数学アルゴリズム問題の研究調査

－有限体の乗法群に関する離散対数問題－

２００１年度

数学アルゴリズム問題の研究調査
— 有限体の乗法群に関する離散対数問題 —

1 はじめに

Diffie-Hellman [6] により "公開鍵暗号方式" の概念が提案されて依頼，その安全性の根拠として，数論に関連するいくつかの問題— 素因数分解・離散対数等 — が注目を浴びてきている．特に，素因数分解の困難さに基づく公開鍵暗号方式としては，

F-1: RSA 暗号方式 [16]
F-2: Rabin 暗号方式 [15]
F-3: Williams 暗号方式 [24, 25]
F-4: 逆数暗号方式 [9]

等が広く知られている．一方，離散対数問題 (特に，有限体の離散対数問題) の困難さに基づく公開鍵暗号方式 (とそれに関連する方式) としては，

D-1: Diffie-Hellman 公開鍵共有方式 [6]
D-2: El-Gamal 公開鍵方式 [7]
D-3: Bellare-Micali 非対話型忘却伝送方式 [2]
D-4: Okamoto 会議用鍵共有方式 [10]
D-5: Shamir 鍵配送 3-パス方式 [20, 21]

等が知られており，暗号プロトコルの安全性の観点から，れらの有効性が広く認められている．

本研究調査では，有限体の乗法群に関する離散対数問題に対して，

(1) 効率的な解法アルゴリズムが存在するための充分条件;
(2) 一般の場合における現状での最良の解法アルゴリズム;
(3) 公開鍵暗号方式の安全性を保証するための方法・条件

を検討し，有限体の乗法群に関する離散対数問題の効果等について検証することを目的とする．

2 準備

本研究調査では (特に断らない限り)，素数を p で表すこととする．また $a \equiv b \pmod{p}$ は，a を p で割った余りが b であることを表すものとする．

定義 2.1 (剰余系 $\mathbf{Z}_n$): 任意の整数 n に対して，$\mathbf{Z}_n$ を

$$\mathbf{Z}_n = \{0, 1, 2, \ldots, n-1\}$$

と定義し，これを (法 n に関する) 剰余系と呼ぶ．

定義 2.2 (有限体 $\mathbf{Z}_p$): 素数 p に対して，$\mathbf{Z}_p = \{0, 1, 2, \ldots, p-1\}$ を要素数 p の有限体と呼ぶ.

定義 2.3 (既約剰余系 $\mathbf{Z}_n^*$): 任意の整数 n に対して，$\mathbf{Z}_n^*$ を

$$\mathbf{Z}_n^* = \{x \in \mathbf{Z}_n : \gcd(x, n) = 1\}$$

と定義し，これを (法 n に関する) 既約剰余系と呼ぶ.

定義 2.4 (有限体の乗法群 $\mathbf{Z}_p^*$): 素数 p に対して，$\mathbf{Z}_p^*$ を

$$\mathbf{Z}_p^* = \mathbf{Z}_p - \{0\} = \{1, 2, \ldots, p-1\}$$

と定義し，これを有限体 $\mathbf{Z}_p$ の乗法群と呼ぶ.

定義 2.5 (位数): 任意の $g \in \mathbf{Z}_p^*$ に対して，

$$g^e \equiv 1 \pmod{p}$$

を満たす最小の正整数 e を $g \in \mathbf{Z}_p^*$ の位数と呼び，$e = \mathrm{ord}(g; p)$ と表す.

定義 2.6 (有限体 $\mathbf{Z}_p^*$ の原始根): 任意の $g \in \mathbf{Z}_p^*$ に対して，$\mathrm{ord}(g; p) = p - 1$ となるとき，$g \in \mathbf{Z}_p^*$ を (素数 p んに関する) 原始根 — Primitive Root — と呼ぶ.

定義2.6 より，$g \in \mathbf{Z}_p^*$ が原始根であるための必要十分条件は，全ての $i, j \in \mathbf{Z}_{p-2}$ $(i \neq j)$ に対して $g^i \not\equiv g^j \pmod{p}$ が成り立つことであることが容易に導かれる．有限体の乗法群における離散対数問題 (DLP: Discrete Logarithm Problem) は以下のように定義される.

定義 2.7 (離散対数問題): 素数 p に対して，任意の $g \in \mathbf{Z}_p^*$ と任意の $y \in \mathbf{Z}_p^*$ が与えられたとき，

$$y \equiv g^x \pmod{p} \tag{1}$$

を満たす最小の $x \in \mathbf{Z}_{p-1}$ を求める問題 (そのような $x \in \mathbf{Z}_{p-1}$ が存在するならば) を，有限体の乗法群における離散対数問題 — DLP: Discrete Logarithm Problem — と呼ぶ.

定義2.7 において，$g \in \mathbf{Z}_p^*$ を原始根とすると，式 (1) を満たす $x \in \mathbf{Z}_{p-1}$ の存在が保証される．しかし一般には，式 (1) を満たす $x \in \mathbf{Z}_{p-1}$ の存在は，必ずしも保証されない．本研究調査では (特に断らない限り)，$g \in \mathbf{Z}_p^*$ は原始根を表すものとして議論を進めることとする．また式 (1) において，素数 p を法とするのではなく，合成数 n を法として定義することも可能であるが，この場合は，特に離散対数問題と区別するために指数計算問題と呼ぶこととする.

3 特殊な場合における離散対数問題に対するアルゴリズム

3.1 Pohlig-Hellman のアルゴリズム

本節では，有限体の乗法群における離散対数問題に関して，ある条件の元での効率的な解法アルゴリズム — Pohlig-Hellman 法 [14] — について述べる.

