June 23, 2015

On Fountain Codes Under ML decoding

Francisco Lázaro

Institute for Communications and Navigation
German Aerospace Center, DLR

Outline

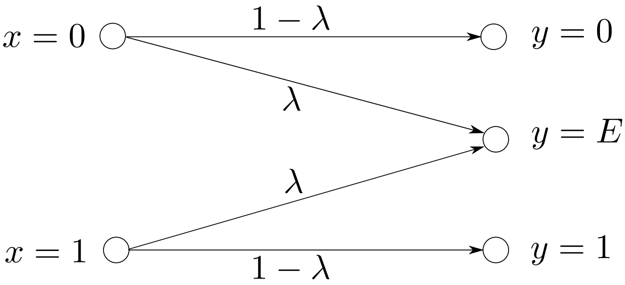
Outline

Introduction - Motivation

- Objective: transmit information from one sender to multiple receivers
- Challenges:
  - Different receivers suffer independent losses
  - Mechanisms based on retransmissions are known to be inefficient

Introduction - Channel Model

- Binary Erasure Channel (BEC)
  - $\Pr\{\text{Erasure}\} = \lambda$
  - Capacity $C = 1 - \lambda$



Introduction - Digital Fountain

- The encoder acts like a fountain
  - Each water drop is a packet
- Receiver: after receiving enough drops the glass if full and decoding is successful

[Byers98] Byers, John W., et al. *A digital fountain approach to reliable distribution of bulk data*, ACM SIGCOMM Computer Communication Review. Vol. 28. No. 4. ACM, 1998.

Introduction - Fountain codes

- Fountain Codes are rateless erasure codes
- Encoding
  - k input symbols
  - n output symbols, where $n = k, \ldots, \infty$
  - rate $r = \frac{k}{n}$
- Decoding
  - Decoding is possible when $m = k + \delta$ symbols are received
  - δ small

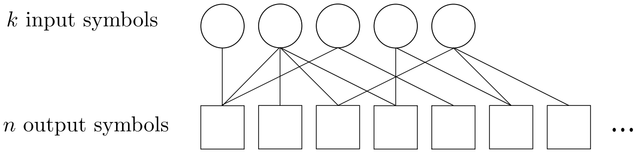
Outline

LT codes - Definition

- First class of practical fountain codes
- Defined by an output degree distribution Ω
  - $\Omega = \{\Omega_1, \Omega_2, \Omega_3, ..., \Omega_{d_{\max}}\}$
  - Ω_i = prob. of output degree i.
  - Ω is a probability mass function
- Encoding:
  - 1. Select output degree d according to Ω
  - 2. Select d input symbols and xor them to generate one output symbol

[Luby02] Luby, M., *LT codes*, Proc. of the 43rd Annual IEEE Symp. on Foundations of Computer Science, nov 2002.

LT codes - Bipartite Graph



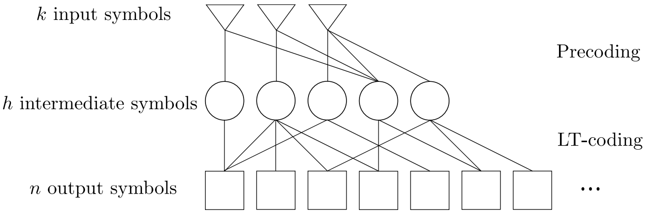
- Potentially an infinite amount of output symbols can be generated

LT codes - Properties

- LT codes were designed for iterative decoding.
- The average output degree $\bar{\Omega}$ needs to be $\mathcal{O}(\log(k))$
- Encoding / Decoding complexity is $\mathcal{O}(k \log(k))$
- Encoding / Decoding cost per symbol is $\mathcal{O}(\log(k))$

Raptor codes - Definition

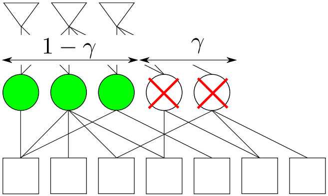
- Raptor codes are a serial concatenation of:
  - A precode as outer code
  - An LT code as inner code



[Shokrollahi06] Shokrollahi, M., *Raptor codes*, IEEE Transactions on Inf. Theory, jun 2006

Raptor codes - Introduction

- Why should this help?
  - *Light* LT code
    - *Constant* average output degree
    - recovers fraction $1 - \gamma$ of intermediate symbols
  - Pre-code
    - recovers all input symbols from this fraction $(1 - \gamma)$



