List decoding for Reed-Muller codes and its application to polar codes

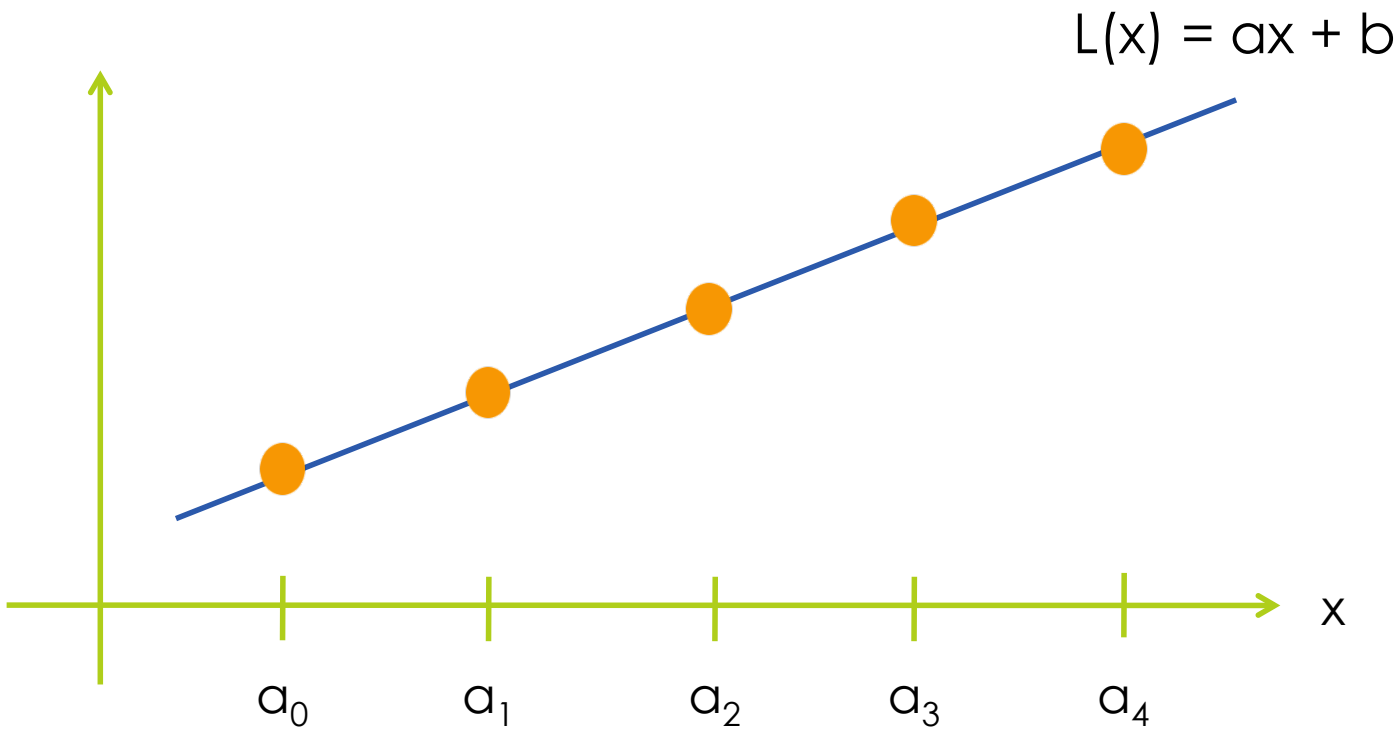
安永 憲司
東京工業大学 大学院情報理工学研究科 数理・計算科学専攻

発表内容

- リスト復号
- Reed-Muller 符号
  - 定義
  - Plotkin 構成
- 研究成果
  - RM符号のリスト復号アルゴリズム
  - Polar 符号への適用
- まとめ

