SCIS 2008 The 2008 Symposium on Cryptography and Information Security
Miyazaki, Japan, Jan. 22-25, 2008
The Institute of Electronics, Information and Communication Engineers

部分体曲線の点圧縮の方法
A Point Compression Method for Subfield Curves

星野 文学* 小林 鉄太郎 *
Fumitaka Hoshino Tetsutaro Kobayashi

あらまし 大きな平文空間を持つ再暗号化可能な ElGamal 暗号は, 巡回群の元に直接平文を埋め込んだり, 巡回群の元から平文が容易に復元出来るような特別な性質を巡回群に要求する. 巡回群として楕円曲線を用いる場合, 曲線の位数が素数であるときは, 平文を x 座標の部分情報に埋め込み, x の残りの部分情報を適当に選んで, 楕円曲線の定義式を満たす y を見つければ十分である. しかし, 楕円曲線の位数が合成数で, コファクター c を持つような場合は上記のようにして見つけた (x, y) が目的の巡回群に載る確率はおよそ $1/c$ である. c が小さい場合はこの埋め込みアルゴリズムは機能するが, OEF で部分体曲線を使う場合のように曲線に大きなコファクターがあると機能しない. コファクターが大きい時は必要としている部分群のサイズに比べて点を定義するデータ (x 座標) のサイズが冗長なため, 部分群の点を効率的に見つけられない事が問題の本質である. 従って部分群の点を小さい情報量で表現する事 (即ち点圧縮) が可能であればこの問題は解決する. 以上の工学的動機に基づき、本稿では比較的大きい部分体で定義された楕円曲線の点圧縮の方法を提案する.

キーワード 楕円曲線, 拡大体, 点圧縮, OEF, Koblitz Curve

1 はじめに

一般に, 離散対数困難な大きな素数位数の巡回群を使って安全性の高い暗号や署名を構成する場合には, 巡回群の元は単に乱数倍されたりハッシュ関数に入力されたりして専ら乱数や乱数の種を生成するのに用いられる. このような安全性の高い暗号を使うとき, 大抵の場合は平文メッセージを巡回群の元に埋め込んだり, 巡回群の元から平文メッセージを取り出したりする必要はない. 平文メッセージは使用している巡回群の数学的な構造とはまったく関係なさそうなハッシュ関数や共通鍵暗号を用いて巧妙に符号化される.

では, どのような場合に巡回群の元にメッセージを埋め込む必要があるだろうか ? 実は, アプリケーションが暗号化以外の特殊な機能を要求する場合に, 平文メッセージの埋め込みを要する事がある.

例えば, 大きな平文空間を持つ再暗号化可能な ElGamal 暗号は, 巡回群の元に埋め込まれた平文が容易に復元出来るような特別な性質を巡回群に要求する. このような暗号は原理的に IND-CCA2 にはなり得ないが, 匿名通信や電子投票あるいは RFID プライバシ保護といった現実的なアプリケーションでしばしば活用される.

従って, 暗号設計者が上記のような用途にも使える万能巡回群としての楕円曲線を設計したいなら, 平文メッセージを効率的に楕円曲線上の点に埋め込むアルゴリズムを用意する必要がある. そして多くの楕円曲線では平文メッセージを楕円上の点の x 座標の部分情報として埋め込み x を構成する残りの情報を適当に選んで, 楕円曲線の定義式 $y^2 = x^3 + ax + b$ を満たす解を探すことでこの目的を果たす [Kob94].

楕円曲線の位数が素数であるような場合、このように選ばれた x に対して y が目的の体上に解を持つ確率はおよそ $1/2$ であり, x 座標の平文以外の部分情報を適当に操作すれば目的の (x, y) がすぐに見つかる. 楕円曲線の位数が合成数のときは, 曲線がコファクター c を持つので, 上記のようにして見つけた (x, y) が目的の巡回群に載る確率はおよそ $1/c$ であり, 点の生成は素数位数の場合より c 倍遅い. よく利用されるような曲線の場合は曲線自体の位数が素数であるか, または小さなコファクターしか持たないよう注意深くパラメタが選ばれる [Sec00].