Raptor codes - Properties

- Encoding cost is constant if a linear time encodable precode is used
- Decoding cost is constant if BP decoding is used
- Raptor codes are universally capacity achieving on the binary erasure channel with constant encoding / decoding cost

Raptor codes - Practice

- BP decoding requires large block lengths ($k \gg 10000$)
- In practice:
  - Due to memory limitations $k \sim 1000$ is used
  - ML decoding is used
  - Sometimes Raptor codes are used as fixed-rate codes

[3GPP-MBMS] 3GPP TS 26.346 V11.1.0: *Technical Specification Group Services and System Aspects; Multimedia Broadcast/Multicast Service; Protocols and Codecs* June 2012

Outline

ML decoding - Overview

- In the BEC channel ML decoding consists of decoding a system of linear equations
  - decoding complexity is $\mathcal{O}\left(k^3\right)$
  - decoding cost is $\mathcal{O}\left(k^2\right)$
- For moderate values of k ML decoding is feasible if:
  - An efficient ML algorithm is used (inactivation decoding).
  - The code design is tailored to the decoding algorithm
  - k is not too large.

Outline

1. Introduction
2. LT & Raptor Codes
3. ML decoding of Fountain Codes
   - LT codes
   - Raptor Codes
4. Distance properties
5. Conclusions

ML Decoding - LT Codes
Overview

- Maximum Likelihood (ML) decoding of LT codes consists of solving:

$$\mathbf{c} = \mathbf{u}\mathbf{G}^T$$

  - $\mathbf{u} = (u_1, u_2, ..., u_k)$ are the source symbols
  - $\mathbf{c} = (c_1, c_2, ..., c_m)$ are the received symbols
  - $\mathbf{G}$ is a $m \times k$ binary matrix
- Inactivation decoding is an *efficient* algorithm for ML

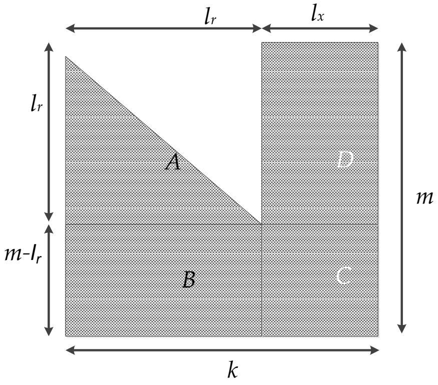
ML Decoding - LT Codes

Inactivation Decoding - Steps

1. *Triangularization*.
2. *Zero matrix procedure*.
3. *Gaussian Elimination*.
4. *Back-substitution*.

ML Decoding - LT Codes

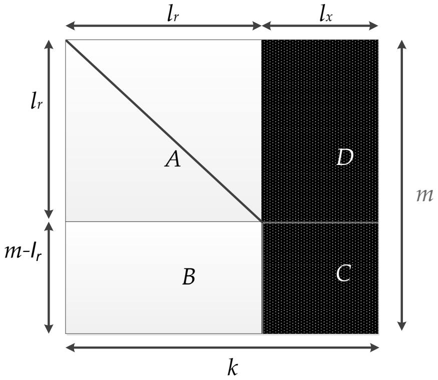
Inactivation Decoding - Triangularization



- Put **G** in approximate lower triangular form
- Column and row permutations
- All matrices are sparse

ML Decoding - LT Codes

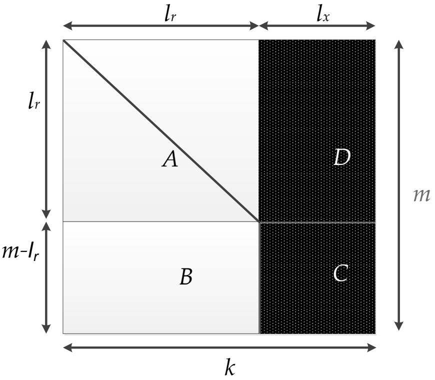
Inactivation Decoding - Zero matrix proc.



- Put **A** i diagonal form.
- Zero out **B**.
- Matrices **C** and **D** become dense.

ML Decoding - LT Codes

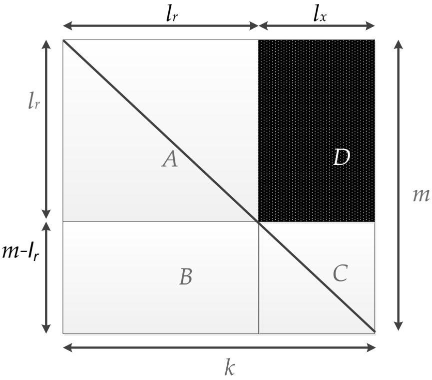
Inactivation Decoding - Gaussian Elimination



- Solve system of equations $\tilde{\mathbf{c}} = \tilde{\mathbf{u}}\mathbf{C}^T$.
- Complexity $\mathcal{O}({l_x}^3)$
- This step drives the decoding complexity.

