シフト演算を利用した噴水符号
Zigzag Decodable Fountain Codes

野崎隆之

山口大学

2017/1/19

概要

噴水符号

パケット通信向けの誤り訂正符号 (特にマルチキャスト)

発表の構成

1. 噴水符号 (目的，既存の構成法・復号法)
2. シフトを利用した噴水符号 (構成法・復号法・性能解析)

詳しい情報

- 萩原学 編著 “進化する符号理論,” 数学評論社
- A. Shokrollahi, M. Luby, “Raptor Codes,” Foundations and Trends in Communications and Information Theory, vol. 6, nos. 3–4, pp. 213–322, 2011
- T. Nozaki, “Zigzag decodable fountain codes,” arXiv preprint arXiv:1605.09125, 2016.

マルチキャストとその問題点

マルチキャスト

複数のユーザにデータを送る通信方式

問題点

- 送信者は受信者の再送要求に応じることができない
- 受信者によって通信路の状況が異なる

送信者	s_1	s_2	s_3	s_4	s_5	s_6
受信者 A	s_1	s_2	s_3	s_4	s_5	s_6
受信者 B	$\times$	s_2	$\times$	s_4	s_5	s_6
受信者 C	$\times$	$\times$	s_3	$\times$	$\times$	$\times$

噴水符号 [Byers 2002]

[送信者] k 個の情報パケットを符号化し，十分大きな数の送信パケットを作成
[受信者] 任意の $k(1+\alpha)$ 個の送信パケットを受信し，復号
$\Rightarrow$ 受信パケットに関するオーバヘッド α を小さくしたい

$$k$$

情報パケット	s_1	s_2	$\cdots$	s_k				
送信パケット	x_1	x_2	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
受信者 A	x_1	x_2	$\cdots$	$\cdots$	$x_{\tilde{k}}$			
受信者 B	x_3	x_{10}	$\cdots$	$\cdots$	$x_{k'}$			
受信者 C	x_{1234}	x_{1236}	$\cdots$	$\cdots$	$x_{k''}$			

$$\tilde{k} = k(1+\alpha)$$

LT 符号 [Luby 2002] (1:概要と符号化)

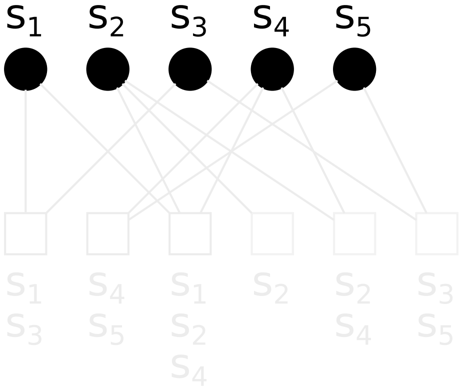
- 噴水符号のコンセプトを初めて実現した符号
- Peeling 復号法によって効率的に復号される (計算量 $\mathcal{O}(k)$)

LT 符号の符号化

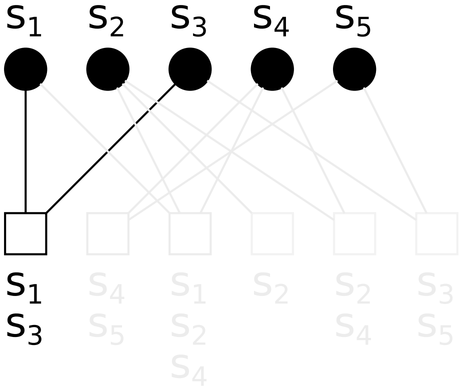
パラメタ：次数分布 $\Omega(x) = \sum_{i=1}^{d_m} \Omega_i x^i$
$(\sum_i \Omega_i = 1)$

1. 整数 d を確率 Ω_d で選ぶ
2. 相異なる d 個のパケットを選び出す
   $(\boldsymbol{s}_{j_1}, \boldsymbol{s}_{j_2}, \ldots, \boldsymbol{s}_{j_d})$
3. 選んだパケットを足し合わせる
   $\boldsymbol{x}_t = \boldsymbol{s}_{j_1} + \boldsymbol{s}_{j_2} + \cdots + \boldsymbol{s}_{j_d}$

