

Distorsion Map の n-進展開を用いた Miller Algorithm の高速化に関する研究

Fast Computation of Miller Algorithm with N-ary expansion of Distortion maps

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 に distortion map を用いた BKLS Algorithm に Window 法を適用し、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く、ペアリング計算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 に distortion map を用いることで改良した BKLS Algorithm に Window 法を適用し、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 p 、有限体 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 O を加えた集合のことであり、 $E(F_q)$ と表す。また、定義体 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 ペアリングの定義

n を整数とする。 G_1, G_2 を単位元 0 の加法アーベル群とする。 G_1, G_2 は位数 n を持つ。 G_3 は単位元 1 の乗法に関する位数 n の巡回群とする。 ペアリングというのは以下の関数である。

$$e: G_1 \times G_2 \longrightarrow G_3$$

全てのペアリングは以下の 2 つの性質を満たす。

・ 双線形性

全ての $P, P' \in G_1$ と $Q, Q' \in G_2$ に対して、

$$e(P + P', Q) = e(P, Q) + e(P', Q)$$

$$e(P, Q + Q') = e(P, Q) + e(P, Q')$$

が成り立つ。

・ 非退縮性

全ての $P \in G_1$ ($P \neq 0$) に対して $e(P, Q) \neq 1$ となるような $Q \in G_2$ が存在する。

全ての $Q \in G_2$ ($Q \neq 0$) に対して $e(P, Q) \neq 1$ となるような $P \in G_1$ が存在する。

3.2 Tate ペアリング

有限体 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 $P \in E(F_q)[n]$ 、 $Q \in E(F_{q^k})$ と定め、Tate ペアリングを次に定義する。

$$E(F_q)[n] \times E(F_{q^k})/nE(F_{q^k}) \rightarrow F_{q^k}^*/(F_{q^k}^*)^n$$
$$e(P, Q) = f_n(Q)^{(q^k-1)/n} = (f_P(Q+S)/f_P(S))^{(q^k-1)/n}$$

3.3 Reduced Tate ペアリング

Tate ペアリングの値は剰余類全体の集合 $F_{q^k}^*/(F_{q^k}^*)^n$ に属しており、 $(q^k - 1)/n$ 乗することで、一意な値を得られる。Reduced Tate ペアリングを次に定義する。

$$P \in E(F_q)[n], Q \in E(F_{q^k}), \mu_n = \left\{ x \in F_{q^k}^* | x^n = 1 \right\}$$

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

さらに、 $N = hn$ に対して次の式が成立する。

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

3.4 Miller Algorithm

ペアリングの計算手法として Miller Algorithm がある。 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。点 $U, V \in E(F_{q^k})$ を通る直線 $g_{U,V}$ とする。

3.5 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 [1] 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: $n, l = \log n, P \in E(\mathbb{F}_q)[n], Q \in E(\mathbb{F}_{q^k})$ 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 to 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

表 1 Miller Algorithm

Input: $P, Q \in E(\mathbb{F}_q)[n]$ Output: $f \in \mathbb{F}_{q^k}$
1: $f \leftarrow 1, V \leftarrow P$
2: $n = \sum_{i=0}^{l-1} n_i 2^i, n_i \in \{0, 1\}$
3: for $j \leftarrow l-1$ down to 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

表 2 BKLS Algorithm

4 提案手法

BKLS Algorithm, Window Miller Algorithm を組みわせることで新しい高速化手法を提案する。

Input: $n, P, Q \in E(\mathbb{F}_q)[n]$ Output: $f \in \mathbb{F}_{q^k}$
(online computation)
1: $P_1 = P, f'_1 = 1$
2: for $i \leftarrow 2$ up to $2^w - 1$
3: $P_i \leftarrow P + P_{i-1}$
4: $f'_i \leftarrow f'_{i-1} \cdot g_{P_i, P}(\psi(Q))$
(main computation)
5: $V \leftarrow P, f \leftarrow 1$
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: $V \leftarrow 2V$
8-2: $f \leftarrow f^2 \cdot g_{V,V}(\psi(Q))$
9: $m' \leftarrow \sum_{j=i-w+1}^i m[j] 2^{j-i+w-1}$
10: if $m' \neq 0$ then
10-1: $f \leftarrow f f'_m \cdot g_{V, P_m}(\psi(Q))$
10-2: $V \leftarrow V + P_{m'}$
11: return f

表 3 Window 方を用いた BKLS Algorithm

3.6 Window Miller Algorithm

window Miller アルゴリズムは、オンライン事前演算を用いる方法である。このアルゴリズムでは、 n/w 回行うことになるため、適切な w を用いれば、楕円加算および直線 $l_{P,-P_i}$, 垂線 v_{P_i} の計算を削減することができる。

Input: $n, P \in E(\mathbb{F}_q)[n], Q \in E(\mathbb{F}_{q^k})$ Output: $f \in \mathbb{F}_{q^k}$
(online computation)
1: $P_1 = P, f'_1 = 1$
2: for $i \leftarrow 2$ up to $2^w - 1$
3: $P_i \leftarrow P + P_{i-1}$
4: $f \leftarrow f \cdot \frac{g_{P,-P_i}(S)g_{O,P_i}(Q+S)}{g_{P,-P_i}(S)g_{O,P_i}(Q+S)}$
(main computation)
5: $T \leftarrow P, f \leftarrow 1$
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: $f \leftarrow f^2 \cdot \frac{g_{T,-2T}(Q+S)g_{O,2T}(S)}{g_{T,-2T}(Q+S)g_{O,2T}(S)}$
9: $m' \leftarrow \sum_{j=i-w+1}^i m[j] 2^{j-i+w-1}$
10: if $m' \neq 0$ then
10-1: $T \leftarrow T + P_{m'}$
10-2: $f \leftarrow f^2 \cdot \frac{g_{T,-2T}(Q+S)g_{O,2T}(S)}{g_{T,-2T}(Q+S)g_{O,2T}(S)}$
11: return f

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] P.S.L.M. Barreto, H.Y. Kim, B. Lynn, and M. Scott: *Efficient implementation of pairing-based cryptosystem*, Journal of Cryptology, 17(4):321-334, 2004.