Vol.2019-DPS-178 No.26
Vol.2019-CSEC-84 No.26
2019/3/5

LRPC符号ベース暗号に対する代数攻撃について

前澤 陽平[1,a)]　Tung CHOU[1,b)]　宮地 充子[1,c)]

概要：2013 年提案された LRPC 符号ベース暗号は，ランク距離基準の誤り訂正符号である LRPC 符号を用いる．その安全性は，LRPC 符号のランクシンドローム復号 (RSD) 問題及びランク符号語発見 (RCF) 問題の計算困難性に依存する．2006 年に提案された Levy-Perret 攻撃は，RSD 問題と本質的に等価なランク距離復号 (RD) 問題に対する代数攻撃である．Levy-Perret 攻撃は拡大した生成行列由来の等式とパリティ検査行列由来の等式とを合わせた多変数多項式系を生成する．本研究では，Levy-Perret 攻撃におけるパリティ検査行列由来の等式を改善を提案するとともに，変数の削減方法を提案する．また，LRPC 符号ベース暗号に対する Levy-Perret 攻撃について，いくつかの条件の下，検証を行った．検証結果の一つとして，パリティ検査行列由来の等式は攻撃への貢献が少ないことが示された．これにより，より効率的な多変数多項式系を形成できる可能性が示唆された．

キーワード：耐量子コンピュータ暗号，符号ベース暗号，LRPC 符号，代数攻撃

An algebraic attack against LRPC code-based cryptography

1. はじめに

現在用いられている公開鍵暗号の安全性の多くは，素因数分解や離散対数問題等の数学的問題の計算困難性に依存している．これらの問題は，将来，大規模な量子コンピュータによって効率的に解かれてしまう．しかし，他の問題の中には，量子コンピュータの解読に耐え得るものが存在する可能性がある．量子コンピュータの解読に耐え得る問題を安全性の根拠とする暗号を，耐量子コンピュータ暗号と呼ぶ．符号ベース暗号は安全性として，使用する符号の復号問題を計算困難性の根拠としており，耐量子コンピュータ暗号の一つとされている．

符号ベース暗号は，McEliece 暗号方式 [1] や Niederreiter 暗号方式 [2] をフレームワークとして，誤り訂正符号とともに用いる．[1] で提案された McEliece 暗号は誤り訂正符号として Goppa 符号を用いる．Goppa 符号を用いた McEliece 暗号は現在でも安全とされているが，公開鍵のサイズがとても大きいという欠点を持つ．

小さい公開鍵長の実現を目的とした LRPC(低ランクパリティ検査) 符号 [3] が提案された．LRPC 符号はランク距離基準の誤り訂正符号であり，MDPC 符号等のハミング距離基準の誤り訂正符号とは異なる構造を持つ．ランク距離基準の誤り訂正符号は，もともと Gabidulin 符号 [4] として 1985 年に提案された．1991 年に Gabidulin 符号を用いた McEliece 暗号 [5] が提案されたが，Gabidulin 符号の構造を利用した攻撃 [6, 7] により既に破壊されいる．2013 年に提案された LRPC 符号は，Gabidulin 符号と同じランク距離基準の誤り訂正符号であるが，Gabidulin 符号のような特徴的な構造を持っていない．QC-LRPC 符号を用いた McEliece 暗号は，比較的小さい公開鍵のサイズ (80 ビットセキュリティで 1500 ビット強) を達成した．

LRPC 符号の計算困難性はランクシンドローム復号 (RSD) 問題とランク符号語発見 (RCF) 問題に依存するが，これらの問題に対する攻撃は元々提案されている．その代表例に代数攻撃がある．代数攻撃 [8–10] は，ランク距離基準の誤り訂正符号の公開情報から導出した多変数多項式系を解くことを通して，RSD 問題や RCF 問題を直接的に解く手法である．LRPC 符号ベース暗号として提案されている暗号方式は，提案論文内で代数攻撃の影響を論じている．しかし，計算量を引用するに留まる等，文章的

[1] 大阪大学大学院工学研究科
Graduate School of Engineering, Osaka University,
Suita, Osaka 565–0871 Japan
[a)] maezawa@cy2sec.comm.eng.osaka-u.ac.jp
[b)] blueprint@crypto.tw
[c)] miyaji@comm.eng.osaka-u.ac.jp

な説明に終始している．また，最も新しい代数攻撃である Levy-Perret 攻撃 [8] は 2006 年に提案されたため，その実験結果は LRPC 符号ベース暗号の暗号方式に即している訳ではない．

Levy-Perret 攻撃は生成行列由来の等式とパリティ検査行列由来の等式とを併せて多変数多項式系とし，その多変数多項式系を Gröbner 基底アルゴリズムにより解いている．しかし，パリティ検査行列由来の等式は [8] の記述のままでは実装することができないため，パリティ検査行列由来の等式を実装可能な形を提案した．また，Levy-Perret 攻撃では，誤りベクトルをその基底の行列 X と基底の組み合わせの行列 A とに分けて扱っているが，その行列 A について変数の数を削減する手法を提案した．ただし，この手法には攻撃失敗の可能性が伴う，つまり攻撃時間の短縮と攻撃失敗の可能性のトレードオフが存在する．続いて，いくつかの条件の下，小さいパラメータの LRPC 符号ベース暗号に対して Levy-Perret 攻撃の検証を行った．検証結果のうち特に注目すべきものは，Levy-Perret 攻撃の多変数多項式系のうち，パリティ検査行列由来の等式は攻撃への貢献度が低いことであった．

第 2 節では，暗号に用いる符号である LRPC 符号について述べ，その LRPC 符号を使用した LRPC McEliece 暗号について説明する．そして，その復号時に用いるビットフリッピングアルゴリズムについて説明する．第 3 節では，LRPC 符号に対する攻撃として代数攻撃について記述する．第 4 節において，代数攻撃を実装する際の提案について記す．第 5 節では，代数攻撃について検証した結果について記述するとともに，それに対する考察を記す．

2. LRPC 符号ベース暗号

2.1 LRPC 符号

体 $\mathbb{F}$，$n \in \mathbb{N}$ について，長さ n のベクトル $v = (v_0, v_1, ..., v_{n-1})\,(v_i \in \mathbb{F})$ で表される空間全体をベクトル空間 V とする．ベクトル空間の基底は，一次独立なベクトルの極大集合のことを表し，V の任意のベクトルは基底の要素ベクトルの一次結合で表すことができる．ここで，その基底に含まれる元の個数を，ベクトル空間の次元という．