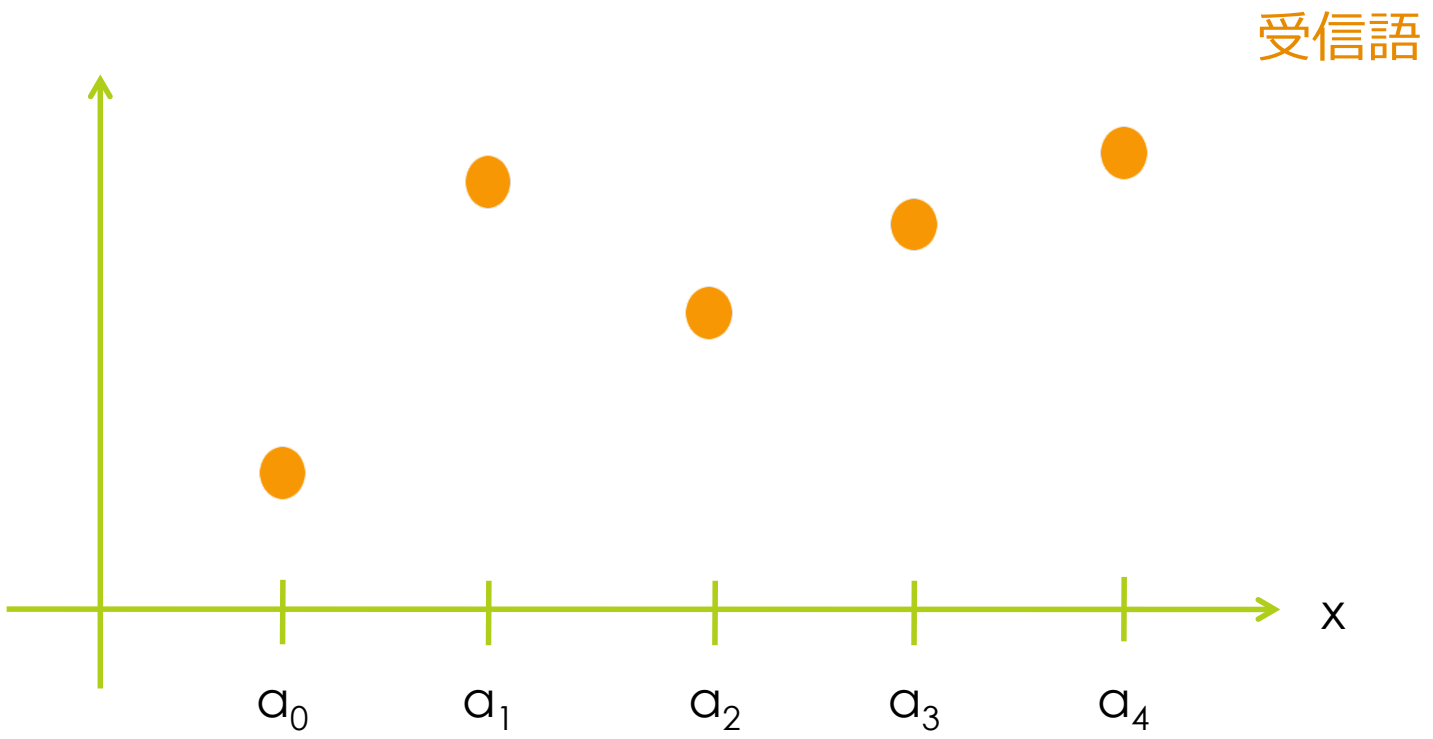
多項式をメッセージとする符号

- メッセージ L(x) に対する符号語は
  ($L(a_0)$, $L(a_1)$, $L(a_2)$, $L(a_3)$, $L(a_4)$)