ML Decoding - LT Codes

Inactivation Decoding - Backsubstitution



- Zero out matrix **D**

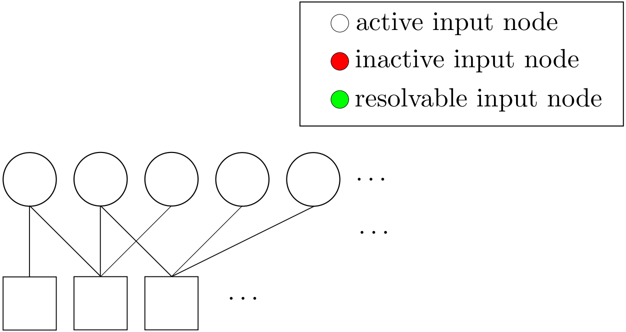
ML Decoding - LT Codes

Inactivation Decoding - Triangularization I

- Build a bipartite graph with input & output symbols
- Mark all input nodes as active
- Iterative algorithm:
  - Search for *active degree* 1 output symbol node
  - If it exists:
    - Mark its only neighbor as *resolvable*
  - If it does not:
    - Mark one input symbol as *inactive*
  - move to next step

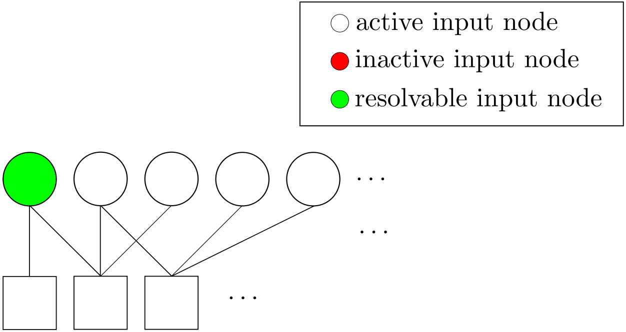
ML Decoding - LT Codes

Inactivation Decoding - Triangularization II



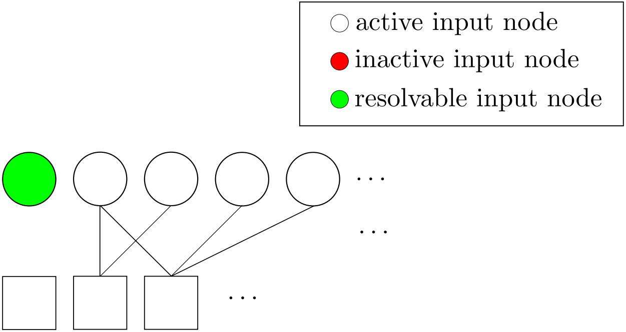
ML Decoding - LT Codes

Inactivation Decoding - Triangularization II



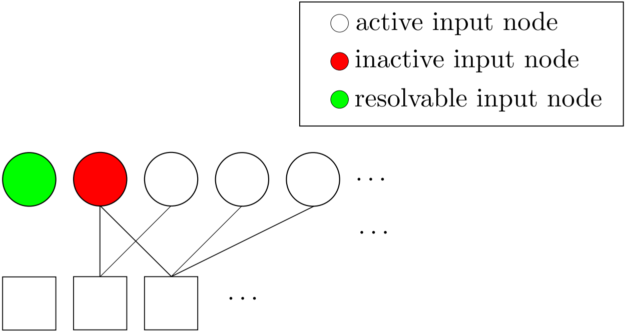
ML Decoding - LT Codes

Inactivation Decoding - Triangularization II



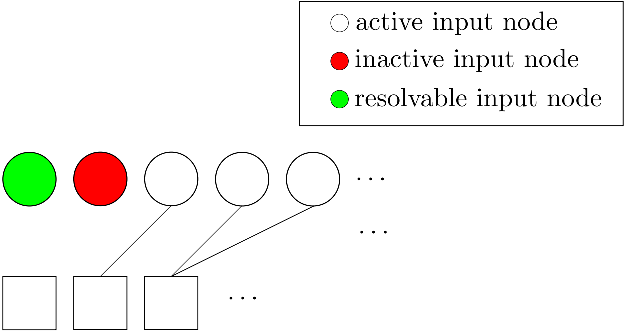
ML Decoding - LT Codes

Inactivation Decoding - Triangularization II



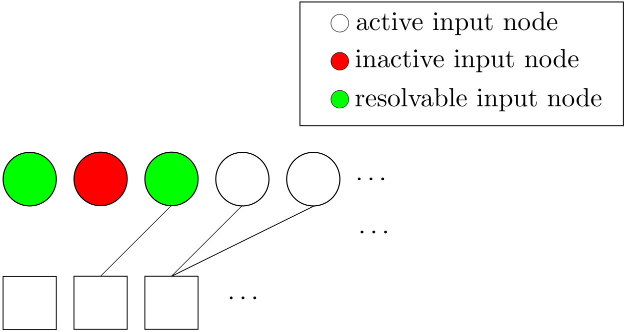
ML Decoding - LT Codes

Inactivation Decoding - Triangularization II



ML Decoding - LT Codes

Inactivation Decoding - Triangularization II



ML Decoding - LT Codes

Inactivation Decoding - Remarks

- Complexity driven by GE
- Triangularization is the critical step:
  - Determines the size system to be solved by GE
- Many possible inactivation techniques
  - We use *random inactivation*
    - Inactivate one input symbol uniformly at random
    - Simple to analyze

ML Decoding - LT Codes

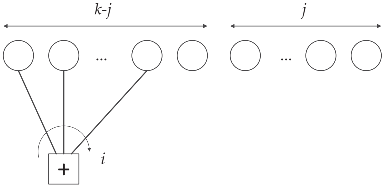
Inactivation Decoding - Model

- Objective:
  - Estimate the expected number of inactivations needed to decode.
- Method:
  - Track the node degree distributions of the output symbols in the reduced graph considering only *active input* symbols.
  - Changes at every decoding step

ML Decoding - LT Codes

Inactivation Decoding - Model - Step *j*

- $R_i^{(j)}$ number of output symbol nodes of active degree *i* at step *j*



- *j* input symbols are resolvable or inactive.
- $k - j$ input symbols are *active*

ML Decoding - LT Codes

Inactivation Decoding - Model - Assumptions & Initialization

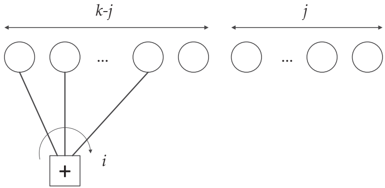
- Assumptions:
  - $R_i^{(j)}$ has a binomial distribution $\mathcal{B}(m^{(j)}, p_i^{(j)})$
    - $m^{(j)}$ number of output symbols in the graph at step j
    - $p_i^{(j)}$ probability that one of the output symbols at step j has active degree i
- Initialization:
  - $R_i^{(0)}$ follows a binomial distribution $\mathcal{B}(m, \Omega_i)$
  - $m = k * (1 + \epsilon)$
  - ϵ relative receiver overhead

ML Decoding - LT Codes

Inactivation Decoding - Model - Udate rule - I

- Assume a symbol has active degree i, $i \geq 2$
- What is the probability that its degree gets reduced?

$$\chi_i^{j+1} = \frac{i}{k-j}.$$

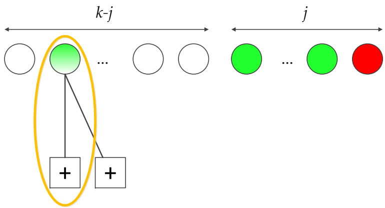


ML Decoding - LT Codes

Inactivation Decoding - Model - Update rule - II

- What happens to degree $i = 1$ output symbols?
  - if $R_1^j \geq 1$, no inactivation
    - Expected number of degree 1 output symbols leaving:

$$N_1^{(j+1)} = \mathrm{E}\left[1 + (R_1^j - 1)\frac{1}{k-j}\right]$$



ML Decoding - LT Codes

Inactivation Decoding - Model - Update rule - III

- What happens to degree $i = 1$ output symbols?
  - if $R_1^j < 1$, inactivation
    - Expected number of degree 1 output symbols that leave:

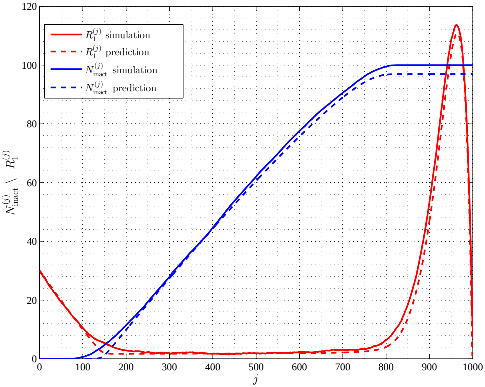
$$N_1^{(j+1)} = 0$$

- Probability of an inactivation happening:

$$\Pr\{R_1^{(j+1)} = 0\}$$

ML Decoding - LT Codes

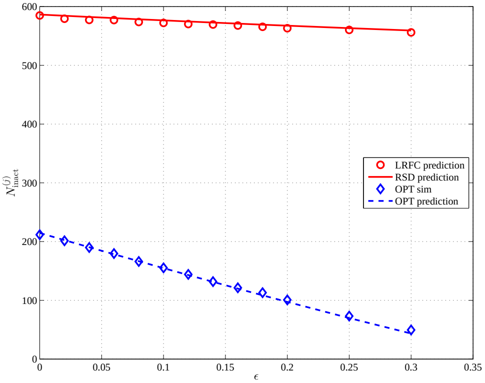
Inactivation Decoding - Model - Example I



- RSD, $\bar{\Omega} = 12$, $k = 1000$, $\epsilon = 0.2$

ML Decoding - LT Codes

Inactivation Decoding - Model- Example II



- LRFC & RSD, $\bar{\Omega} = 12$, $k = 1000$

Outline

ML Decoding - Raptor Codes
Basics

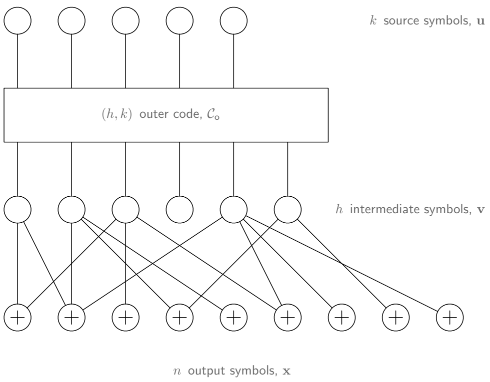
- Rate-r_o outer code with parity-check matrix $\mathbf{H}_o$

$$\mathbf{v}\mathbf{H}_o^T = \mathbf{0}$$

- Inner LT code with generator matrix $\mathbf{G}_i$

$$\mathbf{v}\mathbf{G}_i = \mathbf{x}$$

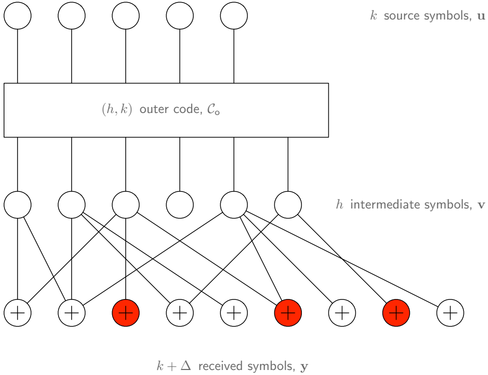
- The output symbol degrees $\sim \{\Omega_i\}$



ML Decoding - Raptor Codes
Basics

- Suppose $k + \Delta$ symbols are received (not erased)
- $\mathbf{y}$:= received vector
- New set of constraints

$$\mathbf{v}\left[\mathbf{H}_o^T \mid \bar{\mathbf{G}}_i\right] = [\mathbf{0} \mid \mathbf{y}]$$

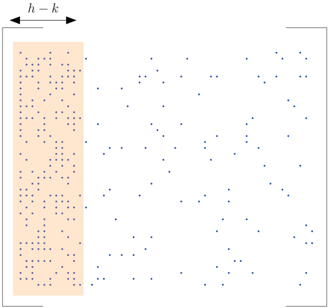


ML Decoding - Raptor Codes

Basics

- Inactivation decoding is used to solve the system of equations.

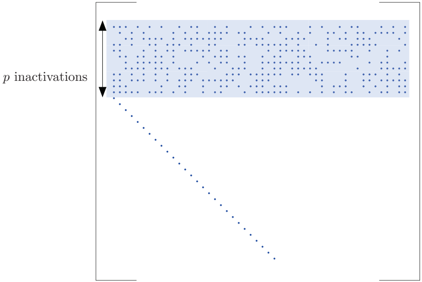
$$\left[\mathbf{H}_{\mathrm{o}}^{T} \mid \bar{\mathbf{G}}_{\mathrm{i}}\right] =$$



ML Decoding - Raptor Codes

Basics

- Inactivated rows are resolved using Gaussian Elimination



ML Decoding - Raptor Codes

Basics

- The same method used for LT codes can be used to estimate the number of inactivations.
- Experimentally demonstrated that the number of inactivations. is proportional to $h - k$
- Outer code rate, r_o,shall be kept as large as possible
- How high can r_o be so that decoding succeeds with probability close to one?