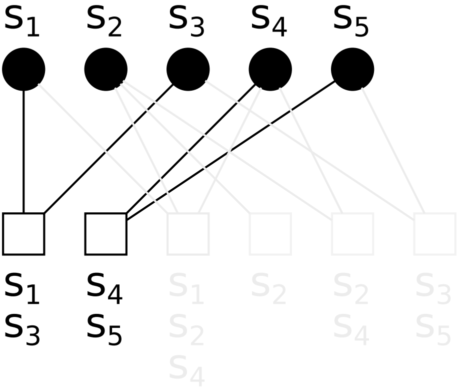
LT 符号 [Luby 2002] (2:符号化の例)



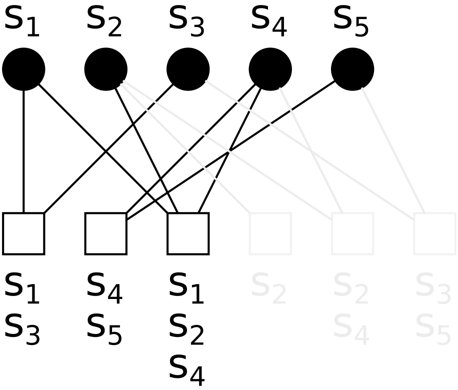
LT 符号 [Luby 2002] (2:符号化の例)



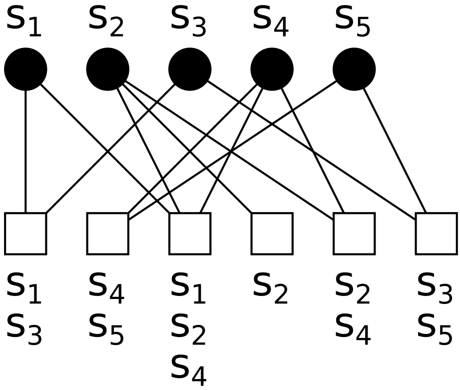
LT 符号 [Luby 2002] (2:符号化の例)



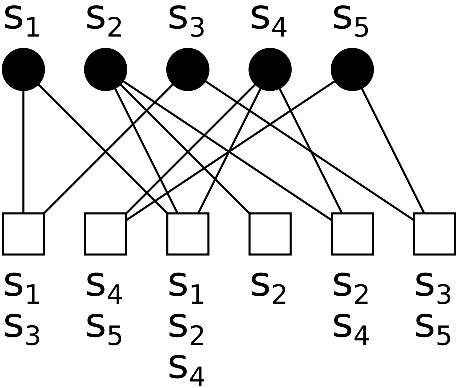
LT 符号 [Luby 2002] (2:符号化の例)



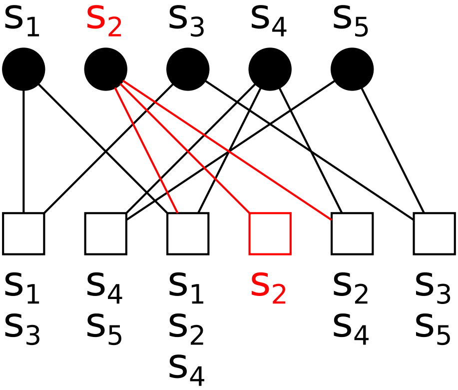
LT 符号 [Luby 2002] (2:符号化の例)



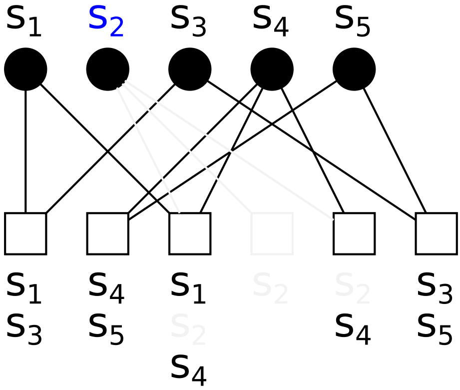
LT 符号 [Luby 2002] (3:復号の例)



