Low-Density Parity-Check Codes for Symmetric and Non-Symmetric Channels

Chih-Chun Wang

chihw@princeton.edu

Department of Electrical Engineering, Princeton University
Content

- **Part I** — Low-density parity-check (LDPC) codes on symmetric & non-symmetric memoryless channels:

- **Part II** — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes.
Content

Part I — Low-density parity-check (LDPC) codes on symmetric & non-symmetric memoryless channels:
  - Progress for symmetric channels: Density Evolution
  - Why non-symmetric channels?

Part II — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes.
Part I — Low-density parity-check (LDPC) codes on symmetric & non-symmetric memoryless channels:

- Progress for symmetric channels: Density Evolution
- Why non-symmetric channels?
- The codeword-dependent error resiliency

Part II — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes.
Part I — Low-density parity-check (LDPC) codes on symmetric & non-symmetric memoryless channels:

- Progress for symmetric channels: Density Evolution
- Why non-symmetric channels?
- The codeword-dependent error resiliency
- Codeword averaging & the perfect projection condition

Part II — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes.
Part I — Low-density parity-check (LDPC) codes on symmetric & non-symmetric memoryless channels:

- Progress for symmetric channels: Density Evolution
- Why non-symmetric channels?
- The codeword-dependent error resiliency
- Codeword averaging & the perfect projection condition
- A generalized DE

Part II — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes.
Content

**Part I** — Low-density parity-check (LDPC) codes on symmetric & non-symmetric memoryless channels:
- Progress for symmetric channels: **Density Evolution**
- Why non-symmetric channels?
- The codeword-dependent error resiliency
- Codeword averaging & **the perfect projection condition**
- A generalized DE
- The **typicality** of linear LDPC codes among the coset code ensemble

**Part II** — EXtrinsic Information (EXIT) chart & Finite-dim. bounds for LDPC codes.
Part I — Low-density parity-check (LDPC) codes on symmetric & non-symmetric memoryless channels:

- Progress for symmetric channels: Density Evolution
- Why non-symmetric channels?
- The codeword-dependent error resiliency
- Codeword averaging & the perfect projection condition
  - A generalized DE
  - The typicality of linear LDPC codes among the coset code ensemble
  - The local optimality of the belief propagation decoder

Part II — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes.
Error Correcting Codes

Channel: $P(dy|x)$

$m \in \{0,1\}^{nR}$

$\begin{align*}
\text{ENC} \quad & \quad \text{MOD} \quad & \quad \text{EM Wave/Laser Pulses} \quad & \quad \text{DEMOD} \quad & \quad \text{DEC} \\
\end{align*}$

$x \in \{0,1\}^n$

$y \in Y^n$

$\hat{m} \in \{0,1\}^{nR}$
Error Correcting Codes

Channel: $P(dy|x)$

- Memoryless channels: $P(dy|x) = \prod_{i=1}^{n} P(dy_i|x_i)$.
- Symmetric channels: $P(y|x = 0) = P(-y|x = 1)$. 

$m \in \{0, 1\}^{nR} \xrightarrow{} x \in \{0, 1\}^{n} \xrightarrow{}$ MOD $\xrightarrow{}$ EM Wave/Laser Pulses $\xrightarrow{}$ DEMOD $\xrightarrow{} y \in Y^n \xrightarrow{}$ DEC $\xrightarrow{} \hat{m} \in \{0, 1\}^{nR}$
Error Correcting Codes

Channel: $P(dy|x)$

- Memoryless channels: $P(dy|x) = \prod_{i=1}^{n} P(dy_i|x_i)$.
- Symmetric channels: $P(y|x = 0) = P(-y|x = 1)$.
- Shannon’s channel coding theorem: $R < C := \max_{P_X} I(X;Y)$. 
Error Correcting Codes

- **Channel:** \( P(dy|x) \)

- Memoryless channels: \( P(dy|x) = \prod_{i=1}^{n} P(dy_i|x_i) \).
- Symmetric channels: \( P(y|x=0) = P(-y|x=1) \).
- Shannon’s channel coding theorem: \( R < C := \max_{P_X} I(X;Y) \).
- **Memoryless symmetric channels:** Capacity-approaching error correcting codes have been constructed, including turbo codes, low-density parity-check (LDPC) codes, irregular RA codes, LT codes, concatenated tree codes, etc.
- Well established analysis tools.
- Performance: \( 0.1 \sim 1.5 \text{dB away from capacity} \).
Ultra high performance on almost all symmetric channels.
Ultra high performance on almost all symmetric channels.