定義 3.1 (*b*-平坦): 整数 n の素因数分解を $n = q_1^{e_1} q_2^{e_2} \cdots q_k^{e_k}$ とするとき，整数 n が "b-平坦" であるとは，全ての $1 \leq i \leq k$ に対して "$q_i \leq b$" が成り立つことを言う．

Pohlig-Hellman 法 [14] の基本となる以下の定理 — フェルマーの定理 — を示す．

定理 3.2 (フェルマーの定理): 任意の $a \in \mathbf{Z}_p^*$ に対して $a^{p-1} \equiv 1 \pmod{p}$ が成り立つ．

ここで，素数 p に対して $p-1 = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ が b-平坦である場合を考える．まず簡単のために，素数 p に対して $p-1 = 2^\ell$ とであると仮定する (一般の場合は後述)．

原始根 $g \in \mathbf{Z}_p^*$ と $y \in \mathbf{Z}_p^*$ に対して，明らかに

$$y \equiv g^x \pmod{p}$$

を満たす整数 $x \in \mathbf{Z}_{p-1}$ が存在するので，これを 2 進数展開する．

$$x_0 = x = a_{\ell-1} 2^{\ell-1} + a_{\ell-2} 2^{\ell-2} + \cdots + a_1 2^1 + a_0 2^0. \tag{2}$$

このとき，定理 3.2 と $g \in \mathbf{Z}_p^*$ が原始根であることから，$y_0 = y$ に対して，

$$\begin{aligned}
y_0^{\frac{p-1}{2}} &\equiv y^{\frac{p-1}{2}} \equiv g^{x\frac{p-1}{2}} \equiv g^{x 2^{\ell-1}} \pmod{p} \\
&\equiv g^{(a_{\ell-1}2^{\ell-1} + a_{\ell-2}2^{\ell-2} + \cdots + a_1 2^1 + a_0 2^0) 2^{\ell-1}} \pmod{p} \\
&\equiv g^{(a_{\ell-1}2^{\ell-2} + a_{\ell-2}2^{\ell-3} + \cdots + a_1 2^0) 2^{\ell} + a_0 2^{\ell-1}} \pmod{p} \\
&\equiv \left(g^{2^\ell}\right)^{a_{\ell-1}2^{\ell-2} + a_{\ell-2}2^{\ell-3} + \cdots + a_1 2^0} \cdot \left(g^{2^{\ell-1}}\right)^{a_0} \pmod{p} \\
&\equiv \left(g^{p-1}\right)^{a_{\ell-1}2^{\ell-2} + a_{\ell-2}2^{\ell-3} + \cdots + a_1 2^0} \cdot \left(g^{\frac{p-1}{2}}\right)^{a_0} \pmod{p} \\
&\equiv (-1)^{a_0} \pmod{p}
\end{aligned} \tag{3}$$

が成り立つ．これより，式 (2) において，その最下位ビット a_0 の値は，式 (3) より，

$$a_0 = \begin{cases} 0 & y_0^{\frac{p-1}{2}} \equiv 1 \pmod{p}; \\ 1 & y_0^{\frac{p-1}{2}} \equiv -1 \pmod{p} \end{cases} \tag{4}$$

によって効率的に判定することができる．ここで式 (2) と y_0 より，y_1 を以下のように定義する．

$$y_1 \equiv y_0 \cdot g^{-a_0 2^0} \equiv g^{x_0 - a_0 2^0} \equiv g^{a_{\ell-1}2^{\ell-1} + a_{\ell-2}2^{\ell-2} + \cdots + a_1 2^1} \pmod{p}. \tag{5}$$

式 (3) と同様に，式 (5) と定理 3.2 および $g \in \mathbf{Z}_p^*$ が原始根であることから，

$$\begin{aligned}
y_1^{\frac{p-1}{4}} &\equiv g^{(a_{\ell-1}2^{\ell-1} + a_{\ell-2}2^{\ell-2} + \cdots + a_1 2^1)\frac{p-1}{4}} \pmod{p} \\
&\equiv g^{(a_{\ell-1}2^{\ell-1} + a_{\ell-2}2^{\ell-2} + \cdots + a_1 2^1) 2^{\ell-2}} \pmod{p} \\
&\equiv g^{(a_{\ell-1}2^{\ell-3} + a_{\ell-2}2^{\ell-4} + \cdots + a_2 2^0) 2^{\ell} + a_1 2^{\ell-1}} \pmod{p} \\
&\equiv \left(g^{2^\ell}\right)^{a_{\ell-1}2^{\ell-3} + a_{\ell-2}2^{\ell-4} + \cdots + a_2 2^0} \cdot \left(g^{2^{\ell-1}}\right)^{a_1} \pmod{p} \\
&\equiv \left(g^{p-1}\right)^{a_{\ell-1}2^{\ell-3} + a_{\ell-2}2^{\ell-4} + \cdots + a_2 2^0} \cdot \left(g^{\frac{p-1}{2}}\right)^{a_1} \pmod{p} \\
&\equiv (-1)^{a_1} \pmod{p}
\end{aligned} \tag{6}$$

が成り立つ．これより，式 (2) において，その下位第 2 ビット a_1 の値は，式 (6) より，

$$a_1 = \begin{cases} 0 & y_1^{\frac{p-1}{4}} \equiv 1 \pmod{p}; \\ 1 & y_1^{\frac{p-1}{4}} \equiv -1 \pmod{p} \end{cases} \tag{7}$$