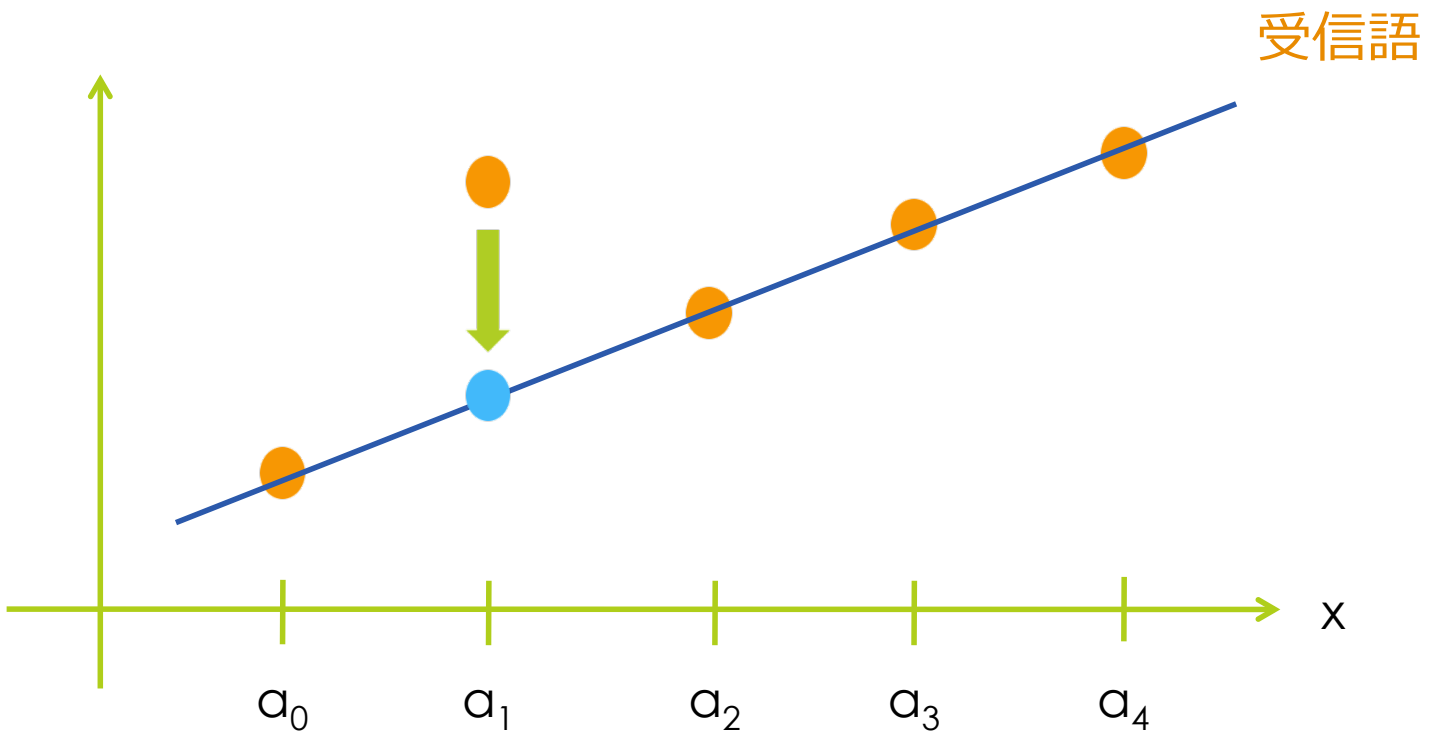
多項式をメッセージとする符号

- 誤り訂正は多項式を補間すること



多項式をメッセージとする符号

- 誤り訂正は多項式を補間すること

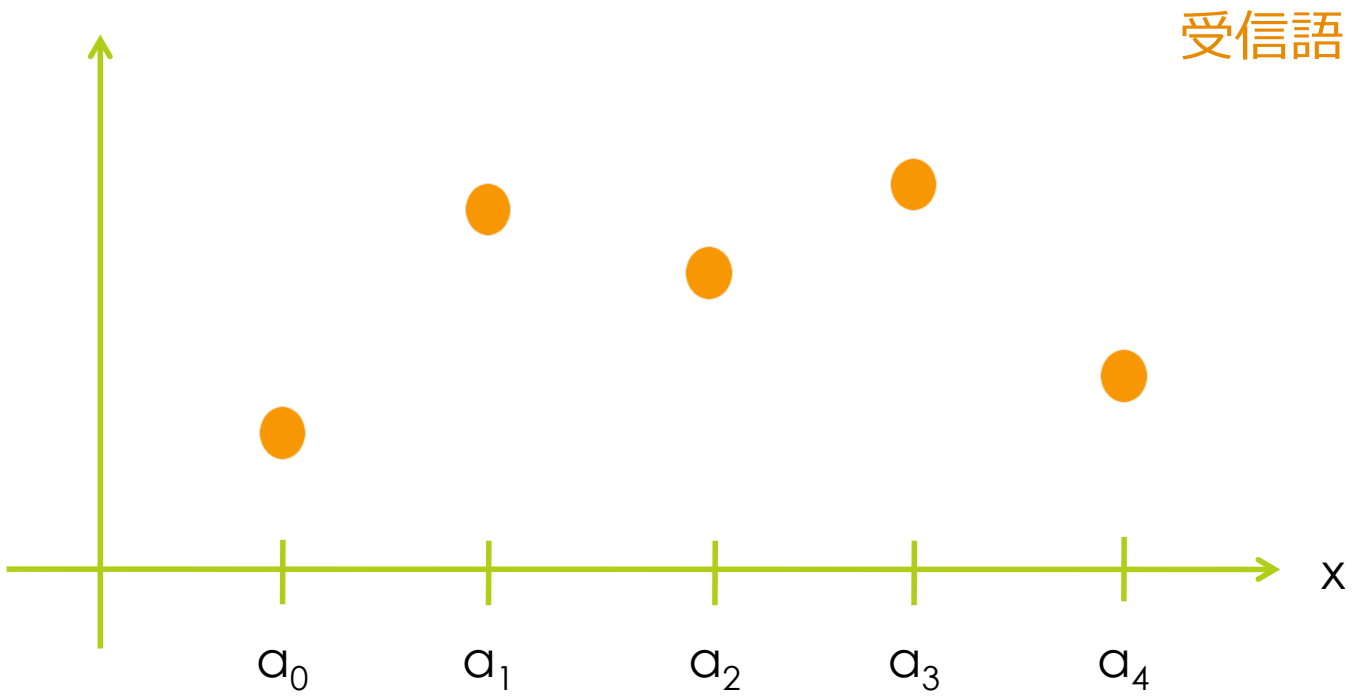


多項式をメッセージとする符号

- 1 変数多項式 → Reed-Solomon 符号
- 多変数多項式 → Reed-Muller 符号

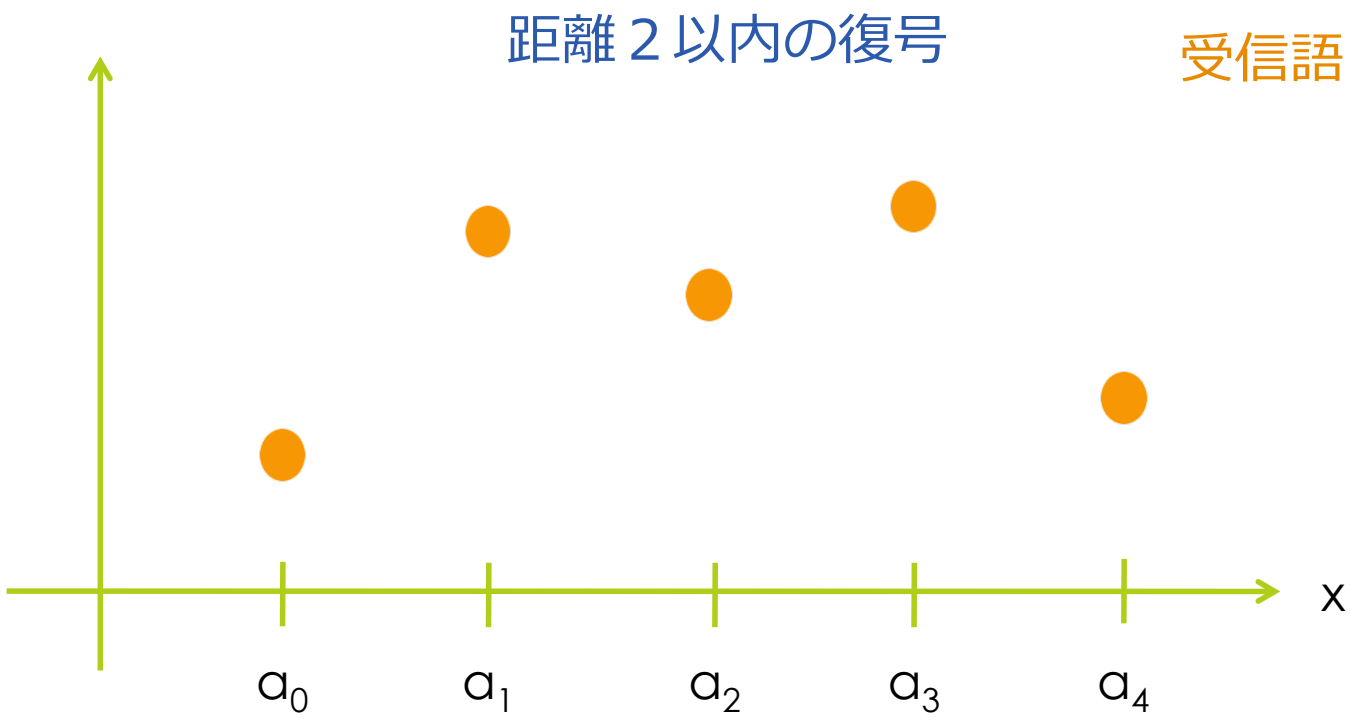
リスト復号

- 受信語から、与えられた距離以内（リスト復号半径）にあるすべての符号語を出力する復号法



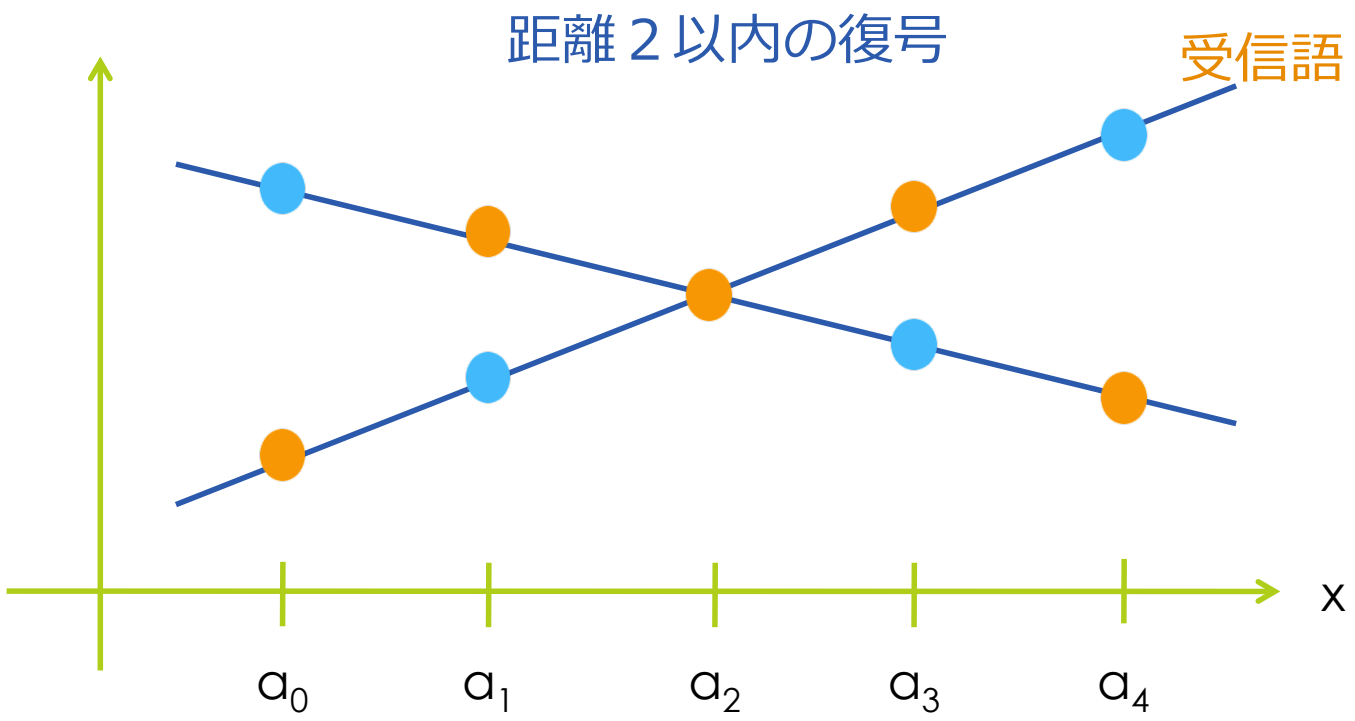
リスト復号

- 受信語から、与えられた距離以内（リスト復号半径）にあるすべての符号語を出力する復号法



リスト復号

- 受信語から、与えられた距離以内（リスト復号半径）にあるすべての符号語を出力する復号法



リスト復号

- アイディア [Elias 1957][Wozencraft 1958]
- 多項式時間アルゴリズム
  - Reed-Solomon 符号 [Sudan 1997][Guruswami, Sudan 1999]
  - 代数幾何符号 [Shokrollahi, Wasserman 1999][Guruswami, Sudan 1999]
  - BCH 符号 [Wu 2008]
  - 連接符号[Guruswami, Sudan 2000]
  - Reed-Muller 符号 [Sudan, Trevisan, Vadhan 1999] [Pellikaan, Wu 2004][Gopalan, Klivans, Zuckerman 2008]
  - folded Reed-Solomon符号 [Guruswami, Rudra 2006]