ところで, OEF で部分体曲線を用いる場合には, 楕円曲線は計算機の語長程度のサイズのコファクターを持つ. 従って, 上記のような平文メッセージの埋め込みアルゴリズムはあまり効率的ではない. 32 bit 語長計算機用の OEF の部分体曲線は 32 bit 程度のコファクターを持っているので, 上記の埋め込みアルゴリズムは素数位数の

* NTT 情報流通プラットフォーム研究所, 〒 180-8585 東京都武蔵野市緑町 3-9-11, NTT Information Sharing Platform Laboratories, 3-9-11 Midori-cho, Musashino-shi, Tokyo, 180-8585 Japan

場合に比べ およそ 2^{32} 倍遅く, パーソナルコンピューター上でなんとか実行できる程度の性能しか実現できなかった.
さらに近年 64 bit 語長計算機が普及してきており、著者らも 64 bit 語長計算機用 OEF の提案を行っているが [HKA06], ここで使用されている曲線はおよそ 2^{61} くらいのコファクターを持っており, 上記の方法での平文メッセージの埋め込みは事実上不可能となっている.
飯島らは twist を使って, 高速な楕円スカラー倍演算が可能であるにもかかわらず, コファクターが 1 となる曲線を構成した. 飯島らの方法はこうした問題の一つの解となるが, この方法には拡大次数に制約がある [IMCT02]. 本質的な問題は, 必要としている部分群の位数のサイズに比べて, 点を定義するデータのサイズが冗長なため, 部分群の点を効率的に見つけられない所にある. 従って部分群の点を小さい情報量で表現する方法, 即ち点圧縮の方法を検討したい.
楕円曲線の点圧縮の方法には、y 座標を 1 bit のみで表現する良く知られた方法の他に, $\mathbb{F}_2$ の拡大体上の楕円曲線で x 座標の trace を用いる方法 [Ser98] や, pairing 用の楕円曲線で拡大体上の点を twist を使って圧縮する方法 [HSV06, BN05] などが研究されている.
本稿では OEF で部分体曲線を使った場合のような比較的大きい部分体で定義された楕円曲線の点圧縮の方法を提案する.

2 準備

q を 5 以上の素数またはその冪とし $a, b \in \mathbb{F}_q$ として, 楕円曲線 $E/\mathbb{F}_q$ を

$$E/\mathbb{F}_q : y^2 = x^3 + ax + b$$

とする. また q 乗 Frobenius 写像 ϕ を

$$\phi : (x, y) \mapsto (x^q, y^q)$$

と定義する. ϕ は E 上の自己準同型写像である.

Lemma 1 $E(\mathbb{F}_{q^m})/E(\mathbb{F}_q)$ が位数 ℓ の巡回群で, $\#E(\mathbb{F}_q)$ と ℓ は互いに素ならば

$$E(\mathbb{F}_{q^m})[\ell] = (\phi - 1)E(\mathbb{F}_{q^m})$$

[証明] $\phi - 1$ は $E(\mathbb{F}_{q^m})$ 上の自己準同型であるから, 準同型定理によって $E(\mathbb{F}_{q^m})$ を

$$\mathrm{Ker}(\phi - 1) \oplus \mathrm{Im}(\phi - 1)$$

に分解できる. $\mathrm{Ker}(\phi - 1) = E(\mathbb{F}_q)$ であるから

$$\mathrm{Im}(\phi - 1) \simeq E(\mathbb{F}_{q^m})/E(\mathbb{F}_q).$$

$E(\mathbb{F}_{q^m})/E(\mathbb{F}_q)$ が位数 ℓ の巡回群で, $\#E(\mathbb{F}_q)$ と ℓ が互いに素なので

$$E(\mathbb{F}_{q^m})[\ell] = (\phi - 1)E(\mathbb{F}_{q^m})$$

である. □

Remark 1 ℓ を素数とする為に OEF で Koblitz 曲線を用いる場合は m を素数とする.

安全で効率的な楕円暗号用のパラメタを選べば, Lemma 1 や Remark 1 が自然に満たされることが多いので, 点圧縮の対象となる部分群を $E(\mathbb{F}_{q^m})[\ell]$ とする代わりに $(\phi - 1)E(\mathbb{F}_{q^m})$ と考えても差し支えない.