によって効率的に判定することができる．以下同様に，式 (2) において，その下位第 j ビット a_j の値は，$y_j \equiv y_{j-1} \cdot g^{-a_{j-1}2^{j-1}} \pmod{p}$ に対して，

$$a_j = \begin{cases} 0 & y_j^{\frac{p-1}{2^{j+1}}} \equiv 1 \pmod{p}; \\ 1 & y_j^{\frac{p-1}{2^{j+1}}} \equiv -1 \pmod{p} \end{cases} \tag{8}$$

を計算することによって，効率的に判定することができるので，$y \equiv g^x \pmod{p}$ の離散対数 $x \in \mathbf{Z}_p^*$ が効率的に計算可能となる．

次に，素数 p に対して，$p-1 = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ が b-平坦である場合について述べる．

定理 3.3 (中国人の剰余定理): 整数 $m_1, m_2, \ldots, m_k$ に対して，$\gcd(m_i, m_j) = 1 \ (i \neq j)$ が成り立つものとする．このとき，任意に与えられた $a_1 \in \mathbf{Z}_{m_1}, a_2 \in \mathbf{Z}_{m_2}, \ldots, a_k \in \mathbf{Z}_{m_k}$ に対して，

$$x \equiv \begin{cases} a_1 \pmod{m_1} \\ a_2 \pmod{m_2} \\ \quad \vdots \\ a_k \pmod{m_k} \end{cases}$$

を満たす $x \in \mathbf{Z}_M$ を求める効率的なアルゴリズムが存在する (ただし $M = m_1 m_2 \cdots m_k$ とする)．また，そのような $x \in \mathbf{Z}_M$ は唯一である．

素数 p に対して，$p-1 = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ であることから，$\gcd(p_i^{e_i}, p_j^{e_j}) = 1 \ (i \neq j)$ が成り立つことに注意する．ここで，原始根 $g \in \mathbf{Z}_p^*$ と $y \in \mathbf{Z}_p^*$ に対して，$y \equiv g^x \pmod{p}$ を満たす $x \in \mathbf{Z}_p^*$ を求める問題を考える．全ての $1 \leq i \leq k$ に対して，

$$q_i = \frac{p-1}{p_i^{e_i}}; \tag{9}$$

$$x_i \equiv x \pmod{p_i^{e_i}}; \tag{10}$$

$$g_i \equiv g^{q_i} \pmod{p} \tag{11}$$

を定義する．このとき，全ての $1 \leq i \leq k$ に対して

$$y_i \equiv y^{q_i} \equiv g^{xq_i} \equiv (g^{q_i})^{x \pmod{p_i^{e_i}}} \equiv (g^{q_i})^{x_i} \equiv g_i^{x_i} \pmod{p} \tag{12}$$

が成り立つ．従って，$y_i \equiv g_i^{x_i} \pmod{p}$ を満たす $x_i \in \mathbf{Z}_{p_i^{e_i}-1}$ を効率的に求めることができれば，定理 3.3 を用いて，原始根 $g \in \mathbf{Z}_p^*$ と $y \in \mathbf{Z}_p^*$ に対して，$y \equiv g^x \pmod{p}$ を満たす $x \in \mathbf{Z}_{p-1}$ を効率的に求めることが可能となる．以下では，$p-1 = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$ が b-平坦であることを用いて，式 (12) を効率的に求めるアルゴリズムについて述べる．

まず，式 (9) と式 (11) および $g \in \mathbf{Z}_p^*$ が原始根であることより，$g_i^e \equiv 1 \pmod{p}$ を満たす最小の正整数は $p_i^{e_i}$ となる．ここで，$x_i \in \mathbf{Z}_{p_i^{e_i}-1}$ であることから，x_i を p_i 進数表示する．

$$x_i = a_{e_i-1} p_i^{e_i-1} + a_{e_i-2} p_i^{e_i-2} + \cdots + a_1 p_i^1 + a_0 p_i^0. \tag{13}$$

このとき，定理 3.2 と $g \in \mathbf{Z}_P^*$ が原始根であることから，$y_{i,0} = y_i$ に対して，

$$\begin{aligned}
y_{i,0}^{p_i^{e_i-1}} &\equiv y_i^{p_i^{e_i-1}} \equiv g_i^{x_i p_i^{e_i-1}} \pmod{p} \\
&\equiv g_i^{(a_{e_i-1}p_i^{e_i-1}+a_{e_i-2}p_i^{e_i-2}+\cdots+a_1p_i^1+a_0p_i^0)p_i^{e_i-1}} \\
&\equiv g_i^{(a_{e_i-1}p_i^{e_i-2}+a_{e_i-2}p_i^{e_i-3}+\cdots+a_1p_i^0)p_i^{e_i}+a_0p_i^{e_i-1}} \\
&\equiv \left(g_i^{p_i^{e_i}}\right)^{a_{e_i-1}p_i^{e_i-2}+a_{e_i-2}p_i^{e_i-3}+\cdots+a_1p_i^0} \cdot \left(g_i^{p_i^{e_i-1}}\right)^{a_0} \\
&\equiv \left(g_i^{p_i^{e_i-1}}\right)^{a_0} \qquad (14)
\end{aligned}$$