  - テンソル積符号・インターリーブ符号 [Gopalan, Guruswami, Raghavendra 2009]

Reed-Muller 符号

- $RM_q(r, m)$: 有限体 $\mathbf{F}_q$ 上の長さ q^m の r 次 RM 符号

- メッセージ集合は、
  $\mathbf{F}_q$ 上 m 変数 r 次以下多項式の集合

- $\mathbf{F}_q = \{ a_0, a_1, \ldots, a_{q-1} \}$ としたとき、
  多項式 $p(x_1, x_2, \ldots, x_m)$ に対応する符号語は

$$
\begin{aligned}
&(p(a_0, a_0, \ldots, a_0), p(a_1, a_0, \ldots, a_0), \ldots, p(a_{q-1}, a_{q-1}, \ldots, a_{q-1}, a_0), \\
&\quad p(a_0, a_0, \ldots, a_1), p(a_1, a_0, \ldots, a_1), \ldots, p(a_{q-1}, a_{q-1}, \ldots, a_{q-1}, a_1), \\
&\qquad \ldots, \\
&\quad p(a_0, a_0, \ldots, a_{q-1}), p(a_1, a_0, \ldots, a_{q-1}), \ldots, p(a_{q-1}, a_{q-1}, \ldots, a_{q-1}, a_{q-1}))
\end{aligned}
$$

$RM_2(2, 2)$ 符号

- メッセージ集合；

  $0, 1, x_1, x_1+1, x_2, x_2+1, x_1+x_2, x_1+x_2+1,$
  $x_1x_2, x_1x_2+1, x_1x_2+x_1, x_1x_2+x_1+1, x_1x_2+x_2, x_1x_2+x_2+1,$
  $x_1x_2+x_1+x_2, x_1x_2+x_1+x_2+1$