定義 1 (LRPC 符号) 上記のベクトル空間 V の k 次元部分空間を，(n, k) 線形符号 C という．

定義 2 (生成行列) 各行が C の基底をなす $k \times n$ 行列を，$G \in \mathbb{F}^{k \times n}$ とする．G を C の生成行列という．

続いて，次式を満たす長さ n のベクトル h を考える．

$$Gh^T = 0$$

h^T は h の転置を，0 は長さ k の零ベクトルを表す．上式を満たすベクトル h の全体は，ベクトル空間 V の $n-k$ 次元の部分空間である．このような h^T を G の核という．

定義 3 (パリティ検査行列) 各行の転置が G の核かつ一次独立である $(n-k) \times n$ 行列を，$H \in \mathbb{F}^{(n-k) \times n}$ とする．H を C のパリティ検査行列という．

符号語 $c \in C$ と誤りベクトル $e \in \mathbb{F}^n$ の和 $c+e$ の**シンドローム**は次式のように表すことができる．

$$s = (c+e)H^T = eH^T$$

以上のように，シンドロームは符号語に乗せられた誤りベクトルの影響を表していることが分かる．線形符号はシンドロームを用いることで，誤りベクトルを取り除き（訂正し），符号語を導きだすことができる．

線形符号が訂正できる最大の誤りの個数を，誤り訂正能力という．LRPC 符号は，一般的な符号のようなハミング距離基準ではなく，ランク距離基準の誤り訂正符号である．そのため，LRPC 符号の誤り訂正能力はランク重みという指標に関係している．

ベクトル $x = (x_1, x_2, ..., x_n) \in \mathbb{F}_{q^m}^n$ について，$(\beta_1, \beta_2, ..., \beta_m) \in \mathbb{F}_q^m$ は，$\mathbb{F}_q$ 上の m 次元ベクトルと見なすことで，$\mathbb{F}_{q^m}$ 上の基底を成す．ベクトル x の各要素 x_j は，この基底を用いることで $\mathbb{F}_q^m$ のベクトルと見なすことができ，$x_j = \sum_{i=1}^{m} z_{ij}\beta_i,\ 1 \le j \le n$ と表すことができる．ベクトル x に関連する $m \times n$ 行列は，$M(x) = (z_{ij})_{\substack{1 \le i \le m \\ 1 \le j \le n}}$ と表記できる．

定義 4 (ランク重み) ベクトル $x \in \mathbb{F}_{q^m}^n$ のランク重み $||x||$ もしくは $\mathrm{Rk}(x \,|\, \mathbb{F}_q)$ は，次のように定義する．

$$||x|| \stackrel{\mathrm{def}}{=} \mathrm{Rank}\,(M(x)\,)$$

ベクトル x のランク重みを，単に x のランクを表記することもある．$\mathbb{F}_{q^m}^n$ 上のベクトル x 及び y 間のランク距離 $d_r(x, y)$ は，ランク重みを用いて $d_r(x, y) = ||x - y|| = ||x|| - ||y||$ と定義される．このように，ランク距離基準の誤り訂正符号では，ハミング重みではなくランク重みを用いる．

定義 5 ($[n, k]_{q^m}$ 線形符号) ランク距離基準を用いた $\mathbb{F}_{q^m}$ 上の (n, k) 線形符号である．つまり，$\mathbb{F}_{q^m}$ 上の長さ n，次元 k の $[n, k]_{q^m}$ 線形符号 C_r は，ランク距離基準の $\mathbb{F}_{q^m}^n$ の k 次元部分空間である．

(n, k) 線形符号同様，$[n, k]_{q^m}$ 線形符号 C_r は，生成行列 $G \in \mathbb{F}_{q^m}^{k \times n}$ を用いた方法とパリティ検査行列 $H \in \mathbb{F}_{q^m}^{(n-k) \times n}$ を用いた方法の二通りで表現できる．

定義 6 (ランク距離の最小距離) ランク距離基準の符号の最小距離は，異なる 2 つの符号語のランク距離の最小値で

ある．

$[n,k]_{q^m}$ 線形符号の最小距離は，非零の符号語の最小ランク重みに等しい．最小距離を d とした際の誤り訂正能力 t は，$t = \lfloor \frac{d-1}{2} \rfloor$ となる．

定義 7 (support) $x = (x_1, x_2, ..., x_n) \in \mathbb{F}_{q^m}^n$ は，ランク重み r のベクトルとする．各 $x_1, x_2, ..., x_n \in \mathbb{F}_{q^m}$ で生成される $\mathbb{F}_{q^m}$ 上の $\mathbb{F}_q$ 部分ベクトル空間を，x の support という．x の support を X とすると，X は次のように表記する．

$$X = \langle x_1, x_2, ..., x_n \rangle_{\mathbb{F}_q}$$

上記の定義について X の次元が r であることから，X の $\mathbb{F}_{q^m}$ 上の基底を $\{X_1, ..., X_r\}$ と表記できる．また，この support の積集合について以下のように定義する．

定義 8 (support の積集合) A と B をそれぞれ次元 α と β の $\mathbb{F}_{q^m}$ 上の $\mathbb{F}_q$ 部分ベクトル空間とする．A と B の基底はそれぞれ $\{A_1, ..., A_\alpha\}, \forall A_i \in \mathbb{F}_{q^m}$ 及び $\{B_1, ..., B_\beta\}, \forall B_i \in \mathbb{F}_{q^m}$ と表記できる．このとき，A と B の積集合 $\langle AB \rangle$ は，集合 $\{ab \mid a \in A, b \in B\}$ で生成される．

上記の定義について，積集合 $\langle AB \rangle$ は，$\{A_1B_1, ..., A_1B_\beta, ..., A_\alpha B_1, ..., A_\alpha B_\beta\}$ で生成される．$\langle AB \rangle$ の次元の上限は，$\alpha\beta$ である．