が成り立つ．従って，式 (14) と $g_i^e \equiv 1 \pmod{p}$ を満たす最小の正整数が $p_i^{e_i}$ であることから，式 (13) において，その最下位ディジット a_0 の値は，テーブル T_i

$$T_i = \left\{ g_i^{p_i^{e_i-1} \cdot j} : 0 \le j \le p_i - 1 \right\}$$

を検索することにより求めることが可能である．ここで $p_i \le b$ であることから，テーブル T_i のサイズは高々 b であるので，テーブル T_i の検索は効率的に実行可能である．以下，

$$y_{i,j} \equiv y_{i,j-1} \cdot g_i^{-a_{j-1}p_i^{j-1}} \pmod{p}$$

を再帰的に計算し，式 (14) と同様な手順により，

$$\begin{aligned}
y_{i,j}^{p_i^{e_i-j-1}} &\equiv y_{i,j-1}^{p_i^{e_i-j-1}} \pmod{p} \\
&\equiv g_i^{(a_{e_i-1}p_i^{e_i-1}+a_{e_i-2}p_i^{e_i-2}+\cdots+a_jp_i^j)p_i^{e_i-j-1}} \\
&\equiv g_i^{(a_{e_i-1}p_i^{e_i-j}+a_{e_i-2}p_i^{e_i-j-1}+\cdots+a_{j+1}p_i^0)p_i^{e_i}+a_jp_i^{e_i-1}} \\
&\equiv \left(g_i^{p_i^{e_i}}\right)^{a_{e_i-1}p_i^{e_i-j}+a_{e_i-2}p_i^{e_i-j-1}+\cdots+a_{j+1}p_i^0} \cdot \left(g_i^{p_i^{e_i-1}}\right)^{a_j} \\
&\equiv \left(g_i^{p_i^{e_i-1}}\right)^{a_j} \qquad (15)
\end{aligned}$$

となることが導かれる．従って，式 (15) より，式 (13) において，その下位第 j ディジット a_j の値は，テーブル T_i を検索することにより効率的に求めることが可能である．

よって，全ての $1 \le i \le k$ に対して，式 (12) を満たす $x_i \in \mathbf{Z}_{p^{e_i}-1}$ を効率的に計算可能となるので，定理 3.3 より，$y \equiv g^x \pmod{p}$ を満たす $x \in \mathbf{Z}_{p-1}$ を効率的に求めることが可能となる．

3.2 佐藤・荒木アルゴリズム

本来，佐藤・荒木アルゴリズム [19] は，楕円曲線 (特に anomalous な曲線) 上の離散対数問題に対する攻撃アルゴリズムとして提案されたものであるが，同様のアイディアは，楕円曲線 (特に anomalous な曲線) 上の離散対数問題以外にも適用可能である．実際，佐藤・荒木アルゴリズムは p^2 を法とする特殊な離散対数問題 (指数計算問題) に対する効率的な攻撃アルゴリズムとしても利用可能である．また，佐藤・荒木アルゴリズと同様なアルゴリズムが，Smart [23] と Semaev [17] によって，ほぼ同時期にそれぞれ独立に提案されている．

素数 p が与えられたとき，任意の $g \in \mathbf{Z}_{p^2}^*$ に対して，

$$y \equiv g^e \pmod{p^2} \tag{16}$$

を満たす最小の正整数 $e \in \mathbf{Z}_{p(p-1)}$ が p であるとする．このとき，任意の $y \in \mathbf{Z}_{p^2}^*$ に対して，

$$y \equiv g^x \pmod{p^2} \tag{17}$$

を満たす最小の正整数 $x \in \mathbf{Z}_p$ を求める問題を考える．このような条件を満たすとき，$y \in \mathbf{Z}_{p^2}^*$ の離散対数 $x \in \mathbf{Z}_p$ は，以下の式により求めることができる．

$$x \equiv \frac{y-1}{g-1} \pmod{p}. \tag{18}$$

明らかに，式 (18) は多項式時間計算可能である．また，この事実を巧みに利用した公開鍵暗号方式として，EPOC [12] が知られている．

3.3 公開鍵暗号方式の安全性に対する影響

一般に，任意の素数 p が与えられたとき，定数 b に対して，$p-1$ の素因数分解

$$p - 1 = p_1^{e_1} p_2^{e_2} \cdots p_k^{e_k}$$

が b-平坦となる場合は，極めて稀である．従って，有限体の乗法群における離散対数問題に基づいて暗号プロトコルを構成する際に，Pohlig-Hellman 法による攻撃を想定する必要性は薄い．しかし，極めて稀に $p-1$ が b-平坦となるような素数 p が存在するので*，離散対数問題を基に暗号プロトコルを設計する際には，素数の選択には若干の配慮が必要となる．

一方，佐藤・荒木アルゴリズムに関しては，第 3.2 でも述べたように，極めて特殊な例であるので，離散対数問題に基づく暗号プロトコル設計の際の大きな障害とはなり得ない．

4 一般的な場合における離散対数問題に対するアルゴリズム

本章では，有限体の乗法群における離散対数問題に関して，一般的な場合 (素数に関して制約のない) における解法アルゴリズムについて述べる．

4.1 列挙法

式 (1) の $y \in \mathbf{Z}_p^*$ から $x \in \mathbf{Z}_{p-1}$ を求める最も素朴で単純な方法 (列挙法) は，$x \in \mathbf{Z}_{p-1}$ を $x = 0, 1, 2, \ldots, p-2$ のように順次列挙し，その都度，式 (1) を満たすかどうかを検証する方法である．この方法では，$g \in \mathbf{Z}_p^*$ が原始根である場合は，常に離散対数を求めることが可能であるが，$g \in \mathbf{Z}_p^*$ が原始根でない場合は，式 (1) を満たす $x \in \mathbf{Z}_p^*$ が存在しない可能性がある．