- $p(x_1, x_2) = x_1x_2+x_1+1$ の符号語；$\mathbf{F}_2 = \{a_0 = 0, a_1 = 1\}$

  $p(0, 0) =1, p(1, 0) = 0, p(0, 1) = 1, p(1, 1) = 1$ なので、

  符号語は $(1, 0, 1, 1)$

RM 符号のリスト復号

- [Sudan, Trevisan, Vadhan 1999] [Pellikaan, Wu 2004]
  - [Guruswami, Sudan 1999] をもとにしたアルゴリズム
- [Gopalan, Klivans, Zuckerman 2008]
  - リスト復号半径が Johnson bound を超える
    - 既存のはすべて $J(2^{-r})$ 以下、この復号法では $J(2^{1-r})$
      - $J(\delta) = (1 - (1 - 2\delta)^{1/2})/2$
  - r = 2 でリスト復号半径 $\eta < 1/2$ まで可能（最小距離の 2 倍）
- [Dumer, Kabatiansky, Tavernier 2008]
  - q = 2 について [GKZ08] アルゴリズムの時間計算量を改善
  - Plotkin 構成を利用している点に着目

本研究の成果

- GKZ アルゴリズムの [DKT08] による改良のアイディアを q >2 の場合に拡張
  - Plotkin 構成を q > 2 に拡張したものを利用
  - 時間計算量は [GKZ08] と単純には比較できない
- polar 符号への適用
  - GKZ-アルゴリズムが適用可能な polar 符号の条件を導出

Plotkin 構成

- 符号長の等しい２つの符号から新しい符号を構成する方法 [Plotkin 1960]

$$C = \{ u \bigcirc (u + v) : u \in C_1, v \in C_2 \}$$

- $RM_2(r, m)$

$$= \{ u \bigcirc (u+v) : u \in RM_2(r, m-1), v \in RM_2(r-1, m-1) \}$$

$$= \{ (u+v) \bigcirc u : u \in RM_2(r, m-1), v \in RM_2(r-1, m-1) \}$$

[DKT08] アルゴリズムのアイディア

- $y = y_0 \bigcirc y_1$ を受信
- $p = (u \bigcirc (u+v))$ or $((u'+v') \bigcirc u')$
  $u, u' \in RM_2(r, m-1), v, v' \in RM_2(r-1, m-1)$
  を求めればよい
- $\Delta(y, p) \leq \eta$ とすると

  $\Delta(x, y)$ = (x と y の相対距離)

  1. $\Delta(y_0, u) \leq \eta$ or $\Delta(y_1, u') \leq \eta$
  2. $\Delta(y_0+y_1 , v) \leq 2\eta$
  3. $\Delta(y_0+y_1 , v') \leq 2\eta$

  - $RM_2(r, m-1)$ に対して距離 η を訂正
    $RM_2(r-1, m-1)$ に対して距離 2η を訂正　できればOK

[DKT08] アルゴリズム

- $RM_2(r, m)$ に対するリスト復号半径 η の復号法
- $\mathrm{ListDec}_2(r, m, \eta)$:
  1. $y = y_0 \circ y_1$ を受信
  2. y_0 を $\mathrm{ListDec}_2(r, m-1, \eta)$ へ入力 → L_0 を出力
  3. y_1 を $\mathrm{ListDec}_2(r, m-1, \eta)$ へ入力 → L_1 を出力
  4. y_0+y_1 を $\mathrm{ListDec}_2(r-1, m-1, 2\eta)$ へ入力 → L_v を出力
  5. 候補符号語集合
     $L' = \{ u \circ (u+v), (u'+v') \circ u' : u \in L_0, u' \in L_1, v, v' \in L_v \}$
     を構成
  6. $\Delta(y, p) \leq \eta$ である $p \in L'$ をすべて出力

[DKT08] アルゴリズムの動作

- $\mathrm{ListDec}_2(r, m, \eta)$ の実行に、
  $\mathrm{ListDec}_2(r, m-1, \eta)$, $\mathrm{ListDec}_2(r-1, m-1, 2\eta)$ を呼び出す
- ベースケースは、(1) $2^m \eta < 1$ or (2) $r = 1$
  - (1) の場合、誤り数 = 0 なので正しく返せる
  - (2) の場合、全数探索 （$r = 1$ では全数探索でも多項式時間）
    ただし，$2^{r-1} \eta < 1$ である必要
- $\eta < 2^{1-r}$ であれば，アルゴリズムは正しく動作する
  - $\mathrm{RM}_2(r, m)$ の相対最小距離は 2^{-r}
  - $r \leq m$ は定義より成立するので，r に制限はない

[DKT08] アルゴリズムの時間計算量

- $LS_q(r, m, \eta) = \max |\{ p \in RM_q(r, m) : \Delta(y, p) \leq \eta \}|$
  - max は $\mathbf{F}_q$ 上の長さ q^mのベクトル y すべてでとる

- このとき以下が成立
  1. $LS_2(r, m-1, \eta) \leq LS_2(r, m, \eta)$
  2. $LS_2(r-1, m-1, 2\eta) \leq LS_2(r, m, \eta)$

  → アルゴリズムの時間計算量は r, m, η, $LS_2(r, m, \eta)$ で評価可能

- 時間計算量 $T_2(r, m, \eta) = O(2^{3m} LS_2(r, m, \eta)^2)$
  - [GKZ08] では $O(2^{3m} LS_2(r, m, \eta)^r)$