以上を踏まえた上で，LRPC 符号の定義を記す．

定義 9 (LRPC 符号) $\mathbb{F}_{q^m}$ 上のランク d，長さ n，次元 k の LRPC 符号は，パリティ検査行列 $H \in \mathbb{F}_{q^m}^{(n-k)\times n}$ が次の特徴を持つ $[n,k]_{q^m}$ 線形符号である．H の各座標 $(h_{ij})_{\substack{1 \le i \le n-k \\ 1 \le j \le n}}$ の support を F とする．

$$F = \langle (h_{ij})_{\substack{1 \le i \le n-k \\ 1 \le j \le n}} \rangle_{\mathbb{F}_q}$$

F の次元は，d である．このとき，d を H のランク重みという．F の $\mathbb{F}_{q^m}$ 上の基底を $\{F_1, ..., F_d\}$ とする．

2.2 LDPC McEliece 暗号

LRPC McEliece 暗号 [3] とは，誤り訂正符号として LRPC 符号を用いた McEliece 暗号方式のことである．LRPC McEliece 暗号の秘密鍵は二つある．一つはランク d のパリティ検査行列 $H \in \mathbb{F}_{q^m}^{(n-k)\times n}$ であり，もう一つはランダムな可逆行列 $R \in \mathbb{F}_{q^m}^{(n-k)\times(n-k)}$ である．公開鍵は，生成行列 $G \in \mathbb{F}_{q^m}^{n\times k}$ に R を左から掛けた $G' = RG$ である．平文を $m \in \mathbb{F}_2^{n-r}$，暗号文を $x \in \mathbb{F}_2^n$ としたとき，暗号化は，$||e|| \le r$ となるベクトル $e \in \mathbb{F}_{q^m}^n$ を用いた，$x \leftarrow mG' + e$ である．復号は，Algorithm 1 を使用する．

2.3 デコードアルゴリズム

LRPC 符号ベース暗号に使用するデコードアルゴリズム [3] を Algorithm 1 に記す．

Algorithm 1 は，主に四つの過程から成る．先ず，シンドローム空間 S の計算である．シンドローム空間とは，シンドロームの各要素の作る $\mathbb{F}_{q^m}$ 上の $\mathbb{F}_q$ 部分ベクトル空間である．シンドローム空間 S は，H の support F と誤りベクトル e の support E の積集合である．次に，シンドローム空間 S から，support F を用いて，support E を算出する．続いて，support E を用いて，シンドロームの計算式を線形方程式系とみなし解く．このとき，この線形方程式系は，$\mathbb{F}_q$ 上の nr 個の変数を含む，$(n-k)rd$ 個の等式から成る．E の次元が r であるため，e は E の r 個の基底の線形結合で表すことができる．よって，e の長さ n より，この系は $\mathbb{F}_q$ 上の nr 個の変数を持つ．また，シンドローム s についても S の rd 個の線形結合で表せる．シンドロームの計算式 $He^t = s^T$ について，s の長さ $n-k$ より，この系は $(n-k)rd$ 個の等式を持つ．最後に，e を用いて受信語から誤りを取り除き，符号語を手に入れる．

2.4 計算困難性

一般的な $[n,k]_{q^m}$ 線形符号の復号の計算困難性は，次の二つの問題に依存する．

定義 10 (ランクシンドローム復号 (RSD) 問題)
Given: $H \in \mathbb{F}_{q^m}^{(n-k)\times n}$, $s \in \mathbb{F}_{q^m}^{n-k}$, an integer $r > 0$.
Find: $e \in \mathbb{F}_{q^m}^n$ such that $\mathrm{Rk}(e \,|\, \mathbb{F}_q) \le r$ and $He^T = s^T$.

定義 11 (ランク符号語発見 (RCF) 問題)
Given: $H \in \mathbb{F}_{q^m}^{(n-k)\times n}$, an integer $r > 0$.
Find: $c \in \mathbb{F}_{q^m}^n$ such that $\mathrm{Rk}(c \,|\, \mathbb{F}_q) \le r$ and $Hc^T = 0^T$.

RSD 問題で与えられる H は，$[n,k]_{q^m}$ 線形符号のパリ

Algorithm 1 Decoding algorithm for LRPC codes

Input: parity check matrix $H \in \mathbb{F}_{q^m}^{(n-k)\times n}$, H's space F, F's dimension d, received word $y \in \mathbb{F}_{q^m}^n (= c + e, c \in C_r)$, error vector's rank r
Output: c

1: Compute $s^T \leftarrow Hy^T$ /* $s = (s_1, ..., s_{n-k})$ */
2: Compute $S = \langle s_1, ..., s_{n-k} \rangle_{\mathbb{F}_q}$
3: **for** $i = 1$ to d **do**
4: Define $S_i = F_i^{-1} S$
5: **end for**
6: Compute $E = S_1 \cap S_2 \cap \cdots \cap S_d$
7: Compute a basis $\{E_1, E_2, .., E_r\}$
8: Solve the system $He^T = s^T$
9: Return $c \leftarrow y - e$

Vol.2019-DPS-178 No.26
Vol.2019-CSEC-84 No.26
2019/3/5

ティ検査行列である．RSD 問題はランク r の誤りベクトルを訂正することに，RCF 問題は重み r の符号語が存在することにそれぞれ関連する．RSD 問題は，確率的縮約により NP 困難であることが証明されている [11]．RSD 問題はパリティ検査行列の側面から問題を定義しているが，生成行列の側面から定義した問題も存在し，RD 問題という．

定義 12 (ランク距離復号 (RD) 問題)
Given: $G \in \mathbb{F}_{q^m}^{k\times n}$, $c \in \mathbb{F}_{q^m}^{n}$, an integer $r > 0$.
Find: $m \in \mathbb{F}_{q^m}^{k}$ such that $e = c - mG$ and $\mathrm{Rk}(e\,|\,\mathbb{F}_q) \leq r$.