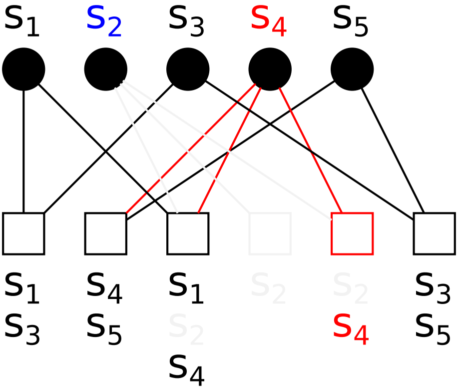
LT 符号 [Luby 2002] (3:復号の例)



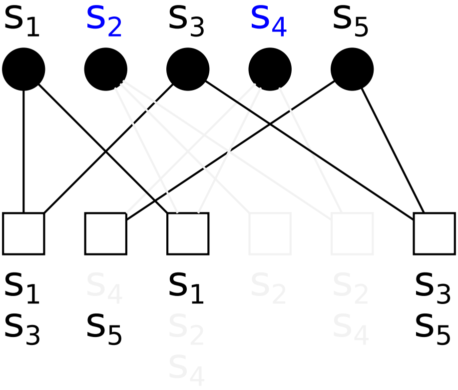
LT 符号 [Luby 2002] (3:復号の例)



LT 符号 [Luby 2002] (3:復号の例)



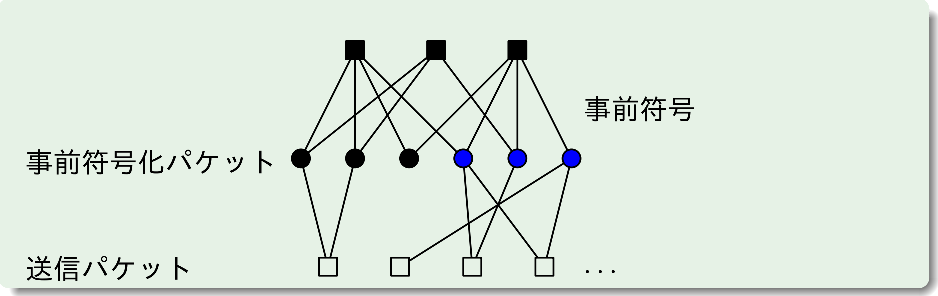
LT 符号 [Luby 2002] (3:復号の例)



Raptor 符号 [Shokrollahi 2006]

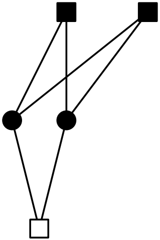
Raptor 符号 $(\mathcal{C}, \Omega(x))$

- 符号化
  1. 事前符号 $\mathcal{C}$ を用いて，情報パケットから事前符号化パケットを作成
  2. 事前符号化パケットを LT 符号 ($\Omega(x)$) で符号化し，送信
- 復号
  - 受信パケットと事前符号のタナーグラフをもとにファクターグラフを作成し，ピーリング復号



ピーリング復号の停止条件

次数 1 の検査ノードがなくなったときに復号停止
⇒ Stopping set にはまった時



研究のアイディア

Stopping set にはまっても復号できるような仕掛けを作る

ZD 噴水符号 (1: Zigzag decodable code [Sung 2013])

符号化

情報パケットに対して排他的論理和とシフト演算を用いる

情報パケット

$s_{1,1}$	$s_{1,2}$	$\cdots$	$s_{1,\ell}$
$s_{2,1}$	$s_{2,2}$	$\cdots$	$s_{2,\ell}$

$s_{1,1}$	$s_{1,2}$	$\cdots$	$s_{1,\ell}$	
	$s_{2,1}$	$\cdots$	$s_{2,\ell-1}$	$s_{2,\ell}$

送信パケット

$x_{1,1}$	$x_{1,2}$	$\cdots$	$x_{1,\ell}$

$x_{2,1}$	$x_{2,2}$	$\cdots$	$x_{2,\ell}$	$x_{2,\ell+1}$

復号法 (Zigzag 復号法)[Gollakota 2008]

- 符号化パケットの先頭から順に復号する.
- ビット毎のピーリング復号法とみなせる

1. $s_{1,1} = x_{2,1}$
2. $s_{2,1} = x_{1,1} - s_{1,1}$
3. $s_{1,2} = x_{2,2} - s_{2,1}$