**Question:** How about non-symmetric memoryless channels?
Ultra high performance on almost all symmetric channels. 

**Question:** How about non-symmetric memoryless channels?

Examples:
Z-Channels

```
1 0 1 1 1 1 0
```

Chih-Chun Wang – p.4/41
Ultra high performance on almost all symmetric channels. **Question:** How about non-symmetric memoryless channels?

Examples:

Z-Channels

```
1 0 1 1 1 1 0
```

Chih-Chun Wang – p.4/41
Ultra high performance on almost all symmetric channels. **Question:** How about non-symmetric memoryless channels?

Examples:

Z-Channels
Ultra high performance on almost all symmetric channels.

**Question:** How about non-symmetric memoryless channels?

Examples:

Z-Channels

\[
p_{1 \rightarrow 0} \quad 0 \\
1 - p_{1 \rightarrow 0} \\
1 \quad 0
\]
Ultra high performance on almost all symmetric channels.

**Question:** How about non-symmetric memoryless channels?

**Examples:**

- **Z-Channels**

  - $0 \xrightarrow{p_{1\to0}} 0$
  - $1 \xrightarrow{1 - p_{1\to0}} 1$

- **BICM**
LDPC Codes & BP Decoders

\[ H = \begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1
\end{pmatrix} \]

\[ Hx = 0 \]
LDPC Codes & BP Decoders

\[ H = \begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 \\
\end{pmatrix} \]

\[ Hx = 0 \]

\[ J = \{1, 2, 3, 4\} \]

\[ i = \{1, 2, 3, 4, 5, 6\} \]

\[ C^6(d_v, d_c), \quad d_v = 2, \quad d_c = 3 \]
LDPC Codes & BP Decoders

\[ H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \]

\[ Hx = 0 \]

Message Passing Algorithms

(i) \(m_0\),  
(ii) \(\Psi_v(m_0, m_1, \cdots, m_{d_v-1})\),  
(iii) \(\Psi_c(m_1, \cdots, m_{d_c-1})\)

\(C^6(d_v, d_c), d_v = 2, d_c = 3\)
LDPC Codes & BP Decoders

\[ H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \]

\[ Hx = 0 \]

Message Passing Algorithms

(i) \( m_0 \), (ii) \( \Psi_v(m_0, m_1, \ldots, m_{d_v-1}) \), (iii) \( \Psi_c(m_1, \ldots, m_{d_c-1}) \)

\( C^6(d_v, d_c), d_v = 2, d_c = 3 \)
LDPC Codes & BP Decoders

\[ H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \]

\[ Hx = 0 \]

Message Passing Algorithms

(i) \( m_0 \), (ii) \( \Psi_v(m_0, m_1, \cdots, m_{d_v-1}) \), (iii) \( \Psi_c(m_1, \cdots, m_{d_c-1}) \)

\( C^6(d_v, d_c), d_v = 2, d_c = 3 \)
LDPC Codes & BP Decoders

\[ H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \]

\[ Hx = 0 \]

Message Passing Algorithms

(i) \( m_0 \), (ii) \( \Psi_v(m_0, m_1, \cdots, m_{d_v-1}) \), (iii) \( \Psi_c(m_1, \cdots, m_{d_c-1}) \)
LDPC Codes & BP Decoders

\[
H = \begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 \\
\end{pmatrix}
\]

\[Hx = 0\]

Message Passing Algorithms

(i) \(m_0\), (ii) \(\Psi_v(m_0, m_1, \ldots, m_{d_v-1})\), (iii) \(\Psi_c(m_1, \ldots, m_{d_c-1})\)
LDPC Codes & BP Decoders

\[ H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \]

\[ Hx = 0 \]

Message Passing Algorithms

(i) \[ m_0 \], (ii) \[ \Psi_v(m_0, m_1, \cdots, m_{d_v-1}) \], (iii) \[ \Psi_c(m_1, \cdots, m_{d_c-1}) \]

\[ C^6(d_v, d_c), d_v = 2, d_c = 3 \]
LDPC Codes & BP Decoders

\[
H = \begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1
\end{pmatrix}
\]

\[
Hx = 0
\]

Belief Propagation Decoder:

(i) \(m_0\), (ii) \(\Psi_v(m_0, m_1, \cdots, m_{d_v-1})\), (iii) \(\Psi_c(m_1, \cdots, m_{d_c-1})\)

\(C^6(d_v, d_c), d_v = 2, d_c = 3\)
**LDPC Codes & BP Decoders**

\[ H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \]

\[ Hx = 0 \]