ML Decoding - Raptor Codes

Basics

- The same method used for LT codes can be used to estimate the number of inactivations.
- Experimentally demonstrated that the number of inactivations. is proportional to $h - k$
- Outer code rate, r_o,shall be kept as large as possible
- How high can r_o be so that decoding succeeds with probability close to one?

ML Decoding - Raptor Codes

Basics

- The same method used for LT codes can be used to estimate the number of inactivations.
- Experimentally demonstrated that the number of inactivations. is proportional to $h - k$
- Outer code rate, r_o,shall be kept as large as possible
- How high can r_o be so that decoding succeeds with probability close to one?

DLR

Outline

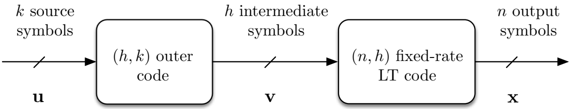
Distance Properties

- It is difficult to analyze the decoding error probability of Raptor codes in a rateless setting.
- However, in a fixed-rate setting one can use standard coding theory tools

Distance Properties

Fixed-Rate Setting

- With respect to fountain codes, we simply stop the encoder after generating n output symbols

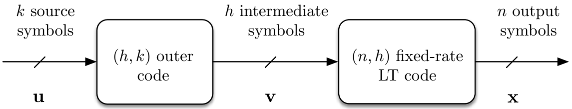


- $w := w_{\mathrm{H}}(\mathbf{u})$ input Hamming weight
- $l := w_{\mathrm{H}}(\mathbf{v})$ intermediate Hamming weight
- $d := w_{\mathrm{H}}(\mathbf{x})$ output Hamming weight

Distance Properties

Fixed-Rate Setting

- With respect to fountain codes, we simply stop the encoder after generating n output symbols

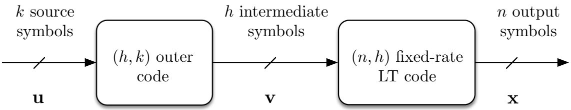


- Sometimes, we use normalized weights
- $\lambda := l/h$ *normalized* intermediate Hamming weight
- $\delta := d/n$ *normalized* output Hamming weight

Distance Properties

Fixed-Rate Setting

- With respect to fountain codes, we simply stop the encoder after generating *n* output symbols



- $\mathrm{r_o} := k/h$ outer code rate
- $\mathrm{r_i} := h/n$ inner code rate
- $r := \mathrm{r_i r_o}$ Raptor code rate

Distance Properties
Fixed-Rate Setting

- Under ML decoding, block error probability can be upper bounded by

$$P_B \leq \sum_d A_d \epsilon^d \qquad \text{(union bound)}$$

  - A_d is the multiplicity of codewords with weight d
  - ϵ the channel erasure probability
- We study the average weight enumerator $\{A_d\}$ for fixed rate Raptor code ensembles $\mathscr{C}$ where

  (i) Precode from the (h, k) binary linear random ensemble $\mathscr{C}_o$

  (ii) LT code with output degree distribution $\Omega = \{\Omega_1, \Omega_2, ..., \Omega_{d_{\max}}\}$

Distance Properties
Weight Enumerator

Theorem

Let A_d be the expected multiplicity of codewords of weight d for a code picked randomly in the ensemble $\mathscr{C}(\mathscr{C}_\mathrm{o}, \Omega, \mathrm{r_i}, \mathrm{r_o}, n)$.

$$A_d = \binom{n}{d} 2^{-h(1-\mathrm{r_o})} \sum_{l=1}^{h} \binom{h}{l} p_l^d (1-p_l)^{n-d}, \qquad d \geq 1. \tag{1}$$

where

$$p_l = \sum_{j=1}^{\mathrm{d_{max}}} \Omega_j \sum_{\substack{i=\max(1,l+j-h)\\ i \text{ odd}}}^{\min(l,j)} \frac{\binom{j}{i}\binom{h-j}{l-i}}{\binom{h}{l}}. \tag{2}$$

