Rate-Compatible Non-Binary LDPC Codes
Concatenated with Multiplicative Repetition Codes

Kenta KASAI
David DECLERCQ
Charly POULLIAT
Kohichi SAKANIWA

Tokyo Institute of Technology, ETIS ENSEA/UCP/CNRS

June 15, 2010
Motivation

- **Goal:** Construct good rate-compatible LDPC codes which cover from low rate to high rate.

- **Strategy:** Produce lower-rate and higher-rate codes from a single good mother code of moderate rate.
Difficulty of Designing Low-Rate LDPC Codes

- Capacity-approaching irregular LDPC code is available \( n \to \infty \).

- Designing short and moderate low-rate LDPC codes is difficult. “One encounters a difficulty when designing very low rate LDPC codes in the standard irregular framework, it is relatively difficult to achieve high thresholds.”

  from Multi-Edge type LDPC codes, 2003.

- Structured low-rate LDPC codes have good thresholds. However they tend to have nodes of high degree.

- The optimized structured low-rate LDPC codes need to be used with large code length to exploit the potential decoding performance.
Low-rate LDPC codes tend to have many check nodes. In fact, LDPC codes with $K$ information nodes and rate $R$ have $M = N(1 - R) = K \left( \frac{1 - R}{R} \right)$ check nodes.

Example: $M = 9K$ for $R = 0.1$ while $M = K/9$ for $R = 0.9$.

Check nodes require complex computations.

We solve these issues of designing and decoding low-rate LDPC codes by a very simple scheme.
Non-Binary LDPC Codes

- Non-binary LDPC codes defined over Galois field $\text{GF}(2^m)$.

\[ \{ x \in \text{GF}(2^m)^N \mid Hx = 0 \} \quad H : M \times N \text{ sparse matrix over } \text{GF}(2^m) \]

- Irregular graph does not help improving thresholds with $\text{GF}(\geq 2^6)$.

- The $(2, d_c \geq 3)$-regular LDPC codes over $\text{GF}(2^8)$ are empirically known as the best efficiently decodable error correcting codes for short code length $K \leq 1000$.

- Efficiently decodable by the FFT-based sum-product decoding.
Designing Low-Rate Non-Binary LDPC Codes

We use \((2, 3)\)-regular non-binary LDPC over \(\text{GF}(2^8)\) as a mother code.

Denote the codeword by \((x_1, \ldots, x_N) \in \text{GF}(2^8)^N\), \(R = 1/3\).

Conventional rate-lowering scheme:

- Klinic et al. use shortening for reducing the rate.
  \[(x_1, 0, x_3, 0, \ldots, x_N), \quad R < 1/3.\]

- Repetition reduces the rate.
  \[(x_1, \ldots, x_N, x_1, \ldots, x_N), \quad R = 1/6.\]

Repetition is the worst thing to do for coding, because repetition just uses more \(N\) channels without improving \(E_b/N_0\) vs BER curve.
Multiplicatively Repeated Non-Binary LDPC Codes

- Instead of repetition, we propose “Multiplicative Repetition.”

\[(x_1, \ldots, x_N, h_1 x_1, \ldots, h_N x_N), \quad R = 1/6,\]

where \(h_1, \ldots, h_N\) are randomly chosen from \(\text{GF}(2^8) \setminus \{0\}\).

- Multiplicative repetition twice gives

\[(x_1, x_2, \ldots, x_N, h_1 x_1, \ldots, h_N x_N, h'_1 x_1, \ldots, h'_N x_N), \quad R = 1/9.\]

- Multiplicative repetition three times gives

\[(x_1, x_2, \ldots, x_N, h_1 x_1, \ldots, h_N x_N, h'_1 x_1, \ldots, h'_N x_N, h''_1 x_1, \ldots, h''_N x_N), \quad R = 1/12.\]
Numerical Results ($K=1024$ bits, BIAWGNC)

![Graph showing block error rate vs. Eb/No in dB for different codes.

- Punctured $C_1$ R=1/2
- $C_1$ R=1/3
- $C_2$ R=1/6
- MET R=1/2
- MET R=1/6
- ARA R=1/6]
Numerical Results ($K=192$ bits, BIAWGNC)

![Graph showing Block Error Rate vs. Eb/No (dB) for different codes and puncturing rates.](image-url)
Decoding Algorithm

Tanner graph of the mother code. A (2,3)-regular non-binary LDPC code over GF($2^8$). $R = 1/3$. 
$R = 1/6.$

Decoder uses only Tanner graph of the mother code.
$$R = 1/9.$$ Decoder uses only Tanner graph of the mother code.
Multiplicative Repetition vs Shortening over BEC

![Graph showing multiplicative repetition vs shortening over BEC]

- **PROPOSED, \( T=1, \ldots, 10 \)**
- **Shortened code**

**Norm. gap to capacity** \( \frac{(1-e^{-R})-R}{R} \) vs **coding rate** \( R \)
Summary and on-going work

Summary

1. We proposed multiplicatively repeated LDPC codes.
2. Decoding complexity does not depend on the rate.
3. Outperforms best-so-far codes at short code length.

On-going works

1. Slepian-Wolf coding
2. Fountain coding
3. Convolutional coding