$\mathbf{F}_q$ (q >2) への拡張のために

$RM_2(r, m)$ の Plotkin 構成

- $RM_2(r, m)$
  $= \{ u \circ (u+v) : u \in RM_2(r, m-1), v \in RM_2(r-1, m-1) \}$
  $= \{ (u+v) \circ u : u \in RM_2(r, m-1), v \in RM_2(r-1, m-1) \}$

これは以下と等価

- $RM_2(r, m)$
  $= \{ p_0(x_1, \ldots, x_{m-1}) + (x_m - b_0)p_1(x_1, \ldots, x_{m-1}) :$
  $p_0 \in RM_2(r, m-1), p_1 \in RM_2(r-1, m-1) \}$
  $= \{ (p_0+(a_0 - b_0)p_1) \circ (p_0 + (a_1 - b_0)p_1) :$
  $p_0 \in RM_2(r, m-1), p_1 \in RM_2(r-1, m-1) \}$
  - ただし $b_0 \in \mathbf{F}_2 = \{ a_0, a_1 \}$

Plotkin 構成の $\mathbf{F}_q$ (q >2) への拡張 (1/3)

- $\mathbf{F}_q = \{ b_0, b_1, \ldots, b_{q-1} \}$ を選ぶ
- 任意の $p \in RM_q(r, m)$ は以下のように書ける（$q - 1 \le r$）

$$
\begin{aligned}
& p(x_1, \ldots, x_m) \\
&= p_0(x_1, \ldots, x_{m-1}) + (x_m - b_0)p_1(x_1, \ldots, x_{m-1}) \\
&\quad + (x_m - b_0)(x_m - b_1)p_2(x_1, \ldots, x_{m-1}) \\
&\quad + (x_m - b_0)(x_m - b_1)(x_m - b_2)p_3(x_1, \ldots, x_{m-1}) + \cdots \\
&= \sum_{i=0}^{q-1} p_i(x_{1,\ldots,}x_{m-1}) \prod_{j=0}^{i-1} (x_m - b_j) \qquad \text{ただし } p_i \in RM_q(r - i, m-1)
\end{aligned}
$$

- 結果として（ RM符号は q! 通りの表現が可能 ）

$$
RM_q(r,m) = \left\{ \sum_{i=0}^{q-1} p_i(x_{1,\ldots,}x_{m-1}) \prod_{j=0}^{i-1} (x_m - b_j) \;:\; p_i(x_1,\ldots,x_{m-1}) \in RM_q(r-i, m-1) \right\}
$$

Plotkin 構成の $\mathbf{F}_q$ (q >2) への拡張 (2/3)

ベクトル表現をすると

- $p(x_1, \ldots, x_m)$

$= p(x_1, \ldots, x_{m-1}, a_0) \bigcirc p(x_1, \ldots, x_{m-1}, a_1)$
$\bigcirc \cdot \cdot \cdot \bigcirc p(x_1, \ldots, x_{m-1}, a_{q-1})$

$$= \bigcirc_{i=0}^{q-1} \left(\sum_{j=0}^{q-1} p_j(x_{1,}\ldots,x_{m-1}) \prod_{k=0}^{j-1} (a_i - b_k) \right)$$

Plotkin 構成の $\mathbf{F}_q$ (q >2) への拡張 (3/3)

- q = 3 の場合を見てみる
- $\mathbf{F}_3 = \{ b_0, b_1, b_2 \}$ を選んだとき、
  任意の $p \in RM_3(r, m)$ は

  $$p(x_1, \ldots, x_m) = p_0 + (x_m - b_0)p_1 + (x_m - b_0)(x_m - b_1)p_2$$
  ただし、$p_i \in RM_q(r - i, m-1)$
- ベクトル表現をすると

  $p(x_1, \ldots, x_m)$
  $= p(x_1, \ldots, a_0) \circ p(x_1, \ldots, a_1) \circ p(x_1, \ldots, a_2)$
  $= p_0 + (a_0 - b_0)p_1 + (a_0 - b_0)(a_0 - b_1)p_2$
  $\circ \; p_0 + (a_1 - b_0)p_1 + (a_1 - b_0)(a_1 - b_1)p_2$
  $\circ \; p_0 + (a_2 - b_0)p_1 + (a_2 - b_0)(a_2 - b_1)p_2$

[DKT08] アルゴリズム を q=3 へ拡張

- $y = y_0 \bigcirc y_1 \bigcirc y_2$ を受信
- ある $p \in RM_3(r, m)$ に対して $\Delta(y, p) \le \eta$
  - $F_3 = \{ b_0, b_1, b_2 \}$ を選ぶと
    $p = p_0 + (x_m - b_0)p_1 + (x_m - b_0)(x_m - b_1)p_2,\ p_i \in RM_q(r - i, m-1)$
- このとき

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \end{bmatrix} \Leftrightarrow \begin{bmatrix} p_0 + (a_0 - b_0)p_1 + (a_0 - b_0)(a_0 - b_1)p_2 \\ p_0 + (a_1 - b_0)p_1 + (a_1 - b_0)(a_1 - b_1)p_2 \\ p_0 + (a_2 - b_0)p_1 + (a_2 - b_0)(a_2 - b_1)p_2 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & a_0 - b_0 & (a_0 - b_0)(a_0 - b_1) \\ 1 & a_1 - b_0 & (a_1 - b_0)(a_1 - b_1) \\ 1 & a_2 - b_0 & (a_2 - b_0)(a_2 - b_1) \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \end{bmatrix}$$

[DKT08] アルゴリズム を q=3 へ拡張

- $(b_0, b_1, b_2) = (a_{i0}, a_{i1}, a_{i2})$ とすると

$$\begin{bmatrix} y_{i0} \\ y_{i1} \\ y_{i2} \end{bmatrix} \Leftrightarrow \begin{bmatrix} 1 & 0 & 0 \\ 1 & a_{i1} - a_{i0} & 0 \\ 1 & a_{i2} - a_{i0} & (a_{i2} - a_{i0})(a_{i2} - a_{i1}) \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \end{bmatrix}$$

