(Q+Sg_{\mathcal{O},2T}(S)}$

11: return f

4 提案手法

5 結論

謝辞

本研究において, あらゆる面でご指導していただいた趙晋輝教授並びに相賀氏, 山岸氏を始めとする諸先輩方, 趙研究室の皆様にも深く感謝いたします.

参考文献

[1] V. S. Dimitrov, L. Imbert, and P.K.Mishra: *Efficient and secure elliptic curve point multiplication using double-base chains*. LNCS 3788, 2005.

[2] C. Zhao, F. Zhang and J. Huang: *Efficient Tate Pairing Computation Using Double-Base Chains*. Science in China Series F: Information Sciences, 2008, vol. 51, no. 8.

[3] S. Matsuda, N. Kanayama, F. Hess, and E. Okamoto. *Optimised versions of the Ate and twisted Ate pairings*. Appear to the 11th IMA International Conference on Cryptography and Coding.