Distance Properties

Growth Rate - Motivation

- Often, ensembles distance properties can be captured in a compact form by letting $n \to \infty$, keeping $\mathrm{r_i}$ and $\mathrm{r_o}$ constant.
- In fact, for large n the weight distribution can be approximated via

$$A_{n\delta} \approx 2^{nG(\delta)}$$

where $G(\delta)$ is referred to as growth rate of the ensemble,

$$\mathrm{G}(\delta) := \lim_{n \to \infty} \frac{1}{n} \log_2 A_{\delta n}$$

- We thus aim at developing simple expressions for $\mathrm{G}(\delta)$

Distance Properties

Growth Rate

Theorem

The growth rate of the fixed-rate Raptor code ensemble weight distribution is

$$\mathrm{G}(\delta) = \mathrm{H_b}(\delta) - \mathrm{r_i}(1 - \mathrm{r_o}) + \mathrm{f_{max}}(\delta).$$

where

$$\mathrm{f_{max}}(\delta) := \max_{\lambda} \mathrm{f}(\delta, \lambda)$$

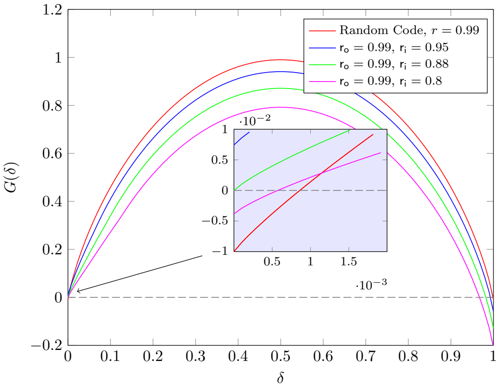
and

$$\mathrm{f}(\delta, \lambda) := \mathrm{r_i}\mathrm{H_b}(\lambda) + \delta \log_2 p_\lambda + (1 - \delta) \log_2 (1 - p_\lambda) .$$

Distance Properties

Growth Rate - Example

- Ensemble $\mathscr{C}_\infty(\mathscr{C}_\mathrm{o}, \Omega, \mathrm{r_i}, \mathrm{r_o} = 0.99)$ for $\mathrm{r_i} = 0.95, 0.88$ and 0.8



Degree	Ω
1	0.0098
2	0.4590
3	0.2110
4	0.1134
10	0.1113
11	0.0799
40	0.0156
$\bar{\Omega}$	4.6314

Distance Properties

Typical Minimum Distance

- The real number

$$\delta^{\star} := \inf\{\delta > 0 : G(\delta) > 0\}$$

  is the typical minimum distance of the ensemble
- It can be proved that the expected minimum distance the Raptor codes of the ensemble is $d_{\min} = \delta^{\star} n$

Distance Properties

Positive Typical Minimum Distance

Positive typical minimum distance region

We define the *positive* typical minimum distance region of a Raptor code ensemble as the set $\mathscr{P}$ of code rate pairs $(\mathsf{r_i}, \mathsf{r_o})$ for which the ensemble possesses a positive typical minimum distance.

Theorem

The positive typical minimum distance region is given by

$$\mathscr{P} = \{(\mathsf{r_i}, \mathsf{r_o}) \,|\, \mathsf{r_i}(1 - \mathsf{r_o}) > \mathsf{f}^*_{\max}(\mathsf{r_i})\}$$

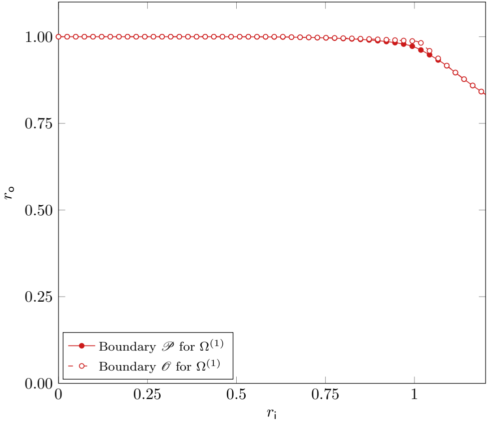
where

$$\mathsf{f}^*_{\max}(\mathsf{r_i}) := \lim_{\delta \to 0^+} \mathsf{f}_{\max}(\delta).$$

DLR

Distance Properties

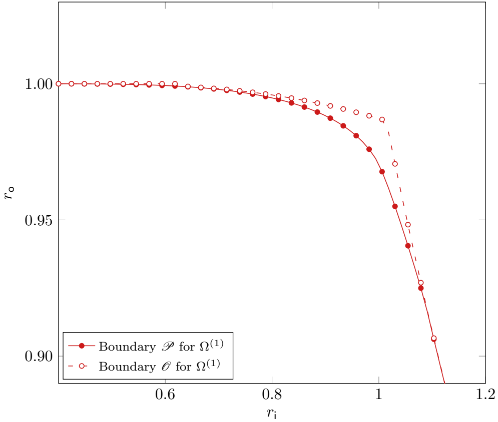
Rate Region Example, distribution from [3GPP-MBMS]