- 復号の方針
  - p_0 は y_{i0} から復元
  - p_1 は y_{i0} と y_{i1} から復元（p_0 の代わりに y_{i0} を使う）
  - p_2 は y_{i0}, y_{i1}, y_{i2} から復元（p_0 の代わりに y_{i0} , p_1 の代わりに $(y_{i1} - y_{i0})/(a_{i1} - a_{i0})$ を使う）

  - $RM_3(r, m-1)$ に対し距離 η, $RM_3(r-1, m-1)$ に対し距離 2η, $RM_3(r-2, m-1)$ に対し距離 3η が訂正できればよい

一般の q への拡張

- 同様の方法で任意の $q < r$ へ拡張可能
- リスト復号半径 $\eta < q^{1-r}$ を達成
- 時間計算量
  $T_q(r, m, \eta) = O((q!)^m q^{2m} \prod_{j=0}^{q-1} LS_q(r-j, m-1, (j+1)\eta))$
  - [GKZ08] では $O((q!)^m q^{2m} LS_q(r, m, \eta)^{r+q})$

    $LS_q(r, m, \eta)$:
    半径 η の球に含まれる $RM_q(r, m)$ の符号語の最大数
    $= \max |\{ c : \Delta(c, y) \le \eta, c \in RM_q(r, m) \}|$ $y \in \mathbf{F}_q$

polar 符号への適用

- polar 符号
  - [Arikan 2009] で提案された、任意の 2 元対称離散無記憶通信路において、通信路容量を達成する符号のクラス
  - $G = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ に対し、$G^{\otimes n}$ から上手に行を選び、それを生成行列としたもの
  - ここでは、生成行列が $G^{\otimes n}$ から選ばれる符号を polar 符号と呼ぶ
  - Reed-Muller 符号は、重みの大きい行から順に選んだとき

polar 符号と多項式

- $\mathbf{F}_2 = \{ a_0 = 0, a_1 = 1 \}$ として、
  多項式とベクトルの対応関係を考えると

$$G^{\otimes 3} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \Leftrightarrow \begin{matrix} x_1x_2x_3 \\ x_2x_3 \\ x_1x_3 \\ x_3 \\ x_1x_2 \\ x_2 \\ x_1 \\ 1 \end{matrix}$$

- n 変数のすべての単項式が行として出てくる

GKZ アルゴリズムが適用可能な polar 符号

- 再帰的に Plotkin 構成をもつ符号であればよい

補題. 長さ 2^n の polar 符号 C に対し、
$x_n\, p(x_1, \ldots, x_{n-1}) \in G(C) \rightarrow p(x_1, \ldots, x_{n-1}) \in G(C)$
ならば、C は Plotkin 構成をもつ

- G(C) : 符号 C の生成行列の行集合

定理. C : 長さ 2^n, $\deg(C) = r$ の polar 符号
すべての $0 \le i \le \deg(C)$, $n - \deg(C) \le j \le n$ に対し
$x_i\, p(x_1, \ldots, x_{i-1}) \in G(C(i, j)) \rightarrow p(x_1, \ldots, x_{i-1}) \in G(C(i, j))$
ならば、C に GKZ アルゴリズムが適用可能

- $C(i, j) := C \cap RM(i, j)$
  $\deg(C) := (C \subseteq RM(r, n)$ なる最小の $r)$

まとめ

- 研究成果
  - [DKT08] で提案された RM 符号のリスト復号を $q > 2$ に拡張
    - Plotkin 構成を $q > 2$ へ一般化
    - 時間計算量は [GKZ08] と単純に比較できない
  - GKZ アルゴリズムが適用可能な polar 符号の条件を導出
- 今後の方向性
  - GKZ アルゴリズムが適用可能な polar 符号の具体的構成とその性能評価（リスト復号サイズの評価）
    - RM 符号はとりうるパラメータが少ないのが弱点
  - polar 符号は GKZ 復号法を使って通信路容量を達成可能か？
  - polar 符号はリスト復号容量を達成可能か？
    - リスト復号容量：
      リスト復号半径 η に対してレート $1 - H(\eta)$

リスト復号

- 受信語から、与えられた距離（リスト復号半径）以内にあるすべての符号語を出力する復号法

- 通常の一意復号は、符号語１つを出力

  ⇒ 符号の最小距離が d のとき、
  誤り数 < d/2 でないと、送信語が復元できない

リスト復号なら d/2 を越えても復元可能！

[DKT08]アルゴリズム を q=3 へ拡張 (1/4)

- $y = y_0 \bigcirc y_1 \bigcirc y_2$ を受信
- ある $p \in RM_3(r, m)$ に対して $\Delta(y, p) \le \eta$
  - $F_3 = \{ b_0, b_1, b_2 \}$ を選ぶと
    $p = p_0 + (x_m - b_0)p_1 + (x_m - b_0)(x_m - b_1)p_2,\ p_i \in RM_q(r - i, m-1)$
- このとき

$$\begin{bmatrix} y_0 \\ y_1 \\ y_2 \end{bmatrix} \Leftrightarrow \begin{bmatrix} p_0 + (a_0 - b_0)p_1 + (a_0 - b_0)(a_0 - b_1)p_2 \\ p_0 + (a_1 - b_0)p_1 + (a_1 - b_0)(a_1 - b_1)p_2 \\ p_0 + (a_2 - b_0)p_1 + (a_2 - b_0)(a_2 - b_1)p_2 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & a_0 - b_0 & (a_0 - b_0)(a_0 - b_1) \\ 1 & a_1 - b_0 & (a_1 - b_0)(a_1 - b_1) \\ 1 & a_2 - b_0 & (a_2 - b_0)(a_2 - b_1) \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \end{bmatrix}$$

[DKT08]アルゴリズム を q=3 へ拡張 (2/4)