RD 問題で与えられる G は，$[n,k]_{q^m}$ 線形符号の生成行列である．RSD 問題と RD 問題とは，定義に用いる行列は異なるものの本質的に等価である．

3. 代数攻撃

LRPC 符号ベース暗号に対する代数攻撃について述べる．LRPC 符号ベース暗号に対する攻撃は大きく分けて二つ存在する．一つは直接攻撃といい，ランク r の誤りベクトル e を見つけることで平文を直接復元することを目的とする．もう一つは構造攻撃といい，公開鍵の構造を用いて秘密鍵を復元することを目的とする．本稿では，直接攻撃のうち，代数攻撃に焦点を当てる．代数攻撃は，LRPC 符号ベース暗号の公開情報から生成した多変数多項式系を解くことを通して，LRPC 符号ベースの依存する各問題を解くことを目的としている．代数攻撃は，主に 3 つの手法が存在し，直接的手法，縮約的手法及び多項式的手法という．ここでは，本研究に関係する直接的手法として，Ourivski-Johansson 攻撃 [9] 及び Levy-Perret 攻撃 [8] について述べる．

3.1 Ourivski-Johansson 攻撃

2002 年に提案された Ourivski-Johansson 攻撃 [9] は，RD 問題を解く手法である．

先ず，[9] における RD 問題のモデル化について記す．符号 C_r に誤りベクトル e による部分ベクトル空間を加えた符号を C_e とする．C_e の生成行列を G_e とすると，次のように表すことができる．

$$G_e = \begin{pmatrix} G \\ mG + e \end{pmatrix} = \begin{pmatrix} I_k & 0 \\ m & 1 \end{pmatrix} \begin{pmatrix} G \\ e \end{pmatrix}$$

式中 I_k は $k \times k$ の単位行列を表す．C_r の最小ランク距離は d であり，誤り訂正可能数は $t = \lfloor (d-1)/2 \rfloor$ である．$\mathrm{Rk}(e\,|\,\mathbb{F}_q) = r$ のとき，$r \leq t$ となる．$r \leq d$ であるため，符号 $\mathcal{C}_e$ のランク r の符号語は，$\epsilon \in \mathbb{F}_{q^m}^{*}$ を用いて $e = \epsilon\bar{e}$ と表すことができる．次に，誤りベクトル $\bar{e}$ の変形について記す．

G_e を組織符号形式で表現すると次のようになる．

$$G_{\mathrm{syst}} = \begin{pmatrix} I_{k+1} & R \end{pmatrix}$$

ここで $R \in \mathbb{F}_{q^m}^{(k+1)\times(n-k-1)}$ である．この G_{syst} に対して，

$$uG_{\mathrm{syst}} = \begin{pmatrix} u & uR \end{pmatrix} = \epsilon\bar{e}$$

を満たすような長さ $k+1$ のベクトル u を掛ける． 誤りベクトル $\bar{e}$ の最初の $k+1$ 個の要素を $\bar{e}_1$，後ろの $n-k-1$ 個の要素を $\bar{e}_2$ とすると，$\bar{e}$ は次のように表せる．

$$\bar{e} = \begin{pmatrix} \bar{e}_1 & \bar{e}_2 \end{pmatrix}$$

この $\bar{e}$ について，$u = \epsilon\bar{e}_1$ 及び $\bar{e}_2 = \bar{e}_1 R$ を用いることで，次式を入手できる．

$$\bar{e} = \begin{pmatrix} \bar{e}_1 & \bar{e}_1 R \end{pmatrix} \tag{1}$$

$\mathrm{Rk}(e\,|\,\mathbb{F}_q) = r$ より，誤りベクトル $\bar{e}$ は，r 個の基底を用いて表現することができる．$\bar{e}$ の $\mathbb{F}_q$ 上の $\mathbb{F}_{q^m}$ の基底を $X = (x_0, x_1, ..., x_{r-1})$ とすると，$\bar{e}$ は次のように表せる．

$$\begin{aligned} \bar{e} &= (x_0, x_1, ..., x_{r-1}) \begin{pmatrix} \alpha_{0,1} & \cdots & \alpha_{0,n} \\ \alpha_{1,1} & \cdots & \alpha_{1,n} \\ \vdots & \ddots & \vdots \\ \alpha_{r-1,1} & \cdots & \alpha_{r-1,n} \end{pmatrix} \\ &= XA \end{aligned} \tag{2}$$

行列 A は，誤りベクトルの各要素が取りうる基底の組み合わせを表しており，$A = (\alpha_{i,j})_{\substack{1\leq i\leq r \\ 1\leq j\leq n}} \in \mathbb{F}_q^{r\times n}$ である．上記の式 (1) 及び (2) より，$\bar{e} = (\bar{e}_1, \bar{e}_1 R) = XA$ となる．行列 A を最初の $k+1$ 列目までと残りの $n-k-1$ 列分とで分割すると，$A = (A_1, A_2)$ と表すことができ，$A_1 \in \mathbb{F}_q^{r\times(k+1)}$，$A_2 \in \mathbb{F}_q^{r\times(n-k-1)}$ である．これにより，$\bar{e} = (\bar{e}_1, \bar{e}_1 R) = X(A_1, A_2)$ となるので，$\bar{e}_1 = XA_1$ 及び $\bar{e}_1 R = XA_2$ を得る．以上より，次式を入手できる．

$$(x_0, x_1, ..., x_{r-1})A_1 R = (x_0, x_1, ..., x_{r-1})A_2 \tag{3}$$

ここで，どんな $\epsilon \in \mathbb{F}_{q^m}^{*}$ でも，$e = \epsilon\bar{e}$ を満たす必要があるため，$\bar{e}$ の基底には必ず単位元 1 を含んでいる．単位元 1 を考慮すると，式 (3) は次のように書くことができる．

$$(1, x_1, ..., x_{r-1})A_1 R = (1, x_1, ..., x_{r-1})A_2 \tag{4}$$

式 (4) は，$\mathbb{F}_{q^m}$ 上の $n-k-1$ 個の等式及び $nr+r-1$ 個の未知の変数を持つ．この式 (4) を列単位でみると，次の式と同等であると言える．

$$(1, x_1, ..., x_{r-1})A_1 R_j = (1, x_1, ..., x_{r-1})(A_2)_j \tag{5}$$

なお，$k+2 \leq j \leq r-1$ である．R_j 及び $(A_2)_j$ は，それぞれの行列の j 個目の列を表している．ここで，$\mathbb{F}_q$ 上の $\mathbb{F}_{q^m}$ の基底を新たに考える．$\mathbb{F}_q$ 上の $\mathbb{F}_{q^m}$ の基底を

$\Omega = (\omega_0, ..., \omega_{m-1})$ とすると，基底 x_i は次式で表せる．

$$x_i = \sum_{j=0}^{m-1} x_{ij}\omega_j, \quad (x_{ij} \in \mathbb{F}_q, \quad 1 \leq i \leq r-1)$$