ZD 噴水符号 (2: 符号化法)

パケットの表現法

$$
\begin{aligned}
\boldsymbol{a}_j &= (a_{j,1},\quad a_{j,2},\quad a_{j,3},\quad \ldots,\quad a_{j,\ell}) \\
a_j(z) &= a_{j,1} + a_{j,2}z + a_{j,3}z^2 + \cdots + a_{j,\ell}z^{\ell-1}
\end{aligned}
$$

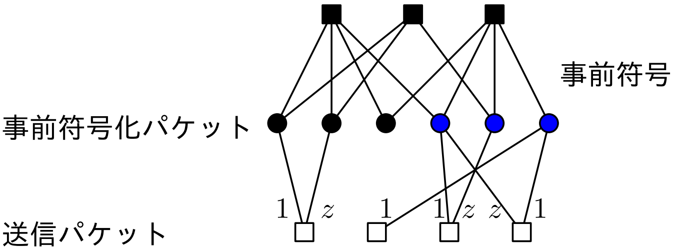
ZD 噴水符号 $(\mathcal{C}, \Omega(x), \Delta(x))$

1. 事前符号 $\mathcal{C}$ を用いて，情報パケット $(\boldsymbol{s}_1, \boldsymbol{s}_2, \ldots, \boldsymbol{s}_k)$ から事前符号化パケット $(\boldsymbol{a}_1, \boldsymbol{a}_2, \ldots, \boldsymbol{a}_n)$ を作成
2. (送信パケットの作成)
   1. 次数分布 $\Omega(x)$ に従って送信パケットの次数 d を選ぶ
   2. 相異なる d 個の事前符号化パケットを一様に選び，そのインデックスを $(j_1, \ldots, j_d)$ と書く
   3. シフト分布 $\Delta(x) := \sum_{i=0}^{s_m} \Delta_i x^i$ に基づき d 個のシフト量 $\delta_1, \ldots, \delta_d$ を独立に選ぶ.
   4. 以下を送信パケットとして送信

$$
\sum_{i=1}^{d} z^{\delta_i} a_{j_i}(z),
$$

ZD 噴水符号 (3: 復号法)

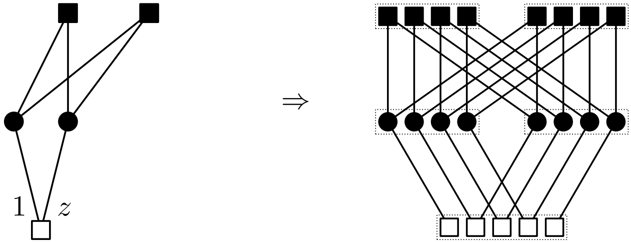
受信者毎に，事前符号と受信パケットからファクターグラフを作成



復号法

1. パケット毎に PA 復号 (次数 1 の検査ノードから復号)
2. ビット毎に ZD 復号

ZD 噴水符号 (4: 復号法の例)



性能評価 (1:オーバヘッド 1)

- 受信パケットに関するオーバヘッド α
  情報パケット数に対して余分に受け取った受信パケットの割合
  受信パケット数 $k(1+\alpha)$
- 受信ビットに関するオーバヘッド β
  情報ビット数に対して余分に受け取った受信ビット数の割合
  受信ビット数 $k\ell(1+\beta)$

Raptor 符号の場合：$\alpha = \beta$
提案符号の場合：$\alpha \leq \beta$

$$\beta = \alpha + \frac{1}{k\ell} \sum_{i=1}^{k(1+\alpha)} \ell_i.$$

但し，i 番目の受信パケットのビット数は $\ell + \ell_i$

性能評価 (2:オーバヘッド 2)

[補題 1] 受信パケットのビット長の期待値

- L : 受信パケットのビット長を表す確率変数
- $\Omega(x) = \sum_i \Omega_i x^i$: 次数分布
- $\Delta(x) = \sum_{i=0}^{s_m} \Delta_i x^i$: シフト分布

$$\mathbb{E}[L] = \ell + s_m - \sum_{i=0}^{s_m-1} \Omega(\Delta_{[0,i]}) - \sum_{i=1}^{s_m} \Omega(\Delta_{[i,s_m]})$$