Lemma 2

$$P \in (\phi - 1)E(\mathbb{F}_{q^m}) \Rightarrow \sum_{i=0}^{m-1} \phi^i P = \mathcal{O}$$

[証明] $P \in (\phi - 1)E(\mathbb{F}_{q^m})$ より

$$\sum_{i=0}^{m-1} \phi^i P \in (\phi^m - 1)E(\mathbb{F}_{q^m}) = \{\mathcal{O}\}$$

従って $\sum_{i=0}^{m-1} \phi^i P = \mathcal{O}$. □

Remark 2 m と $\#E(\mathbb{F}_q)$ が互いに素なら逆が言える.

3 $m = 3$ の場合

一般の場合を議論する前に, 簡単な場合として $m = 3$ を考え, より大きな m についての手がかりを得たい. この場合 $P \in (\phi - 1)E(\mathbb{F}_{q^m})$ なら Lemma 2 より

$$P + \phi P + \phi^2 P = \mathcal{O}$$

である, 従って $P, \phi P, \phi^2 P$ の 3 点は 1 本の直線上に載っている. この事がこの場合の問題を単純にする.

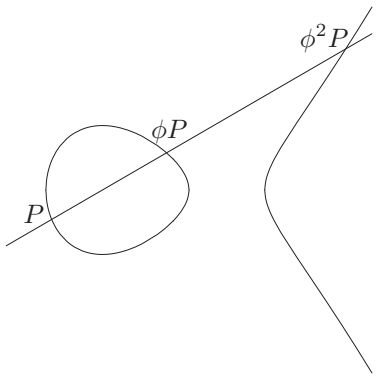

図 1: $m = 3$ の場合 (イメージ図)

Lemma 3 (encoding) $m = 3$ とし $P = (x, y) \in E(\mathbb{F}_{q^m}) \setminus E(\mathbb{F}_q)$ とする.

$$\sum_{i=0}^{m-1} \phi^i P = \mathcal{O} \;\Rightarrow\; \exists \lambda_2, \lambda_0 \in \mathbb{F}_q,\; y + \lambda_2 x + \lambda_0 = 0$$

[証明] $P \in E(\mathbb{F}_{q^m}) \setminus E(\mathbb{F}_q)$ であるから, 3 次拡大の場合には $x \in \mathbb{F}_{q^m} \setminus \mathbb{F}_q$ としてよい. 従って x, x^q, x^{q^2} は全て異なる値を持つ. $P + \phi P + \phi^2 P = \mathcal{O}$ なので, $P, \phi P, \phi^2 P$ の 3 点は 1 本の直線上に載っており, $P, \phi P$ を結ぶ直線と $\phi P, \phi^2 P$ を結ぶ直線の傾きを比較すると

$$\lambda_2 = -\frac{y^q - y}{x^q - x} = -\frac{y^{q^2} - y^q}{x^{q^2} - x^q} = \lambda_2^q$$

であるから

$$\lambda_2 \;\in\; \mathbb{F}_q$$

また, $\lambda_2 = -\dfrac{y^q - y}{x^q - x}$ より,

$$\lambda_0 = -y - \lambda_2 x = (-y - \lambda_2 x)^q = \lambda_0^q$$

であるから

$$\lambda_0 \;\in\; \mathbb{F}_q$$

従って, $y + \lambda_2 x + \lambda_0 = 0$ なる $\lambda_2, \lambda_0 \in \mathbb{F}_q$ が存在する. □

Lemma 4 (decoding) $m = 3$ として, 上記の $\lambda_2, \lambda_0 \in \mathbb{F}_q$ が与えられれば, $(x, y) \in (\phi - 1)E(\mathbb{F}_{q^m})$ なる (x, y) の候補を 3 個に絞る $\log q$ に関する (確率的) 多項式時間アルゴリズムが存在する.

[証明] 連立方程式

$$\begin{aligned} y + \lambda_2 x + \lambda_0 &= 0 \qquad (1) \\ y^2 &= x^3 + ax + b \end{aligned}$$

より, x に関する 3 次方程式

$$x^3 - \lambda_2^2 x^2 + (a - 2\lambda_2\lambda_0)x + (b - \lambda_0^2) = 0 \qquad (2)$$

を得るので, これを $\mathbb{F}_{q^m}$ 上で解いて x の候補を 3 つ得ることが出来る. それぞれに対応する y は (1) 式より一意に定まる. 方程式の求解には解の公式や一般的な有限体上多項式の因数分解アルゴリズムを用いれば良い [Ber70, Sho05]. □

Remark 3 (2) 式の解と係数の関係により,

$$\mathrm{Tr}_{\mathbb{F}_{q^m}/\mathbb{F}_q}(x) = \lambda_2^2$$

である. 従って $\mathrm{Tr}_{\mathbb{F}_{q^m}/\mathbb{F}_q}(x)$ は $\mathbb{F}_q$ 上平方剰余である.