一般に列挙法においては，全ての $x \in \mathbf{Z}_{p-1}$ に対して，$y \equiv g^x \pmod{p}$ を計算する必要があるので，その計算量は $O(2^n n \lg n \lg\lg n)$ となる (ただし $n = \lg p$ とする)．

*例えば，$p = 2^{448}5^2 + 1$ は素数であり，$p - 1 = 2^{448}5^2$ は 5-平坦である [14]．

4.2 Shanks 法: Baby-Step/Giant-Step

列挙法が離散対数を一次元的に探索する手法であるのに対し，Shanks 法 [22] は，離散対数を二次元的に探索する手法と位置付けることができる．Shanks 法は列挙法に比べて記憶領域は増大するもの，その実行時間は $O(2^{n/2} n \lg n \lg\lg n)$ となる．その基本的なアイディアは，素数 p に対して $m = \lceil \sqrt{p} \rceil$ とし，式 (1) を満たす $x \in \mathbf{Z}_{p-1}$ に対して，

$$x \equiv qm + r \ (0 \leq r \leq m) \tag{19}$$

を満たす (除数 m に関する) 商 q と剰余 r を一意に定め，$\langle q, r \rangle$ を二次元的に効率的に探索することにより，$y \in \mathbf{Z}_p^*$ の離散対数を効率的に求めるというものである．

まず，式 (1) と式 (19) より，$y \equiv g^x \equiv g^{qm+r} \pmod{p}$ となることから，

$$(g^m)^q \equiv y \cdot g^{-r} \pmod{p} \tag{20}$$

が成り立つことに注意する．そこで，はじめに Baby-Step として，集合 B を

$$B = \{\langle y \cdot g^{-r}, r \rangle : 0 \leq r < m\} \tag{21}$$

のように定義する．ここで $q = 0$ であるならば，$\langle 1, r \rangle \in B$ となり，$y \in \mathbf{Z}_p^*$ の離散対数 $x \in \mathbf{Z}_{p-1}$ として "$x = r$" を出力する．一方，$q \neq 0$ であるならば，$h = g^m \pmod{p}$ を計算し，各 $1 \leq i \leq m$ に対して逐一 $h_i \equiv h^i \pmod{p}$ を求め (Giant-Step)，その都度 $\langle h_i, r \rangle \in B$ となるような $0 \leq r < m$ が存在するかどうかを検証する．式 (20) および式 (21) より，明らかに $i = q$ において Giant-Step は終了し，そのとき $y \in \mathbf{Z}_p^*$ の離散対数 $x \in \mathbf{Z}_{p-1}$ として "$x = qm + r$" を出力する．

上記の手順の実行時間は $O(2^n n \lg n \lg\lg n)$ であるが，集合 B に対するハッシュ表を用いることで，$\langle h_i, r \rangle \in B$ となるような $0 \leq r < m$ が存在するかどうかの検証を，定数回の比較で実行可能となる．これにより，Shanks 法の実行時間は $O(2^{n/2} n \lg n \lg\lg n)$ に低減される．

4.3 Pollard 法: ρアルゴリズム

Pollard 法 [13] は，列挙法・Shanks 法とは全くことなるアイディアに基づいた手法である．Pollard 法において，その実行時間は Shanks 法と概ね同様であるが，その記憶領域に関しては，Shanks 法と比較して遥かに優れるという特徴を有する．実際，Shanks 法と Pollard 法の実行時間と記憶領域を比較すると，以下のようになる．

	実行時間	記憶領域
Shanks 法	$O(2^{n/2} n \lg n \lg\lg n)$	$O(2^{n/2} \lg n)$
Pollard 法	$O(2^{n/2} n \lg n \lg\lg n)$	$O(1)$

† ただし，$n = \lg p$ とする．

まず $\mathbf{Z}_p^*$ に対し，$\mathbf{Z}_p^* = \mathbf{P}_1 \cup \mathbf{P}_2 \cup \mathbf{P}_3$ となるような $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ を定める．ただし，$\mathbf{P}_i \cap \mathbf{P}_j = \phi$ $(i \neq j)$ を満たすものとする．ここで，関数 $f : \mathbf{Z}_p^* \to \mathbf{Z}_p^*$ を以下のように定義する．

$$f(h) \equiv \begin{cases} gh \pmod{p} & h \in \mathbf{P}_1; \\ h^2 \pmod{p} & h \in \mathbf{P}_2; \\ yh \pmod{p} & h \in \mathbf{P}_3. \end{cases} \tag{22}$$

また，一様無作為に選ばれた $x_0 \in \mathbf{Z}_{p-1}$ に対し $h_0 \equiv g^{x_0} \pmod{p}$ を計算し，h_{i+1} を

$$h_{i+1} = f(h_i) \tag{23}$$

と定義する．この数列 $\{h_i\}_{i\geq 0}$ は，全ての $i \geq 0$ に対して，一般に以下のように表現される．

$$h_i \equiv g^{x_i} \cdot y^{z_i} \pmod{p}. \tag{24}$$

ここで，$z_0 = 0$ とすると，$\{x_i\}, \{z_i\}$ に関して，以下の件関係が成り立つ．

$$x_{i+1} \equiv \begin{cases} x_i + 1 \pmod{p-1} & h_i \in \mathbf{P}_1; \\ 2x_i \pmod{p-1} & h_i \in \mathbf{P}_2; \\ x_i \pmod{p-1} & h_i \in \mathbf{P}_3, \end{cases} \tag{25}$$

$$z_{i+1} \equiv \begin{cases} z_i \pmod{p-1} & h_i \in \mathbf{P}_1; \\ 2z_i \pmod{p-1} & h_i \in \mathbf{P}_2; \\ z_i + 1 \pmod{p-1} & h_i \in \mathbf{P}_3. \end{cases} \tag{26}$$