Degree	Ω
1	0.0098
2	0.4590
3	0.2110
4	0.1134
10	0.1113
11	0.0799
40	0.0156
$\bar{\Omega}$	4.6314

Distance Properties

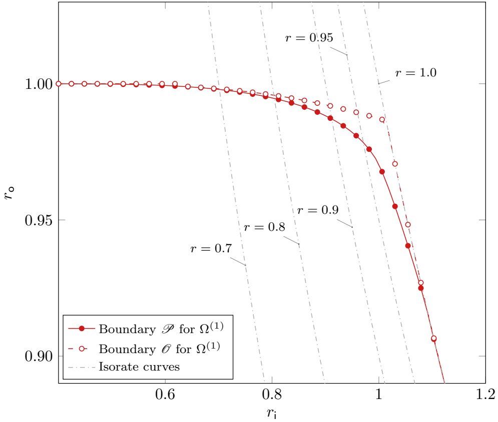
Rate Region Example, distribution from [3GPP-MBMS]



Degree	Ω
1	0.0098
2	0.4590
3	0.2110
4	0.1134
10	0.1113
11	0.0799
40	0.0156
$\bar{\Omega}$	4.6314

Distance Properties

Rate Region Example, distribution from [3GPP-MBMS]



Degree	Ω
1	0.0098
2	0.4590
3	0.2110
4	0.1134
10	0.1113
11	0.0799
40	0.0156
$\bar{\Omega}$	4.6314

Outline

1. Introduction
2. LT & Raptor Codes
3. ML decoding of Fountain Codes
4. Distance properties
5. Conclusions

Conclusions I

A. The decoding complexity of Raptor and LT codes under ML has been analyzed

B. We have a characterization of the distance properties of Raptor ensembles with fixed rate and random outer codes

C. We can analyze the complexity - decoding error probability trade-off of Raptor codes.

D. Many open points: Extension to arbitrary outer codes, stronger results on minimum distance (e.g., via expurgated ensembles), non-binary Raptor codes, calculation of thresholds under ML decoding, exact finite length analysis of inactivation decoding.

Conclusions I

A. The decoding complexity of Raptor and LT codes under ML has been analyzed

B. We have a characterization of the distance properties of Raptor ensembles with fixed rate and random outer codes

C. We can analyze the complexity - decoding error probability trade-off of Raptor codes.

D. Many open points: Extension to arbitrary outer codes, stronger results on minimum distance (e.g., via expurgated ensembles), non-binary Raptor codes, calculation of thresholds under ML decoding, exact finite length analysis of inactivation decoding.

Conclusions I

A. The decoding complexity of Raptor and LT codes under ML has been analyzed
B. We have a characterization of the distance properties of Raptor ensembles with fixed rate and random outer codes
C. We can analyze the complexity - decoding error probability trade-off of Raptor codes.
D. Many open points: Extension to arbitrary outer codes, stronger results on minimum distance (e.g., via expurgated ensembles), non-binary Raptor codes, calculation of thresholds under ML decoding, exact finite length analysis of inactivation decoding.

Conclusions I

A. The decoding complexity of Raptor and LT codes under ML has been analyzed
B. We have a characterization of the distance properties of Raptor ensembles with fixed rate and random outer codes
C. We can analyze the complexity - decoding error probability trade-off of Raptor codes.
D. Many open points: Extension to arbitrary outer codes, stronger results on minimum distance (e.g., via expurgated ensembles), non-binary Raptor codes, calculation of thresholds under ML decoding, exact finite length analysis of inactivation decoding.

THANKS!

The results presented can be found in:

[Lazaro14] F. Lázaro, G. Liva, G. Bauch, *LT Code Design for Inactivation Decoding*, IEEE Information Theory Workshop 2014, Hobart, Tasmania, Australia.

[Lazaro15-1] F. Lázaro, E. Paolini, G. Liva, G. Bauch, *On The Weight Distribution of Fixed-Rate Raptor Codes*, IEEE International 2015 Symposium on Information Theory, Hong Kong, China.

[Lazaro15-02] F. Lázaro, E. Paolini, G. Liva, G. Bauch, *Distance Spectrum of Fixed-Rate Raptor Codes with Linear Random Precoders*, submitted to IEEE Journal on Selected Areas in Communications.