Theorem 1 $m = 3$ のとき $\log q$ に関する 2 つの多項式時間アルゴリズム

$$\begin{aligned} &\texttt{encode} : \mathbb{F}_{q^m}^2 \to \mathbb{F}_q^{m-1} \times \{1, 2, 3\} \\ &\texttt{decode} : \mathbb{F}_q^{m-1} \times \{1, 2, 3\} \to \mathbb{F}_{q^m}^2 \end{aligned}$$

が存在し, $P = (x, y) \in (\phi - 1)E(\mathbb{F}_{q^m})$ なるとき,

$$\texttt{decode}(\texttt{encode}(x, y)) = (x, y)$$

が成立する.

[証明] 以下に具体的アルゴリズムの構成を示す.

`encode`:

入力: $(x, y) \in \mathbb{F}_{q^m}^2$
step 0: $(x, y) \notin (\phi - 1)E(\mathbb{F}_{q^m})$ なら例外処理.
step 1: $\lambda_2 = -\dfrac{y^q - y}{x^q - x}$, $\lambda_0 = -y - \lambda_2 x$ とする.
step 2: `root[]` $= \{x, x^q, x^{q^2}\}$.
step 3: `root[]` を適当な順序でソート.
step 4: $x =$ `root[`i`]` なる i を見つける.
出力: $(\lambda_0, \lambda_2, i) \in \mathbb{F}_q^{m-1} \times \{1, 2, 3\}$

`decode`:

入力: $(\lambda_0, \lambda_2, i) \in \mathbb{F}_q^{m-1} \times \{1, 2, 3\}$
step 1: `root[]` $= \{(2)$ の解 $\in \mathbb{F}_{q^m}\}$.
step 2: `root[]` を上記の順序でソート.
step 3: $x =$ `root[`i`]`.
step 4: $y = -\lambda_2 x - \lambda_0$.
step 5: $(x, y) \notin (\phi - 1)E(\mathbb{F}_{q^m})$ なら例外処理.
出力: $(x, y) \in \mathbb{F}_{q^m}^2$

□

Remark 4 暗号の安全性を保証するために符号化や復号の例外処理は注意深く取り扱うべきである.

Remark 5 pairing 用の ordinary の楕円曲線で embedding degree が 3 の倍数の時に, 同様のテクニックが使える. しかし, 特殊な曲線に対しては twist を使ったエレガントな方法 (point reduction でかつ圧縮率も高い) が提案されている [HSV06, BN05].

4 一般の場合 ($m \geq 2$)

3 次拡大の場合は $P, \phi P, \phi^2 P$ を通過する直線が小さい情報量で指定出来たので, $P \in (\phi - 1)E(\mathbb{F}_{q^m})$ なる点 P の圧縮された表現を構成できた. 一般の場合も同様にして, 点 $P, \phi P, \ldots, \phi^{m-1} P$ を通過するような曲線を小さい情報量で指定する事が出来れば同様の圧縮を行えるはずである.

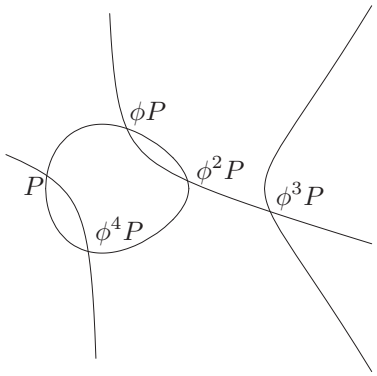

図 2: 一般の場合 (イメージ図)

本稿では X の重みを 2, Y の重みを 3 として X, Y の多項式の重み付きの次数について議論を行う. 誤解が生じないよう以降この重み付きの次数を単に “重み” と記述し, 重みを指定する際には “重み n の多項式” 等と記述する. “次数” あるいは単に “n 次式” と記述した場合は通常の意味の次数であるとする. また “モニック多項式” と記述した場合は, 重みの意味で最高次の係数が 1 の多項式とする. また $\mathbb{N}$ は 0 を含む自然数とする.
今, 点 $P, \phi P, \ldots, \phi^{m-1}P$ を通過するような曲線を指定したいので, これらの点を台に持つような因子 D_P を考える.

$$D_P = \sum_{i=0}^{m-1} (\phi^i P) - m(\mathcal{O})$$

Lemma 5 $P \in (\phi - 1)E(\mathbb{F}_{q^m}) \setminus \{\mathcal{O}\}$ なる P に対し $\mathrm{div}(f) = D_P$ なる $f \in \mathbb{F}_q[X, Y]$ が存在し, f は重み m のモニック多項式として良い.