これを用いることで，式 (5) は，$\mathbb{F}_q$ 上の $m(n-k-1)$ 個の等式及び $nr + m(r-1)$ 個の未知の変数を持つ．

続いて，[9] の提案手法について述べる．$|\mathcal{J}| = z$ となる $\mathcal{J} \subseteq [k+2, .., n]$ を選択する．式 (5) の R 及び A_2 のうち $\mathcal{J}$ から成る行列を R' 及び A'_2 とする．この R'，A'_2 を用いた式は，

$$(1, x_1, ..., x_{r-1})A_1R' = (1, x_1, ..., x_{r-1})A'_2 \qquad (6)$$

となる．$\mathcal{J}$ を選択した式 (6) は，$\mathbb{F}_q$ 上の $m(n-k-1)$ 個の等式及び $r(z+k+1)+m(r-1)$ 個の未知数を持つ．この式 (6) を解くための Ourivski-Johansson の提案攻撃は二種類存在する．両手法の共通点は，式 (6) 内の値を推測，列挙し，線形方程式系に変換してから解くことである．一つの手法は，二次の項に影響を与える値のうち，$\bar{e}$ の基底である X の値を推測する．x_0 は単位元 1 であるため，それ以外の $r-1$ 個の x_i を推測することになる．よって，推測する値は，$\mathbb{F}_q$ 上の $(r-1)m$ 個である．推測後に得られる線形方程式系は，$\mathbb{F}_q$ 上の mz 個の等式及び $z+k+1+m(r-1)$ 個の未知の変数を持つ．計算量は $O\left((rm)^3 q^{(m-r)(k+1)+2}\right)$ である．二つ目の手法は，二次の項に影響を与える値のうち，A_1 及び A'_2 の $\alpha_{i,j}$ を推測する．単位元 $x_0 = 1$ と掛け合わされる A_1 及び A'_2 の最初の一行目は，掛けても二次の項にならないため推測の対象から除外される．よって，推測する値は，$(r-1)(k+1)+(r-1)z = (r-1)(k+z+1)$ 個の q 進数となる．推測後に得られるシステムは，$\mathbb{F}_q$ 上の mz 個の等式及び $r(m+k+1)$ 個の未知の変数を持つ．計算量は $O\left((k+r)^3 r^3 q^{(m-r)(r-1)+2}\right)$ である．

3.2 Levy-Perret 攻撃

2006 年に提案された Levy-Perret 攻撃 [8] は，Ourivski-Johansson 攻撃 [9] を基にしている．

式 (4) にシンドローム計算式を加えた次の式を考える．

$$\begin{cases} (1, x_1, ..., x_{r-1})A_1R = (1, x_1, ..., x_{r-1})A_2 \\ (1, x_1, ..., x_{r-1})AH^T = yH^T \end{cases} \qquad (7)$$

式 (7) 中，シンドローム計算式内のベクトル $y\,(= mG+e) \in \mathbb{F}_{q^m}^n$ は受信語である．この式 (7) を $\mathbb{F}_q$ 上の $\mathbb{F}_{q^m}$ の基底 Ω を用いて解くので，$\mathbb{F}_q$ 上の $m(2(n-k)-1)$ 個の等式及び $nr+m(r-1)$ 個の未知の変数を持つ．式 (7) を解く手順は次の通りである．手順内で使用する Gröbner 基底アルゴリズムについては後述する．Levy-Perret の手法では，X 及び A を変数とした多変数多項式系 (7) を，Gröbner 基底アルゴリズムを用いることで解きやすい多項式に変換し，X 及び A を求めている．

(1) 式 (7) に対して Gröbner 基底アルゴリズム (F_4 アルゴリズム) を実行し，$\mathbb{F}_q$ 上の変数 $\mathcal{V}$ を手に入れる．
(2) 変数 $\mathcal{V}$ を $(\bar{X}, \bar{A}) \in \mathbb{F}_q^{m\times r} \times \mathbb{F}_q^{r\times n}$ として組み立て直す．
(3) $\bar{e} = \bar{X}\bar{A}$ 及び $\mathrm{Rk}(\bar{e}\,|\,\mathbb{F}_q)$ を計算する．
(4) $\mathrm{Rk}(\bar{e}\,|\,\mathbb{F}_q) \leq r$ 及び $y - \bar{e} \in C_r$ を確認する．
(5) $\bar{e}H^T$ 及び $cH^T = \epsilon\bar{e}H^T$ から $\epsilon \in \mathbb{F}_{q^m}^*$ を計算する．
(6) $e = \epsilon\bar{e}$ を計算する．

Gröbner 基底とは，端的に表現すると，解くのが難しい多項式の集合に対して，その多項式と等価な解きやすい多項式の集合のことを指す．Gröbner 基底アルゴリズムとは Gröbner 基底を求めるアルゴリズムであり，端的に表現すると，連立多項式を解くアルゴリズムである．

4. 代数攻撃に関する提案

Levy-Perret 攻撃 [8](以下，LP 攻撃と呼ぶ．) に関する提案について記す．

4.1 Levy-Perret 攻撃の実装に関する提案

LP 攻撃は，式 (7) を主軸としており，式 (7) に対して Gröbner 基底アルゴリズム (F_4 アルゴリズム) を用いる．この式 (7) は，生成行列由来の式とパリティ検査行列由来の式から成る．誤りベクトルに注目すると，$\epsilon \in \mathbb{F}_{q^m}^*$ を用いて，$e = \epsilon\bar{e}$ と表すことができた．この $\bar{e}$ は基底を用いて $\bar{e} = XA$ と書くことができ，ϵ を用いるため，X は単位元 1 を含む必要があり $X = (1, X_1, ..., X_{r-1})$ となる．このことを踏まえて，式 (7) に再度注目する．式 (7) の二つ目の式について，左辺は

$$XAH^T = \bar{e}H^T$$

であるのに対して，右辺は

$$yH^T = (mG+e)H^T = eH^T = \epsilon\bar{e}H^T$$

となる．そのため，式 (7) の二つ目の式は成り立たず，

$$\bar{e}H^T \neq \epsilon\bar{e}H^T$$

である．$\bar{e}$ のシンドロームを $\bar{e}H^T = \sigma = (\sigma_1, ..., \sigma_{n-k})$ とすると，e のシンドローム $s = (s_1, ..., s_{n-k})$ は，$eH^T = \epsilon\bar{e}H^T = \epsilon\sigma = (\epsilon\sigma_1, ..., \epsilon\sigma_{n-k})$ と表記でき，$s = \epsilon\sigma$ である．ここで，s と σ の各要素の比を求める．任意の $1 \leq i, j \leq n-k$ について，$s_i : s_j = \sigma_i : \sigma_j$ であり，$s_i \neq 0$ のとき，