但し，$\Delta_{[i,j]} := \sum_{k\in[i,j]} \Delta_k$

β の期待値は

$$\bar{\beta} = (1+\alpha)\frac{\mathbb{E}[L]}{\ell} - 1$$

性能評価 (3:復号誤り率の比較)

[定理 1] 復号誤り率の比較

$P_{\mathrm{B}}(\alpha, \mathcal{C}, \Omega, \Delta)$：オーバヘッド α のときの ZD 噴水符号 $(\mathcal{C}, \Omega(x), \Delta(x))$ の復号誤り率
任意の $\alpha, \mathcal{C}, \Omega(x), \Delta(x)$ に対して，

$$P_{\mathrm{B}}(\alpha, \mathcal{C}, \Omega, 1) \geq P_{\mathrm{B}}(\alpha, \mathcal{C}, \Omega, \Delta)$$

定理の意味
Raptor 符号より ZD 噴水符号の方が復号誤り率が低い

性能評価 (4:復号計算量)

- 空間計算量
  - Raptor 符号　$\ell(2n + \beta k)$
  - ZD 噴水符号 $\ell(2n + \beta k)$

  $\Rightarrow$ オーバヘッド β に依存
- 最大反復回数 (最悪時間計算量)
  - Raptor 符号　$\beta k + n$
  - ZD 噴水符号 $\ell(\beta k + n)$

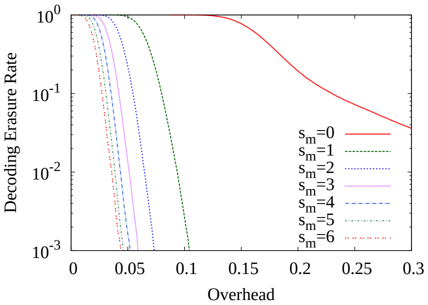
  $\Rightarrow$ ZD 噴水符号は復号時間が大きい

数値実験 (0:パラメタ設定)

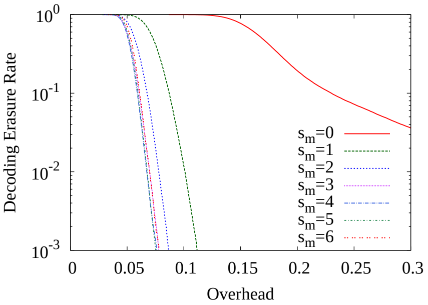
設定パラメタ

- パケット
  パケット長：100
- 事前符号
  (3,30)-正則 LDPC 符号
  情報パケット数 k： 900,1800,3600
  事前符号化パケット数 n：1000,2000,4000
- 送信パケット符号化
  次数分布：
  $\Omega(x) =$
  $0.007969x + 0.493570x^2 + 0.166220x^3 + 0.072646x^4 + 0.032558x^5 + 0.056058x^8 + 0.037229x^9 + 0.055590x^{19} + 0.025023x^{65} + 0.003135x^{66}$
  シフト分布： $\Delta(x) = \sum_{i=0}^{s_m} \frac{1}{1+s_m} x^i$

数値実験 (1:復号誤り率)



α



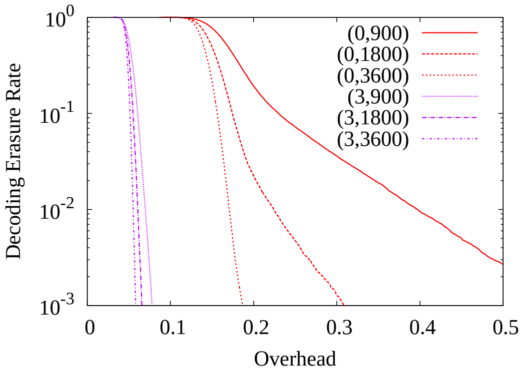
β

復号性能の比較 (赤：Raptor 符号, その他：提案法)

$s_m = 0, 1, 2, \ldots, 6,\ k = 900,\ n = 1000$

- ZD 噴水符号の方が復号誤り率が低い
- ZD 噴水符号の方が空間計算量が低い

数値実験 (2:復号誤り率)