[証明] (付録 A 参照) □

Remark 6 重み $i \in \mathbb{N}$ を持つ $\mathcal{O}$ 以外で正則な単項式を M_i とすると $i \neq 1$ のとき

$$M_i = \begin{cases} X^{i/2}, & i\text{が偶数のとき} \\ YX^{(i-3)/2}, & i\text{が奇数のとき} \end{cases}$$

と出来るが, M_1 は存在しないので, 重み m のモニック多項式の自由度は $\mathbb{F}_q^{m-1}$ である. 即ち, $m-1$ 個の係数 $\lambda_0, \lambda_2, \ldots, \lambda_{m-1} \in \mathbb{F}_q$ が与えられれば, 重み m のモニック多項式

$$M_m + \lambda_0 M_0 + \sum_{i=2}^{m-1} \lambda_i M_i$$

が一意に定まる.

Lemma 6 (encoding) $P \in (\phi - 1)E(\mathbb{F}_{q^m}) \setminus \{\mathcal{O}\}$ なる P に対し $\mathrm{div}(f) = D_P$ なる重み m のモニック多項式 $f \in \mathbb{F}_q[X, Y]$ を与える $\log q$ に関する多項式時間アルゴリズムが存在する.

[証明] (付録 B 参照) □

Lemma 7 (decoding) $\mathrm{div}(f) = D_P$ なる重み m のモニック多項式 $f \in \mathbb{F}_q[X, Y]$ に対し $P \in (\phi - 1)E(\mathbb{F}_{q^m}) \setminus \{\mathcal{O}\}$ なる P の候補を m 個に絞る $\log q$ に関する (確率的) 多項式時間アルゴリズムが存在する.

[証明] (付録 C 参照) □

Theorem 2 一般の m に対して $\log q$ に関する 2 つの (確率的) 多項式時間アルゴリズム

$$\texttt{encode} : \mathbb{F}_{q^m}^2 \to \mathbb{F}_q^{m-1} \times \{1, \ldots, m\}$$
$$\texttt{decode} : \mathbb{F}_q^{m-1} \times \{1, \ldots, m\} \to \mathbb{F}_{q^m}^2$$

が存在し, $P = (x, y) \in (\phi - 1)E(\mathbb{F}_{q^m})$ なるとき,

$$\texttt{decode}(\texttt{encode}(x, y)) = (x, y)$$

が成立する.

[証明] 以下に具体的アルゴリズムの構成を示す.

`encode`:

入力: $P = (x, y) \in \mathbb{F}_{q^m}^2$
step 0: $(x, y) \notin (\phi - 1)E(\mathbb{F}_{q^m})$ なら例外処理.
step 1: $\mathrm{div}(f) = D_p$ なる f を Lemma 6 のアルゴリズムにより計算し, $m-1$ 個の f の係数 $\lambda_0, \lambda_2, \ldots, \lambda_{m-1} \in \mathbb{F}_q$ を求める.
step 2: `root[]` $= \{P, \phi P, \ldots, \phi^{m-1}P\}$.
step 3: `root[]` を適当な順序でソート.
step 4: $P =$ `root[`i`]` なる i を見つける.
出力: $(\lambda_0, \lambda_2, \ldots, \lambda_{m-1}, i) \in \mathbb{F}_q^{m-1} \times \{1, \ldots, m\}$

`decode`:

入力: $(\lambda_0, \lambda_2, \ldots, \lambda_{m-1}, i) \in \mathbb{F}_q^{m-1} \times \{1, \ldots, m\}$
step 1: `root[]` $=$ {Lemma 7 の候補 }.
step 2: `root[]` を上記の順序でソート.
step 3: $P =$ `root[`i`]`.
step 4: $P = (x, y) \notin (\phi - 1)E(\mathbb{F}_{q^m})$ なら例外処理.
出力: $(x, y) \in \mathbb{F}_{q^m}^2$

□

Remark 7 $m = 2$ の場合, 点 (x, y) に対する重み m のモニック多項式は $X - x$ である. 従って, この時 $x \in \mathbb{F}_q$ であり, $x^3 + ax + b \in \mathbb{F}_q$ は $\mathbb{F}_q$ 上平方非剰余となる. このような事実は以前から非常に良く知られていた.