$$\sigma_i \frac{s_j}{s_i} = \sigma_j \qquad (8)$$

が成り立つ．パリティ検査行列由来の式として式 (8) を用いて式 (7) を書き直すと，

$$\begin{cases} (1, x_1, ..., x_{r-1})A_1R = (1, x_1, ..., x_{r-1})A_2 \\ \sigma_i \dfrac{s_j}{s_i} = \sigma_j, \quad 1 \leq i, j \leq n-k,\ i \neq j,\ s_i \neq 0 \end{cases} \qquad (9)$$

となる．

4.2 行列 A の組織的符号化

LP 攻撃において，式 (7) 内で用いられている行列 A は

$$A = \begin{pmatrix} \alpha_{0,1} & \cdots & \alpha_{0,n} \\ \alpha_{1,1} & \cdots & \alpha_{1,n} \\ \vdots & \ddots & \vdots \\ \alpha_{r-1,1} & \cdots & \alpha_{r-1,n} \end{pmatrix}$$

という形式であった．行列 A は，誤りベクトル $\bar{e}$ の取り得る基底の組み合わせを表している．この行列 A を組織的符号のように扱うことで変数の数を削減可能となる．つまり，行列 A にガウスの消去法を適用して，

$$A' = \begin{pmatrix} & & \alpha_{0,r+1} & \cdots & \alpha_{0,n} \\ & & \alpha_{1,r+1} & \cdots & \alpha_{1,n} \\ & I & \vdots & \ddots & \vdots \\ & & \alpha_{r-1,r+1} & \cdots & \alpha_{r-1,n} \end{pmatrix}$$

とする．行列 A' 内の I は $r \times r$ の単位行列である．なお，行列 A にガウスの消去法を適用した行階段系の行列は，左端が単位行列の形式をしているとは限らない．そのため，行列 A' を用いた場合の攻撃は失敗する可能性がある．つまり，攻撃の失敗の可能性と，変数の数とのトレードオフの関係性があることが分かる．この方法により，$\mathbb{F}_q$ 上の変数を r^2 個減らすことができる．

例として，$r = 3$ の場合について述べる．$r = 3$ のときの e の基底を $\{\beta_1, \beta_2, \beta_3\}$ とする．単位元 1 を含む $\bar{e}$ の基底の候補は，e の基底より，$\{1, \frac{\beta_2}{\beta_1}, \frac{\beta_3}{\beta_1}\}$，$\{1, \frac{\beta_1}{\beta_2}, \frac{\beta_3}{\beta_2}\}$ 若しくは $\{1, \frac{\beta_1}{\beta_3}, \frac{\beta_2}{\beta_3}\}$ という形式をしている．ここで，$\{\beta_1, \beta_2, \beta_3\}$ は e の基底であることから線形独立であるので，各 $\bar{e}$ の基底の候補も線形独立となる．よって，ベクトル e の初めの r 個の要素に 0 を含まないならば，A を組織的符号の形式である A' で表せる可能性は高いと考える．

この A' を式 (9) に用いる．

$$\begin{cases} (1, x_1, ..., x_{r-1}) A'_1 R = (1, x_1, ..., x_{r-1}) A'_2 \\ \sigma_i \dfrac{s_j}{s_i} = \sigma_j, \quad 1 \leq i, j \leq n-k,\ i \neq j,\ s_i \neq 0 \end{cases} \tag{10}$$

とする．なお，σ_i，σ_j の計算にも A' を使用する．

検証については， 基本的にこの式 (10) を用いた．

5. 代数攻撃に関する検証

LP 攻撃に関する検証内容を記す．また，検証においては $q = 2$，パリティ検査行列の次元 d について $d = 3$ とした．なお，結果を示した各表内の攻撃時間の単位は全て秒である．実行環境は以下の通りである．

CPU: Intel(R) Xeon(R) CPU E7-4830 v4 @ 2.00GHz
RAM: 3TB
OS: Ubuntu 16.04.5 LTS
言語: Magma version 2.24-1

5.1 基底の変数に関する検証

X 中の単位元 1 を除く x_i に変数を充てることになる．ここで，変数の取り方は二通り考えることができる．一つ目は $\mathbb{F}_2$ 上の $\mathbb{F}_{2^m}$ の基底 $\Omega = (\omega_0, ..., \omega_{m-1})$ を用いる方法である．この場合，X 内に $\mathbb{F}_2$ 上の変数が $m(r-1)$ 個存在することになるので，A' を合わせて変数の合計は $m(r-1) + r(n-r)$ となる．二つ目は，各 x_i を $\mathbb{F}_{2^m}$ 上の変数として扱う方法である．ここで，A' は $\mathbb{F}_2$ 上の行列であるが，magma の使用上，$\mathbb{F}_{2^m}$ 上の変数と $\mathbb{F}_2$ 上の変数とを同時に扱うことができない．そのため，A' の各変数 $\alpha_{i,j}$ については，$\mathbb{F}_{2^m}$ 上の変数を充当する他，$\alpha_{i,j}^2 - \alpha_{i,j} = 0$ という等式と一緒に用いることで，疑似的に $\mathbb{F}_2$ 上の変数として振る舞うようにした．この場合，X 内に $\mathbb{F}_{2^m}$ 上の変数が $r-1$ 個，全体で $r - 1 + r(n-r)$ 個となる．以上をまとめると，次のようになる．

CASE-a0: 各 x_i について，Ω の $\mathbb{F}_2$ 上の変数を使用．
CASE-a1: 各 x_i について，$\mathbb{F}_{2^m}$ 上の変数を使用．

CASE-a0 及び CASE-a1 に対する検証結果を表 1 に示す．なお，誤りベクトルの次元 r について $r = 2$ とした．変数の順序については，基底 X から先に充て，続いて A' とした．このときの変数の順序を $X \to A$ と記述する．また，式 (10) の両方の等式を実装している．検証結果として，実行時間とは別に d_{reg} を求めた．この d_{reg} は，F_4 アルゴリズム実行結果内の各 step degree の最大値のことである．d_{reg} は F_4 アルゴリズムにおいて計算量の指標とされる．