数列 $\{h_i\}_{i\geq 0}$ の値域は有限であるので，以下の関係が成り立つような $\ell > i \geq 0$ が存在する．

$$h_\ell \equiv h_i \pmod{p} \tag{27}$$

ここで，式 (23) より，“$g^{x_\ell} \cdot y^{z_\ell} \equiv g^{x_i} \cdot y^{z_i} \pmod{p}$” が成り立つので，

$$g^{x_\ell - x_i} \equiv y^{z_i - z_\ell} \pmod{p} \tag{28}$$

を得る．従って，式 (28) より，式 (1) の離散対数 x に関して，以下の線形合同式が成り立つ．

$$x_\ell - x_i \equiv x\,(z_i - z_\ell) \pmod{p-1}. \tag{29}$$

ここで，$z_i - z_\ell \in \mathbf{Z}^*_{p-1}$ であるならば，式 (1) の離散対数 x は，

$$x \equiv (x_\ell - x_i) \cdot (z_i - z_\ell)^{-1} \pmod{p-1} \tag{30}$$

により計算可能となる．一方，$z_i - z_\ell \notin \mathbf{Z}^*_{p-1}$ でないならば，改めて一様無作為に $x_0 \in \mathbf{Z}_{p-1}$ を選択し，式 (30) が成り立つまで同様な手順を繰り返す．

Pollard 法の実行時間と記憶領域の解析は繁雑であるので省略するが，本節の冒頭でも示したように，Shanks 法と比較して大幅な記憶領域の削減が実現されている．

4.4 Adleman 法

Adleman 法 [1] は，離散対数問題に対する (現在知られている) 最も効率的なアルゴリズムの基礎と位置付けることができる．以降簡単のため，関数 $L_n[a, b]$ を

$$L_n[a, b] = \mathsf{exp}\left\{a(\ln n)^b (\ln\ln n)^{1-b}\right\} \tag{31}$$

のように定義する．Adleman 法の基本的なアイディアは，比較的単純ではあるものの，その実行時間は $L_p[1/2, c + o(1)]$ で与えられる (ただし c は定数で $0 < c < 1$ を満たす)．

素数の集合を Prime とし，定数 $b > 0$ に対して，

$$\mathsf{Prime}[b] = \{q \in \mathsf{Prime} : q \leq b\} = \{q_1, q_2, \ldots, q_k\}$$

を定義する．Adleman 法の基本的なアイディアは，全ての $q_j \in \mathsf{Prime}[b]$ に対して，

$$q_j \equiv g^{e(q_j)} \pmod{p} \tag{32}$$

を満たす最小の非負整数 $e(q_j)$ を求めることに帰着される．実際には，以下の手順を繰り返し実行することにより，全ての $q_j \in \mathsf{Prime}[b]$ に対して，式 (32) を満たす $e(q_j)$ を求める．

(1) 一様無作為に $\alpha_i \in \mathbf{Z}_{p-1}$ を選択する．

(2) $\beta_i \equiv g^{\alpha_i} \pmod{p}$ を計算する．

(2-1) β_i が b-平坦でないならば (1) へ．

(2-2) β_i が b-平坦ならば，β_i の素因数分解を計算し，$c(q_1, \alpha_i), c(q_2, \alpha_i), \ldots, c(q_k, \alpha_i)$ を出力．

$$\beta_i = \prod_{q_j \in \mathsf{Prime}[b]} q_j^{c(j,\alpha_i)}.$$

ここで，α_i と $c(q_1, \alpha_i), c(q_2, \alpha_i), \ldots, c(q_k, \alpha_i)$ に関して，以下が成り立つことに注意する．

$$\alpha_i \equiv c(q_1, \alpha_i)e(q_1) + c(q_2, \alpha_i)e(q_2) + \cdots + c(q_k, \alpha_i)e(q_k) \pmod{p-1}. \tag{33}$$

従って，式 (33) において，$\alpha_1, \alpha_2, \ldots, \alpha_k$ に関して，k 個の線形独立な式が得られるまで，上記の手順を繰り返すことで，以下の連立線形合同式を得る．

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \vdots \\ \alpha_i \\ \vdots \\ \alpha_k \end{bmatrix} = \begin{bmatrix} c(q_1,\alpha_1) & c(q_2,\alpha_1) & \cdots & c(q_k,\alpha_1) \\ c(q_1,\alpha_2) & c(q_2,\alpha_2) & \cdots & c(q_k,\alpha_2) \\ \vdots & \vdots & \vdots & \vdots \\ c(q_1,\alpha_i) & c(q_2,\alpha_i) & \cdots & c(q_k,\alpha_i) \\ \vdots & \vdots & \vdots & \vdots \\ c(q_1,\alpha_k) & c(q_2,\alpha_k) & \cdots & c(q_k,\alpha_k) \end{bmatrix} \begin{bmatrix} e(q_1) \\ e(q_2) \\ \vdots \\ e(q_i) \\ \vdots \\ e(q_k) \end{bmatrix} \pmod{p-1}. \tag{34}$$

式 (34) より，全ての $q_j \in \mathsf{Prime}[b]$ に対して，式 (32) を満たす $e(q_j)$ を求めることができる．

ここで，任意の $y \in \mathbf{Z}_p^*$ に対して，式 (1) を満たす $x \in \mathbf{Z}_{p-1}$ は，以下のように求められる．