Remark 8 decode を使って次のような確率的な平文メッセージの埋め込みアルゴリズムを構成できる. まず, $\mathbb{F}_q^{m-1} \times \{1, \ldots, m\}$ の部分情報に平文を埋め込み, 残りの部分情報をランダムに選んで decode が成功すれば良い. $(\phi - 1)E(\mathbb{F}_{q^m})$ の位数がおよそ q^{m-1} なのに対し $\mathbb{F}_q^{m-1} \times \{1, \ldots, m\}$ のバリエーションは mq^{m-1} であるから埋め込みが成功する確率はおよそ $1/m$ である. 従来法の確率はおよそ $1/q$ であったので $m \ll q$ の場合はこの方法の効率が良い.

[例] $m = 5$ の場合, 重み m の多項式は

$$XY, X^2, Y, X, 1$$

の線形結合で表現できる. 従ってこの場合の $\mathbb{F}_q$ 係数モニック多項式は $\lambda_4, \lambda_3, \lambda_2, \lambda_0 \in \mathbb{F}_q$ を用いて

$$XY + \lambda_4 X^2 + \lambda_3 Y + \lambda_2 X + \lambda_0$$

と書くことが出来る. 従って $P = (x, y) \in (\phi - 1)E(\mathbb{F}_{q^m})$ に対して Lemma 6 のアルゴリズム (付録 B 参照) を実行すれば

$$xy + \lambda_4 x^2 + \lambda_3 y + \lambda_2 x + \lambda_0 = 0$$

なる $\lambda_4, \lambda_3, \lambda_2, \lambda_0 \in \mathbb{F}_q$ を計算することが出来る.

5 まとめと謝辞

本研究では, 比較的大きい部分体 $\mathbb{F}_q$ 上定義された楕円曲線 $E(\mathbb{F}_{q^m})$ 対し点 (x, y) がそのある部分群の元であるとき, $\mathbb{F}_{q^m}^2 \to \mathbb{F}_q^{m-1} \times \{1, \ldots, m\}$ なる点圧縮を実現する $\log q$ に関する 2 つの多項式時間アルゴリズム encode および decode を提案した. この部分群の元を表現するのに要する bit 数の情報理論的下界はおよそ $(m-1)\log_2 q$ であるが, 上記のアルゴリズムでは, この元は, およそ $(m-1)\log_2 q + \log_2 m$ bit で表現される. 従って本方法は $m \ll q$ の場合に適している.
また m が小さいとき decode を使うと効率的な確率的平文メッセージの埋め込みアルゴリズムを構成できる.
本研究に関して, 青木和麻呂氏, 鈴木幸太郎氏, 山本剛氏, 安田幹氏に幾つかの議論に参加頂き, 有益な意見を頂いた.

参考文献

[Ber70] E. Berlekamp, Factoring polynomials over large finite fields, Math. Comp. 24(1970), 713-735

[BN05] P. Barreto and M. Naehrig, "Pairing-friendly elliptic curves of prime order," Cryptology ePrint Archive: Report 2005/133, http://eprint.iacr.org/2005/133

[HKA06] F.Hoshino, T.Kobayashi and K.Aoki, "Compressed Jacobian Coordinates for OEF," VIETCRYPT 2006, LNCS 4341, pp.147-156, 2006. Springer-Verlag Berlin Heidelberg 2006

[HSV06] F.Hess, N.Smart, and F.Vercauteren, "The Eta pairing revisited," IEEE Trans. Information Theory, Vol 52, pp 4595-4602, 2006.

[IMCT02] T.Iijima, K.Matsuo, J.Chao and S.Tsujii, "Construction of Frobenius maps of twist elliptic curves and its application to elliptic scalar multiplication," Proc. of SCIS2002, IEICE Japan, January 2002, pp.699-702.