表 1 より，CASE-a1 は実行時間が増加していることが分かる．また，F_4 アルゴリズムの計算量の指標である d_{reg} も CASE-a1 の方が高くある傾向にある．その原因一つとして，行列 A' の各要素 $\alpha_{i,j}$ に $\mathbb{F}_{2^m}$ 上の変数を充てた上で，等式 $\alpha_{i,j}^2 - \alpha_{i,j} = 0$ をイデアルに追加し，$\mathbb{F}_2$ 上の変数として扱おうとしたことが挙げらるだろう．イデアルに等式を追加しても，もとが $\mathbb{F}_{2^m}$ 上の変数であるため，解の候補が多いことには変わりない．また，F_4 アルゴリズムの入力であるイデアル内の式が多くなると，処理に時間も増加する傾向にある．

5.2 変数の順序に関する検証

変数を充てる箇所が X と A' の二か所であるため，変数を充てる順序が生じる．順序について，次の二つの場合を考える．

CASE-b0: 先に X，続いて A' $(X \to A)$.
CASE-b1: 先に A'，続いて X $(A \to X)$.

CASE-b0 及び CASE-b1 に関する検証結果を表 2 に示す．なお，$r = 2$ としている．基底の変数は CASE-a0 に準

じた．また，式 (10) の両方の等式を実装している．

表 2 より，変数の順序を変えるだけで実行時間に変化が生じる．また，CASE-b1 では，同じパラメータでも d_{reg} が異なる場合があり，その d_{reg} に実行時間も即した値になっている．全体的な傾向として，CASE-b1 の方が実行時間が増加している．

5.3 等式の選択に関する検証

式 (10) は，生成行列由来の等式 $(1, x_1, ..., x_{r-1})A_1'R = (1, x_1, ..., x_{r-1})A_2'$ 及びパリティ検査行列由来の等式 $\sigma_i \frac{s_j}{s_i} = \sigma_j$ から成り立っている．そのため，等式の組み合わせは三通り存在する．両方の等式を用いる場合，各等式の攻撃への寄与の度合いを知ることができない．しかし，別々に用いることで，それぞれの等式がどの程度攻撃に貢献できるかが分かると考えた．次の三つの場合について検証した．

CASE-c0: 生成行列由来の等式のみ使用．

CASE-c1: パリティ検査行列由来の等式のみ使用．

CASE-c2: 両方の等式を使用．

CASE-c0，CASE-c1 及び CASE-c2 に関する検証結果を表 3 に示す．なお，$r = 2$ としている．変数の順序については，$X \to A$ とした．また，基底の変数は CASE-a0 に準じた．

表 3 より，各パラメータにおいて，CASE-c1 と CASE-c2 は同じような結果を示している．これより，式 (10) のうちパリティ検査行列由来の等式は，攻撃への貢献度が低いとも言うことができる．この結果より，LP 攻撃はより最適な多項式系を形成する余地があると考える．

5.4 QC 構造の影響に関する検証

LRPC 符号は $\frac{n}{n-k}$ が割り切れるとき，パリティ検査行列内の $(n-k) \times (n-k)$ の各ブロックは巡回行列の形を取ることができる．各ブロックが巡回行列である LRPC 符号を QC-LRPC 符号と呼ぶ．LRPC 符号を用いた場合と QC-LRPC 符号を用いた場合との検証結果を示す．QC-LRPC 符号については，変数の順序を加味して，計三通りの場合を考える．

CASE-d0: LRPC 符号を使用 $(X \to A)$.

CASE-d1: QC-LRPC 符号を使用 $(X \to A)$.

CASE-d2: QC-LRPC 符号を使用 $(A \to X)$.

CASE-d0，CASE-d1 及び CASE-d2 に関する検証結果を表 4 に示す．なお，$r = 2$ としている．基底の変数は CASE-a0 に．等式の選択は CASE-c2 に準じた．

表 4 より，各パラメータにおいて，QC-LRPC 符号を用いた CASE-d1 及び CASE-d2 では d_{reg} が全て 4 であるが安定していることが分かる．

5.5 r の増加に関する検証

$r = 2$ と $r = 3$ の場合を比較するとともに，変数の順序を加味して，次の三通りの場合を検証した．

CASE-e0: $r = 2$ の場合 $(X \to A)$.
CASE-e1: $r = 3$ の場合 $(X \to A)$.
CASE-e2: $r = 3$ の場合 $(A \to X)$.

CASE-e0，CASE-e1 及び CASE-e2 に関する検証結果を表 5 に示す．なお，誤り訂正符号として LRPC 符号を用いた．基底の変数は CASE-a0 に，等式の選択は CASE-c2 に準じた．表 5 中の—は，処理が重すぎて観測できなかった箇所である．

128 ビットセキュリティの LRPC McEliece 暗号 [3] において $r = 5$ であることと比較して，$r = 2$ や $r = 3$ はとても小さい値である．$r = 3$ の場合で観測できない箇所が存在することより，実行環境を見直す必要がある．

6. おわりに

LRPC 符号ベース暗号 [3] は 2013 年に提案されて以来，従来の符号ベース暗号と比較して公開鍵長が小さいことにより注目を集めている．代数攻撃は，LRPC 符号ベース暗号が提案される以前から存在する．しかし，LRPC 符号ベース暗号に対する代数攻撃に関する研究が少なく，LRPC 符号が代数攻撃に対して真に安全であるかは不明であった．本研究では，先ず Levy-Perret 攻撃 [8] の改良について二つ提案した．一つ目は，Levy-Perret 攻撃は生成行列由来の等式とパリティ検査行列由来の等式とを併せて多変数多項式系としているが，パリティ検査行列由来の等式を実装可能な形を示した．二つ目は，誤りベクトルの基底の取り方の行列に関して，変数の数を削減する手法である．続いて，いくつか条件を変え，小さいパラメータの LRPC 符号ベース暗号に対して Levy-Perrret 攻撃の検証を行った．特に注目すべき検証結果はパリティ検査行列由来の等式は攻撃に貢献する度合いが低いことである．この結果より，Levy-Perret 攻撃はより最適な多項式系を形成できる可能性が示唆された．