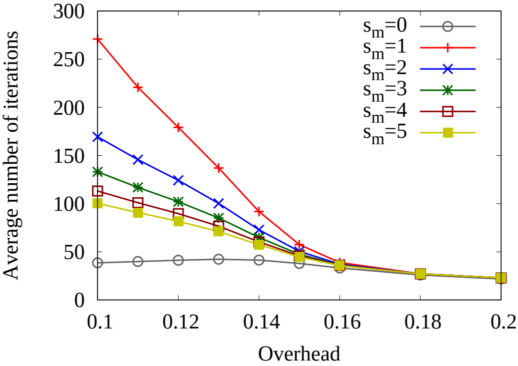


β

復号性能の比較 (赤：Raptor 符号, その他：提案法)
$s_m = 3$, $(k, n) = (900, 1000), (1800, 2000), (3600, 4000)$

しきい値 α^*, β^* が存在

数値実験 (3:復号反復回数)



α

反復回数の比較 (灰色：Raptor 符号，その他：提案符号)

$\ell = 100$, $(k, n) = (3600, 4000)$, $s_m = 0, 1, \ldots, 5$

欠点： 復号反復回数が大きい

漸近性能解析

α^*, β^* : 復号しきい値 ($k \to \infty$ のときの性能指標)

密度発展法による解析

$$\alpha^* = \min\left\{\alpha \mid \exists\tau\forall i \in [1,\ell]\ Q_i^{(\tau)} = 0\right\}.$$

$$\beta^* = (1+\alpha^*)\frac{\mathbb{E}[L]}{\ell} - 1$$

$$x_{1,i}^{(0)} = 1, \quad x_{2,i}^{(0)} = 1 \qquad \forall i \in \{1,\dots,\ell\}$$

$$x_{1,i}^{(\tau)} = \lambda(y_{1,i}^{(\tau)})I(y_{2,i}^{(\tau)}), \quad x_{2,i}^{(\tau)} = \Lambda(y_{1,i}^{(\tau)})\iota(y_{2,i}^{(\tau)})$$

$$y_{1,i}^{(\tau)} = 1 - \rho(1 - x_{1,i}^{(\tau-1)}), \quad y_{2,i}^{(\tau)} = 1 - \textstyle\sum_s \Delta_s \omega(1 - \hat{x}_{2,i+s}^{(\tau-1)}),$$

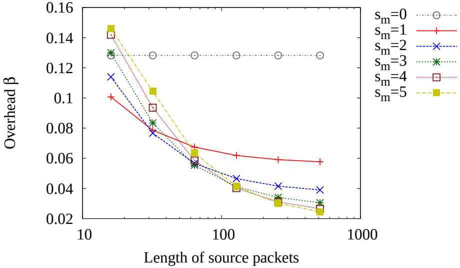
$$\hat{x}_{2,i}^{(\tau)} := \textstyle\sum_{s=0}^{s_m} \Delta_s x_{2,i-s}^{(\tau)}$$

$$Q_i^{(\tau)} = \Lambda(y_{1,i}^{(\tau)})I(y_{2,i}^{(\tau)}),$$

漸近性能解析の数値例 (1)

Table: Overhead α^* for $k \to \infty$

		s_m					
		0	1	2	3	4	5
ℓ	16	0.1282	0.0561	0.0338	0.0171	-0.0011	-0.0244
	32	0.1282	0.0563	0.0365	0.0265	0.0205	0.0156
	64	0.1282	0.0563	0.0365	0.0269	0.0219	0.0189
	128	0.1282	0.0563	0.0365	0.0269	0.0220	0.0190
	256	0.1282	0.0563	0.0365	0.0269	0.0220	0.0190



まとめと今後の課題

まとめ

- 噴水符号の既存研究 (LT 符号・Raptor 符号) を紹介した
- Zigzag decodable 噴水符号を紹介した

今後の課題

- 次数分布・シフト分布の最適化
  (次数分布の最適化：3 月の全国大会で発表予定)
- 復号法の反復回数の削減・効率化
  (ISITA2016 で提案したが，改善の余地あり)