[Kob94] Neal Koblitz, "A Course in Number Theory and Cryptography, 2nd edition", §6.2, 1994 Splinger-Verlag

[Mil86] V. Miller, "Short program for functions on curves," unpublished manuscript, 1986, http://crypto.stanford.edu/miller/miller.pdf

[Sec00] SECG, "Recommended Elliptic Curve Domain Parameters," SEC 2, 2000, http://www.secg.org/

[Ser98] Cadiel Seroussi, Compact Representation of Elliptic Curve Points over $\mathbb{F}_{2^n}$, April 1998. Research Manuscript, Hewlett-Packard Laboratories,.

[Sho05] V.Shoup, A Computational Introduction to Number Theory and Algebra, Version 1, Cambridge University Press, 2005, http://www.shoup.net/ntb/

[Sil86] J.H.Silverman, The arithmetic of elliptic curves, volume 106 of Graduate Texts in Mathematics. Springer-Verlag, New York, 1986

付録 A

Lemma 5 $P \in (\phi - 1)E(\mathbb{F}_{q^m}) \setminus \{\mathcal{O}\}$ なる P に対し $\mathrm{div}(f) = D_P$ なる $f \in \mathbb{F}_q[X, Y]$ が存在し, f は重み m のモニック多項式として良い.

[証明] $\deg D_P = 0$ であり, $P \in (\phi - 1)E(\mathbb{F}_{q^m})$, Lemma 2 より

$$\sum_{i=0}^{m-1} \phi^i P = \mathcal{O}$$

なので, $\mathrm{div}(f) = D_P$ なる $f \in \overline{\mathbb{F}}_q(X, Y)$ が存在する. ところで, 因子 D に対して

$$\mathcal{L}(D) = \{g \in \overline{\mathbb{F}}_q(E)^* : \mathrm{div}(g) \geq -D\} \cup \{0\}$$

とすると, $f \in \mathcal{L}(m(\mathcal{O}))$ である. $\mathcal{L}(m(\mathcal{O}))$ は有限次元 $\overline{\mathbb{F}}_q$-ベクトル空間であり, Riemann-Roch の定理より

$$\dim_{\overline{\mathbb{F}}_q} \mathcal{L}(m(\mathcal{O})) = m$$

であるから, f は m 個の独立な基底関数の線形結合でかける [Sil86]. $\mathcal{L}(m(\mathcal{O}))$ の基底関数として重み m 以下の

$$\{X^i, X^j Y : i, j \in \mathbb{N}\}$$

を選ぶことが出来るので, $f \in \overline{\mathbb{F}}_q[X, Y]$ で f の重みは m 以下と出来る. さらに, σ を $\overline{\mathbb{F}}_q$ の q 乗 Frobenius 写像とすると

$$D_P^\sigma = \sum_{i=0}^{m-1} (\phi^{i+1} P) - m(\mathcal{O}) = \sum_{i=0}^{m-1} (\phi^i P) - m(\mathcal{O}) = D_P$$

であるから, $f \in \mathbb{F}_q[X, Y]$ である. また, 全ての i に関して $\phi^i P \neq \mathcal{O}$ であるから, $f \notin \mathcal{L}((m-1)(\mathcal{O}))$ と出来る. 従って $f \in \mathbb{F}_q[X, Y]$ を重み m のモニック多項式として良い. □

付録 B

Lemma 6 (encoding) $P \in (\phi-1)E(\mathbb{F}_{q^m}) \setminus \{\mathcal{O}\}$ なる P に対し $\mathrm{div}(f) = D_P$ なる重み m のモニック多項式 $f \in \mathbb{F}_q[X,Y]$ を与える $\log q$ に関する多項式時間アルゴリズムが存在する.

[証明] 基本的にペアリングを計算する為の Miller のアルゴリズムと同様の手続きを $\mathbb{F}_{q^m}(X,Y)$ 上で行えばよい [Mil86]. 計算の便宜のため以下を定義する.

$$Q_n = \sum_{i=0}^{n-1} \phi^i P$$
$$\mathrm{div}(f_n) = \sum_{i=0}^{n-1} (\phi^i P) - (Q_n) - (n-1)(\mathcal{O})$$