(1) 一様無作為に $r \in \mathbf{Z}_{p-1}$ を選択する．

(2) $z \equiv y \cdot g^r \pmod{p}$ を計算する．

(2-1) z が b-平坦でないならば (1) へ．

(2-2) z が b-平坦ならば，z の素因数分解を計算し，$c(q_1, z), c(q_2, z), \ldots, c(q_k, z)$ を出力．

$$z = \prod_{q_j \in \mathsf{Prime}[b]} q_j^{c(j,z)}.$$

このとき，$x, r \in \mathbf{Z}_{p-1}$ と $c(q_1, z), c(q_2, z), \ldots, c(q_k, z)$ に関して，以下が成り立つ．

$$x + r \equiv c(q_1, z)e(q_1) + c(q_2, z)e(q_2) + \cdots + c(q_k, z)e(q_k) \pmod{p-1}. \tag{35}$$

式 (34) より，式 (32) を満たす $e(q_j)$ は既知であるので，式 (35) より，以下を得る．

$$x \equiv c(q_1, z)e(q_1) + c(q_2, z)e(q_2) + \cdots + c(q_k, z)e(q_k) - r \pmod{p-1}.$$

4.5 数体ふるい法

離散対数に対する数体ふるい法 [8] は，現在最も高速なアルゴリズムとして知られており，その実行時間は $L_p[1/3, (64/9)^{1/3}]$ である．離散対数に対する数体ふるい法は，素因数分解アルゴリズムとしての数体ふるい法 [11, 3] に基づいているが，その基本的なアイディアは Adleman 法の延長線上にあるので，ここではそのアルゴリズムの記述と解析は省略する．

4.6 公開鍵暗号方式の安全性に対する影響

4.6.1 列挙法

第 4.1 節で述べたように，列挙法は，素数 p に対して制限のない一般の場合に対しても適用可能であるが，その計算時間は入力長に対して指数時間となるため，十分大きな入力に対しては，事実上実行不可能となる．従って，有限体の乗法群における離散対数問題に基づいて暗号プロトコルを設計する際に，列挙法による攻撃を想定するならば，列挙法の計算時間が事実上実行不可能となるように鍵サイズを設定する必要がある．

4.6.2 Shanks 法: Baby-Step/Giant-Step

第 4.2 節で述べたように，Shanks 法は，素数 p に対して制限のない一般の場合に対しても適用可能であり，列挙法に比べてその計算時間は少ないものの，入力長に対して指数時間となるため，十分大きな入力に対しては，事実上実行不可能となる．従って，有限体の乗法群における離散対数問題に基づいて暗号プロトコルを設計する際に，Shanks 法による攻撃を想定するならば，Shanks 法の計算時間が事実上実行不可能となるように鍵サイズを設定する必要がある．

4.6.3 Pollard 法: ρアルゴリズム

第 4.3 節で述べたように，Pollard 法は，素数 p に対して制限のない一般の場合に対しても適用可能であり，列挙法に比べてその計算時間は少ないものの，入力長に対して指数時間となるため，十分大きな入力に対しては，事実上実行不可能となる．従って，有限体の乗法群における離散対数問題に基づいて暗号プロトコルを設計する際に，Pollard 法による攻撃を想定するならば，Pollard 法の計算時間が事実上実行不可能となるように鍵サイズを設定する必要がある．

4.6.4 Adleman 法

第 4.4 節で述べたように，Adleman 法は，素数 p に対して制限のない一般の場合に対しても適用可能であり，列挙法・Shanks 法・Pollard 法に比べてその計算時間は少ないものの，入力長に対して準指数時間となる．従って，列挙法・Shanks 法・Pollard 法同様，十分大きな入力に対しては，Adleman 法は事実上実行不可能となる．よって，有限体の乗法群における離散対数問題に基づいて暗号プロトコルを設計する際に，Adleman 法による攻撃を想定するならば，Adleman 法の計算時間が事実上実行不可能となるように鍵サイズを設定する必要がある．

4.6.5 数体ふるい法

第4.5節で述べたように，数体ふるい法は，素数pに対して制限のない一般の場合に対しても適用可能であり，Adleman法に比べてその計算時間は少ないものの，入力長に対して準指数時間となる．従って，Adleman法同様，十分大きな入力に対しては，数体ふるい法は事実上実行不可能となる．よって，有限体の乗法群における離散対数問題に基づいて暗号プロトコルを設計する際に，数体ふるい法による攻撃を想定するならば，数体ふるい法法の計算時間が事実上実行不可能となるように鍵サイズを設定する必要がある．

5 結論

本研究調査では，有限体の乗法群に関する離散対数問題に対して，既存の解読アルゴリズムを概観し，その効果について検討した．本研究調査で示したアルゴリズムの多くは，一般の(巡回群ではない)有限群における離散対数問題にも適用可能であるので，従って，一般の指数計算アルゴリズムとしても利用可能であることに注意する．

現在に至るまでの計算機ハードウェアの進歩を考慮すると，アルゴリズムの進歩が全く無い場合を想定しても，安全性が保たれる鍵サイズの増加 — 高々数10ビット程度 — が見込まれるが，これは近い内に飽和するものと予想される．また，離散対数問題に体するアルゴリズムの改良により，より高速な解読法が発見される可能性も否定できないが，現在の計算モデルを想定する限り，極めて特殊な場合を除いて，離散対数問題(および一般の指数計算)に対する多項式時間アルゴリズムが開発される可能性は極めて低いものと予想する．