Belief Propagation Decoder:

(i) \( m_0 \),  (ii) \( \Psi_v(m_0, m_1, \ldots, m_{d_v-1}) \),  (iii) \( \Psi_c(m_1, \ldots, m_{d_c-1}) \)

<table>
<thead>
<tr>
<th>BP Decoder</th>
<th>Optimality</th>
<th>Analysis Tools</th>
<th>Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cycle-free + Non-Sym. Chs.</td>
<td>✓</td>
<td>✓</td>
<td>optimal</td>
</tr>
<tr>
<td>LDPC Codes + Sym. Chs.</td>
<td>?</td>
<td>✓</td>
<td>outstanding</td>
</tr>
<tr>
<td>LDPC Codes + Non-Sym. Chs.</td>
<td>?</td>
<td>?</td>
<td>very good</td>
</tr>
</tbody>
</table>
The Density Evolution

\[ i = j - 1 - i - 2 - i - 3 - i - 4 - i - 5 - i - 6 \]

\[ j = 1 \]

\[ i = 1 \]
The Density Evolution

\[ \text{The Density Evolution} \]

\[ i = 1 - i = 2 - i = 3 - i = 4 - i = 5 - i = 6 \]

\[ j = 1 - 2 - 3 - 4 \]

Chih-Chun Wang – p.6/41
The Density Evolution

\[ i = 1 - 2 - 3 - 4 - 5 - 6 \]

\[ j = 1 \]

Diagram showing the density evolution with nodes and arrows connecting them.
The Density Evolution

\[ i = 1 \]

\[ j = 1 \]

\[ 2 \quad 3 \quad 4 \]

\[ 5 \quad 6 \]

Chih-Chun Wang – p.6/41
The Density Evolution

\[ i = 1 - i_1 - i_2 - i_3 - i_4 - i_5 - i_6 \]

\[ j = 1, 2, 3, 4 \]

Chih-Chun Wang – p.6/41
Sym. Chs: Assuming $x = 0$. 

\[ i = 1 - 2 - 3 - 4 - 5 - 6 \]

\[ j = 1 \]

\[ \begin{align*}
  j &= 1 \\
  i &= 1 \\
  2 \\
  3 \\
  4 \\
  5 \\
  6
\end{align*} \]
The Density Evolution

Sym. Chs: Assuming \( x = 0 \).
Sym. Chs: Assuming $x = 0$.

\[ i = 1 \rightarrow \square \rightarrow \square \rightarrow \square \rightarrow \square \rightarrow \square \rightarrow \square \]
\[ j = 1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 6 \]
The Density Evolution

Sym. Chs: Assuming $x = 0$.

$$P^{(l)} = P^{(0)} \otimes \left( Q^{(l-1)} \right)^{\otimes (d_v-1)}$$

$$Q^{(l-1)} = \Gamma^{-1} \left( \left( \Gamma \left( P^{(l-1)} \right) \right)^{\otimes (d_c-1)} \right)$$
[Richardson et al. 01] The DE perfectly describes the decoding behavior for sufficiently large codeword length $n$. 
[Richardson et al. 01] The DE perfectly describes the decoding behavior for sufficiently large codeword length $n$.

The tail error probability goes to zero $\iff$ the channel of interest is decodable.
[Richardson et al. 01] The DE perfectly describes the decoding behavior for sufficiently large codeword length $n$.

The tail error probability goes to zero $\iff$ the channel of interest is decodable.

In practice, the performance for $n = 10^4 \sim 10^6$ bits is well-predicted.
[Richardson et al. 01] The DE perfectly describes the decoding behavior for sufficiently large codeword length $n$.

The tail error probability goes to zero $\iff$ the channel of interest is decodable.

In practice, the performance for $n = 10^4 \sim 10^6$ bits is well-predicted.

Advantages:
- **Accurate** performance prediction
- **Deterministic**, one-time computation instead of random Monte Carlo simulations
- Can be used as the **ultimate performance metric** for comparisons/code optimization
**BP Decoder** | **Optimality** | **Analysis Tools** | **Simulation**
--- | --- | --- | ---
Cycle-free + Non-Sym. Chs. | ✓ | ✓ | optimal
LDPC Codes + Sym. Chs. | ? | Dens. Evo. | outstanding
LDPC Codes + Non-Sym. Chs. | ? | ? | very good
Codeword Dependence of DE

Sym. Chs: Assuming $x = 0$.