本研究における検証パラメータは，実際に提案されているパラメータと比較して誤りベクトルの次元 r が特に小さい．そのため，実際のパラメータに置き換えた際に検証結果と同じことが言えるかは不明である．今後の課題として，実際のパラメータと同程度の大きさで検証を行う必要性が挙げられる．

謝辞 本研究の一部は JSPS 科研費基盤 C (JP15K00183), Microsoft Research Asia の共同研究費, 科学技術振興機構（JST）の CREST(JPMJCR1404) と国際科学技術協力基盤整備事業（日本—台湾研究交流），及び文部科学省の情報技術人材育成のための実践教育ネットワーク形成事業 分野・地域を越えた実践的情報教育協働ネットワークさらに文部科学省の平成 30 年度「Society

5.0 実現化研究拠点支援事業」の助成を受けています.

n	k	m	CASE-a0		CASE-a1	
			d_{reg}	time	d_{reg}	time
18	9	15	3	0.063	4	36.4
24	12	20	3	0.293	4	1070
30	15	25	4	5.97	4	10500

表 1 基底の変数の取り方による攻撃結果の比較

n	k	m	CASE-b0		CASE-b1	
			d_{reg}	time	d_{reg}	time
24	12	20	3	0.293	3	0.328
30	15	25	4	5.97	3	1.00
36	18	30	4	10.1	3	3.00
					4	32.0
42	21	35	3	7.32	3	7.23
48	24	40	3	10.4	3	19.1
54	27	45	3	19.2	4	895
60	30	50	4	584	3	83.7
66	33	55	3	68.2	3	172
72	36	60	3	120	3	340
					4	1,450
78	39	65	3	174	4	4,390
84	42	70	3	338	3	1,150
90	45	75	3	470	3	1,910
96	48	80	3	715	3	3,200
					4	90,000

表 2 変数の順序による攻撃結果の比較

n	k	m	CASE-c0		CASE-c1		CASE-c2	
			d_{reg}	time	d_{reg}	time	d_{reg}	time
24	12	20	3	0.196	3	0.330	3	0.293
30	15	25	4	5.75	4	6.15	4	5.97
36	18	30	4	10.1	4	10.8	4	10.3
42	21	35	4	6.21	4	7.36	3	7.32
48	24	40	3	7.00	3	10.5	3	10.4
54	27	45	3	12.6	3	19.3	3	19.2
60	30	50	4	622	4	579	4	584
66	33	55	3	40.8	3	59.4	3	68.2
72	36	60	3	73.6	3	111	3	120
78	39	65	3	115	3	176	3	174
84	42	70	3	211	3	329	3	338
90	45	75	3	289	3	464	3	470
96	48	80	3	437	3	754	3	715

表 3 式 (10) 内の各等式の攻撃結果の比較

n	k	m	CASE-d0		CASE-d1		CASE-d2	
			d_{reg}	time	d_{reg}	time	d_{reg}	time
24	12	20	3	0.293	4	0.535	4	0.533
30	15	25	4	5.97	4	5.57	4	1.38
36	18	30	4	10.3	4	8.56	4	8.62
48	24	40	3	10.4	4	61.5	4	59.3
60	30	50	4	584	4	295	4	276
72	36	60	3	120	4	2,530	4	2,120
84	42	70	3	338	4	7,600	4	5,800
96	48	80	3	715	4	22,000	4	16,200

表 4 LRPC 符号及び QC-LRPC 符号に対する攻撃結果の比較

n	k	m	CASE-e0		CASE-e1		CASE-e2	
			d_{reg}	time	d_{reg}	time	d_{reg}	time
24	12	20	3	0.293	5	12,000	4	425
30	15	25	4	5.97	5	126,000	4	2,330
							5	64,500
							7	60,000
36	18	30	4	10.3	5	953,000	4	12,700
							—	—
42	21	35	3	7.32	—	—	4	53,000
							—	—

表 5 $r = 2$ の場合及び $r = 3$ の場合に対する攻撃結果の比較

参考文献

[1] Robert J. McEliece. A public-key cryptosystem based on algebraic coding theory. JPL DSN Progress Report, 44:114–116, 05 1978.

[2] Harald Niederreiter. Knapsack type cryptosystems and algebraic coding theory. Problems of Control and Information Theory, 15:159–166, 01 1986.

[3] Philippe Gaborit, Gaétan Murat, Olivier Ruatta, and Gilles Zémor. Low rank parity check codes and their application to cryptography. In Matthew G. Parker Lilya Budaghyan, Tor Helleseth, editor, The International Workshop on Coding and Cryptography (WCC 13), page 13 p., Bergen, Norway, April 2013.

[4] Ernst M. Gabidulin. Theory of codes with maximum rank distance (translation). Problems of Information Transmission, 21:1–12, 01 1985.

[5] Ernst M. Gabidulin, A. V. Paramonov, and O. V. Tretjakov. Ideals over a non-commutative ring and their applications in cryptology. In Advances in Cryptology - EUROCRYPT' 91, Workshop on the Theory and Application of of Cryptographic Techniques, Brighton, UK, April 8-11, 1991, Proceedings, pages 482–489, 1991.

[6] Raphael Overbeck. Structural attacks for public key cryptosystems based on gabidulin codes. J. Cryptology, 21(2):280–301, 2008.

[7] Ernst M. Gabidulin. Attacks and counter-attacks on the GPT public key cryptosystem. Des. Codes Cryptography, 48(2):171–177, 2008.

[8] Françoise Levy-dit-Vehel and Ludovic Perret. Algebraic Decoding of Rank Metric Codes. Proceedings of YACC 2006.

[9] Alexei V. Ourivski and Thomas Johansson. New technique for decoding codes in the rank metric and its cryptography applications. Probl. Inf. Transm., 38(3):237–246, 2002.

[10] Florent Chabaud and Jacques Stern. The cryptographic security of the syndrome decoding problem for rank distance codes. In Advances in Cryptology - ASIACRYPT '96, International Conference on the Theory and Applications of Cryptology and Information Security, Kyongju, Korea, November 3–7, 1996, Proceedings, pages 368–381, 1996.

[11] Philippe Gaborit and Gilles Zémor. On the hardness of the decoding and the minimum distance problems for rank codes. IEEE Trans. Information Theory, 62(12):7245–7252, 2016.