一方，これまでとは全く異なる計算モデルが実現された場合，離散対数問題(および一般の指数計算)に対する多項式時間アルゴリズムが開発される可能性が考えられる．実際，量子計算機[4, 5]上において，離散対数問題に対する多項式時間アルゴリズム[18]が知られている．しかし現在までのところ，量子計算機がハードウェアとして実現される可能性は(極めて)低いため，離散対数問題に基づく暗号プロトコルの安全性を考える場合，量子計算機のハードウェア実現に対する画期的なブレイクスルーが発見されない限り，量子計算機上での解読アルゴリズムを想定する必要性は極めて低いと判断される．

以上より，現状および近未来の技術水準から総合的に判断すると，有限体の乗法群に関する離散対数問題は，暗号プロトコルの基盤要素として十分な安全性を有するものと考えられる．

参考文献

[1] Adleman, L.M., "A Subexponential Algorithm for the Discrete Logarithm Problem with Applications to Cryptography," *Proc. of the 20th IEEE Annual Symposium on Foundations of Computer Science*, pp. 55-60 (1979).

[2] Bellare, M. and Micali, S., "Noninteractive Oblivious Transfer and Applications," *Advances in Cryptology — Crypto'89*, Lecture Notes in Computer Science 435, Springer-Verlag, pp.547-557 (1990).

[3] Coppersmith, D., "Modifications to the Number Field Sieve," *J. Cryptology*, Vol.6, pp.169-180 (1993).

[4] Deutsch, D., "Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer," *Proc. of R. Soc. Lond.*. Vol.A, No.400, pp.97-117 (1985).

[5] Deutsch, D., "Quantum Computational Networks," *Proc. of R. Soc. Lond.*. Vol.A, No.425, pp.73-90 (1989).

[6] Diffie, W. and Hellman, M.E., "New Directions in Cryptography," *IEEE Trans. on Inform. Theory*, Vol.IT-22, No.6, pp.644-654 (1976).

[7] El-Gamal, T., "A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," *IEEE Trans. on Inform. Theory*, Vol.IT-31, No.4, pp.469-472 (1985).

[8] Gordon, D., "Discrete Logarithms in $\mathbf{GF}(p)$ Using the Number Field Sieve," *SIAM J. Discrete Math.*, Vol. 6, pp.124-138 (1993).

[9] Kurosawa, K., Itoh, T., and Takeuchi, M., "Public-Key Cryptosystem Using a Reciprocal Number with the Same Intractability as Factoring a Large Number, *Cryptologia*, Vol.XII, No.4, pp.225-233 (1988).

[10] Okamoto, T., "Encryption and Authentication Schemes Based on Public-Key Systems," Ph.D. Thesis, The University of Tokyo (1988).

[11] Lenstra, A.K., Lenstra, H.W., Manasse, M.S., Pollard, J.M., "The Number Field Sieve," *Proc. of the 22nd ACM Annual Symposium of Theory of Computing*, pp.564-572 (1999).

[12] Okamoto, T. and Uchiyama, S., "A New Public-Key Cryptosystem as Secure as Factoring," *Advances in Cryptology — Eurocrypt'98*, Lecture Notes in Computer Science 1403, Springer-Verlag, pp.308-318 (1998).

[13] Pollard, J.M., "Monte Carlo Methods for Index Computation mod p," *Math. Comp.*, Vol.32, pp.918-924 (1978).

[14] Pohlig, S.C. and Hellman, M.E., "An Improved Algorithm for Computing Logarithms over $GF(p)$ and its Cryptographic Significance," *IEEE Trans. Inform. Theory*, Vol.IT-24, No.1, pp.106-110 (1978).

[15] Rabin, M.O., "Digitalized Signatures and Public-Key Functions as Intractable as Factorization," MIT/LCS/TR-212, MIT Laboratory for Computer Science (1979).

[16] Rivest, R.L., Shamir, A., and Adleman, L.M., "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Comm. of the ACM*, Vol.21, No.2, pp.120-126 (1978).

[17] Semaev, I.A., "Evaluation of Discrete Logarithms in a Group of p Torsion Points of an Elliptic Curve in Characteristic p," *Math. Comp.*, Vol.67, pp.353-356 (1998).

[18] Shor, P.W., "Algorithms for Quantum Computation: Discrete Log and Factoring," *Proc. of the 35th Annual IEEE Symposium on Foundations of Computer Science*, pp.20-32 (1994).

[19] Satoh, T. and Araki, K., "Fermat Quotients and the Polynomial Time Discrete Log Algorithm for Anomalous Elliptic Curves," *Commentarii Math*, Univ. St. Pauli, Vol.47, pp.81-92 (1998).

[20] Rivest. R.L., "Cryptography," *Handbook of Theoretical Computer Science*, Vol.A, Cambridge, pp.717-755 (1990).

[21] Shamir, A., Rivest, R.L., and Adleman, L.M., "Mental Poker," MIT/LCS/TM-125 (1979).

[22] Shanks, D., "Class Number, a Theory of Factorization, and Genera," *Proc. of Symposia in Pure Mathematics*, Vol.20, pp. 415-440 (1969).

[23] Smart, N.P., "The Discrete Logarithm Problem on Elliptic Curves of Trace One," *J. Cryptology*, Vol.12, pp.193-196 (1999).

[24] Williams, H.C., "A Modification of the RSA Public-Key Encryption Procedure," *IEEE Trans. on Inform. Theory*, Vol.IT-26, No.6, pp.726-729 (1980).

[25] Williams, H.C., "Some Public-Key Crypto-Functions as Intractable as Factorization," *Cryptologia*, Vol.9, No.3, pp.223-237 (1985).