$Q_n = (x_n, y_n), \phi^n P = (x'_n, y'_n)$ とし l_n, v_n を

$$l_n = (x'_n - x_n)(Y - y_n) - (y'_n - y_n)(X - x_n)$$
$$v_n = (X - x_{n+1})$$

とすると,

$$\mathrm{div}(l_n) = (Q_n) + (\phi^n P) + (-Q_{n+1}) - 3(\mathcal{O})$$
$$\mathrm{div}(v_n) = (-Q_{n+1}) + (Q_{n+1}) - 2(\mathcal{O})$$

であるから,

$$f_{n+1} = \frac{f_n l_n}{v_n} \tag{3}$$

である. また $\mathrm{div}(f_1) = (P) - (P) - 0(\mathcal{O})$ であるから

$$f_1 = 1 \tag{4}$$

と出来る. $f = f_m$ であるから (3), (4) の漸化式を Y の 2 次以上の項を楕円の定義式で還元し, 分母と分子を通分しながら $m-1$ 回繰り返して f を求めることが出来る. この過程の計算量は $\mathbb{F}_{q^m}$ 上演算 $O(m^2)$ 回である. □

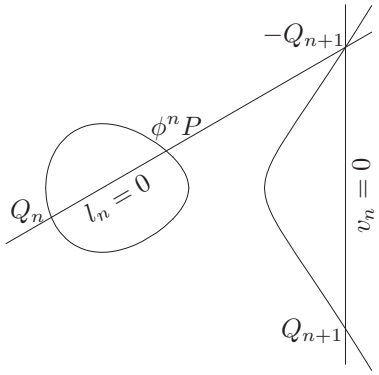

図 3: f の計算 (イメージ図)

Remark 9 実際の f の計算は必ずしも上記の方法で行う必要は無い. 例えば

$$\mathrm{div}(f_n) = \sum_{i=0}^{n-1} (\phi^i P) - (Q_n) - (n-1)(\mathcal{O})$$

に対して

$$\mathrm{div}(f_n^{\sigma^n}) = \sum_{i=n}^{2n-1} (\phi^i P) - (\phi^n Q_n) - (n-1)(\mathcal{O})$$

であるから,

$$\mathrm{div}(l'_n) = (Q_n) + (\phi^n Q_n) + (-Q_{2n}) - 3(\mathcal{O})$$
$$\mathrm{div}(v'_n) = (-Q_{2n}) + (Q_{2n}) - 2(\mathcal{O})$$

なる l'_n, v'_n を使えば

$$f_{2n} = \frac{f_n f_n^{\sigma^n} l'_n}{v'_n}$$

等とできる.

付録 C

Lemma 7 (decoding) $\mathrm{div}(f) = D_P$ なる重み m のモニック多項式 $f \in \mathbb{F}_q[X,Y]$ に対し $P \in (\phi-1)E(\mathbb{F}_{q^m}) \setminus \{\mathcal{O}\}$ なる P の候補を m 個に絞る $\log q$ に関する (確率的) 多項式時間アルゴリズムが存在する.

[証明] $g \in \mathbb{F}_q[X]$ を $\lfloor (m-3)/2 \rfloor$ 次多項式 $h \in \mathbb{F}_q[X]$ を $\lfloor m/2 \rfloor$ 次多項式とする. $f(X,Y) = g(X)Y + h(X)$ と書くことが出来る. $g(x) \neq 0$ の時, 連立方程式

$$g(x)y + h(x) = 0 \tag{5}$$
$$y^2 = x^3 + ax + b \tag{6}$$

より, x に関する m 次方程式

$$g(x)^2(x^3 + ax + b) - h(x)^2 = 0 \tag{7}$$

を得るので, これを $\mathbb{F}_{q^m}$ 上で解いて x の候補を m 個得ることが出来, それぞれに対応する y は (5) 式より一意に定まる. $g(x) = 0$ の時 m は必ず偶数で, (5) 式 $h(x) = 0$ を $\mathbb{F}_{q^m}$ 上で解いて x の候補を $m/2$ 個 得ることが出来, それぞれに対応する y は (6) より 2 個づつ得ることが出来る. 方程式の求解には一般的な有限体上の多項式の因数分解アルゴリズムを用いれば良い [Ber70, Sho05]. この因数分解の計算量の期待値は $\mathbb{F}_{q^m}$ 上演算 $O((m^3 \log m) \log q)$ 回である [Sho05]. □

Remark 10 (7) 式が $\mathbb{F}_{q^m}$ 上で x の一次式の積に分解される事が分かっているので, 一般の有限体係数多項式の因数分解より単純な問題である.