- $(b_0, b_1, b_2) = (a_{i0}, a_{i1}, a_{i2})$ とすると

$$\begin{bmatrix} y_{i0} \\ y_{i1} \\ y_{i2} \end{bmatrix} \Leftrightarrow \begin{bmatrix} 1 & 0 & 0 \\ 1 & a_{i1} - a_{i0} & 0 \\ 1 & a_{i2} - a_{i0} & (a_{i2} - a_{i0})(a_{i2} - a_{i1}) \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \end{bmatrix}$$

- 復号の方針
  - p_0 は y_{i0} から復元
  - p_1 は y_{i0} と y_{i1} から復元（p_0 の代わりに y_{i0} を使う）
  - p_2 は y_{i0}, y_{i1}, y_{i2} から復元（p_0 の代わりに y_{i0} , p_1 の代わりに $(y_{i1} - y_{i0})/(a_{i1} - a_{i0})$ を使う）

    p_2 の復元には、y_{i0}, y_{i1}, y_{i2} を使うので、誤りがたくさん乗っているが、p_2 の次数は $r - 2$ なので何とかなるかもしれない

[DKT08]アルゴリズム を q=3 へ拡張 (3/4)

- この復号方針は、

$$\begin{bmatrix} 1 & 0 & 0 \\ (a_{i0}-a_{i1})^{-1} & (a_{i1}-a_{i0})^{-1} & 0 \\ ((a_{i0}-a_{i1})(a_{i0}-a_{i2}))^{-1} & ((a_{i1}-a_{i0})(a_{i1}-a_{i2}))^{-1} & ((a_{i2}-a_{i0})(a_{i2}-a_{i1}))^{-1} \end{bmatrix} \begin{bmatrix} y_{i0} \\ y_{i1} \\ y_{i2} \end{bmatrix} \Leftrightarrow \begin{bmatrix} p_0 \\ p_1 \\ p_2 \end{bmatrix}$$

の左側の縦ベクトルを復号器への入力とするもの
（入力を $[w_0, w_1, w_2]$ とおく）

次に、この復号方針でうまくいくことを説明する

[DKT08]アルゴリズム を q=3 へ拡張 (4/4)

- $\Delta_{xi=bj}(y, p)$:= ($x_i = b_j$ である点における相対距離)
- このとき、以下が成立
  $\Delta_{xm=b0}(y, p) \leq \Delta_{xm=b1}(y, p) \leq \Delta_{xm=b2}(y, p)$ ・・・(＊)
  ならば、$\Delta_{xm=bi}(y, p) \leq (3/(3-i))\ \Delta(y, p)$
- 選んできた $\mathbf{F}_3 = \{ b_0, b_1, b_2 \}$ に対し (＊) 成立を仮定
- 復元方針
  - p_0 は y_{i0} から復元
    → $RM_2(r, m-1)$ に対し距離 η を訂正
  - p_1 は y_{i0} と y_{i1} から復元
    → $RM_2(r-1, m-1)$ に対し距離 $((1 + 3/2)/2)\eta \leq (3/2)\eta$ を訂正
  - p_2 は y_{i0}, y_{i1}, y_{i2} から復元
    → $RM_2(r-2, m-1)$ に対し距離 $((1 + 3/2 + 3)/3)\eta \leq 3\eta$ を訂正

q = 3 へ拡張したアルゴリズム

- $RM_3(r, m)$ に対するリスト復号半径 η の復号法
- $\mathrm{ListDec}_3(r, m, \eta)$:
  1. $y = y_0 \circ y_1 \circ y_2$ を受信
  2. すべての $\mathbf{F}_3 = \{ b_0, b_1, b_2 \}$ の組み合わせに対して
     1. w_0 を $\mathrm{ListDec}_2(r, m-1, \eta)$ へ入力 → L_0 を出力
     2. w_1 を $\mathrm{ListDec}_2(r-1, m-1, (3/2)\eta)$ へ入力 → L_1 を出力
     3. w_2 を $\mathrm{ListDec}_2(r-2, m-1, 3\eta)$ へ入力 → L_2 を出力
     4. 候補符号語集合 L' を構成
        $L' = \{ p = p_0 + (x_m - b_0)p_1 + (x_m - b_0)(x_m - b_1)p_2$
        $: p_0 \in L_0, p_1 \in L_1, p_2 \in L_2 \}$
     5. $\Delta(y, p) \leq \eta$ である $p \in L'$ をすべて出力

q = 3 へ拡張したアルゴリズムの時間計算量

- 以下が成立
  1. $LS_3(r, m-1, \eta) \leq LS_3(r, m, \eta)$
  2. $LS_3(r-1, m-1, (3/2)\eta) \leq LS_3(r, m, \eta)$
  3. $LS_3(r-2, m-1, 3\eta) \leq LS_3(r, m, \eta)$

  → 時間計算量は $r, m, \eta, LS_3(r, m, \eta)$ で評価可能

- 時間計算量は $T_3(r, m, \eta) = O(3^{3m} LS_3(r, m, \eta)^3)$

polar 符号の例

$$G^{\otimes 3} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$G(RM_2(1,3)) = \begin{bmatrix} \\ \\ \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

重み４以上の行

$$G(RM_2(2,3)) = \begin{bmatrix} \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

重み２以上の行

GKZ アルゴリズムが適用可能な符号

- $C \subseteq RM_2(r, n)$ が分解可能

  以下を満たす $C_0 \subseteq RM_2(r, n-1)$, $C_1 \subseteq RM_2(r-1, b-1)$ が存在

  $$C = \{ p_0 + (x_n - b)\, p_1 : p_0 \in C_0,\ p_1 \in C_1 \},\ b \in \mathbf{F}_2$$

- C_0, C_1 も分解可能なとき、C は再帰的分解可能

- C が再帰的分解可能 → GKZ アルゴリズムが適用可能