\[ P^{(l)} = P^{(0)} \otimes \left( Q^{(l-1)} \right)^{\otimes (d_v - 1)} \]

\[ Q^{(l-1)} = \Gamma^{-1} \left( \left( \Gamma \left( P^{(l-1)} \right) \right)^{\otimes (d_c - 1)} \right) \]
Codeword Dependence of DE

Non-sym. Chs: Codeword-dependent.

\[ j = 1 \quad 2 \quad 3 \quad 4 \]

\[ i = 1 \quad 2 \quad 3 \quad 4 \]

\[
P^{(l)}(x) = P^{(0)}(x) \otimes \left( Q^{(l-1)}(x) \right)^{\otimes(d_v-1)}
\]

\[
Q^{(l-1)}(x) = \Gamma^{-1} \left( \left( \Gamma \left( P^{(l-1)}(x) \right) \right)^{\otimes(d_c-1)} \right)
\]
Codeword Averaging?

\[ j = 1 \quad 2 \quad 3 \quad 4 \]
\[ i = 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \]

\[ x^1_{(1,1)} := \{x_1x_5x_6 : x_1x_5x_6 = 000, 011, 101, 110\} \]
Codeword Averaging?

\[ j = 1, 2, 3, 4 \]

\[ i = 1, 2, 3, 4, 5, 6 \]

\[ H = \begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1
\end{pmatrix} \]

\[ Hx = 0 \iff x_6 = 0 \]

\[ X^1_{(1,1)} := \{ x_1x_5x_6 : x_1x_5x_6 = 000, 011, 101, 110 \} \]
Codeword Averaging?

\[ j = 1, 2, 3, 4 \]
\[ i = 1, 2, 3, 4, 5, 6 \]

\[
H = \begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1 \\
\end{pmatrix}
\]

\[ Hx = 0 \iff x_6 = 0 \]

\[ X_{(1,1)}^1 := \{ x_1x_5x_6 : x_1x_5x_6 = 000, 011, 101, 110 \} \]
No matter how large the tree is, averaging the trimmed tree code may not be equivalent to averaging the original code.
Definition (Perfect Projection) The supporting tree $\mathcal{N}^{2l}$ is perfectly projected, if

$$\left| \left\{ x \in \mathcal{X} : x_{\text{tree}} = x_t \right\} \right| = \frac{1}{|X_t|}.$$

In other words, averaging the trimmed tree code is equivalent to averaging over the original code. ★★★
Perfect Projection Condition

**Definition (Perfect Projection)** The supporting tree $\mathcal{N}^{2l}$ is perfectly projected, if

$$\frac{|\{x \in X : |x|_{\text{tree}} = x_t\}|}{|X|} = \frac{1}{|X_t|}.$$ 

♡♡♡ In other words, averaging the trimmed tree code is equivalent to averaging over the original code. ♡♡♡

$$P^{(l)}(x) := \langle P^{(l)}(x) \rangle_{\{x \in X : x_0 = x\}} = \langle P^{(l)}(x_t) \rangle_{\{x_t \in X_t : x_t_0 = x\}}$$

$$p^{(l)}_e = \frac{1}{2} \left( \int_{m=-\infty}^{0} P^{(l)}(0)(dm) + \int_{m=-\infty}^{0} P^{(l)}(1)(dm) \right)$$
New Iterative Formula for DE

\[
\forall x \in \{0, 1\}, \ P^{(l)}(x) = P^{(0)}(x) \otimes \left( Q^{(l-1)}(x) \right)^{\otimes (d_v-1)}
\]

\[
Q^{(l-1)}(x) = \Gamma^{-1} \left( \frac{1}{2^{d_c-2}} \sum_{x^1 \in X^1(x)}^{d_c-1} \otimes \Gamma \left( P^{(l-1)}(x_v) \right) \right)
\]

\[
\therefore \text{Parity-check equation,}
\]

\[
x = 0, \ x_1x_2 = 00, 11
\]

\[
x = 1, \ x_1x_2 = 01, 10
\]
New Iterative Formula for DE

\[
\begin{align*}
\forall x \in \{0,1\}, \quad P^{(l)}(x) &= P^{(0)}(x) \otimes (Q^{(l-1)}(x))^{(d_c-1)} \\
Q^{(l-1)}(x) &= \Gamma^{-1} \left( \frac{1}{2^{d_c-2}} \sum_{x^1 \in X^1(x)}^{d_c-1} \bigotimes_{v=1} \Gamma \left( P^{(l-1)}(x_v) \right) \right) \\
&= \Gamma^{-1} \left( \Gamma \left( \frac{P^{(l-1)}(0) + P^{(l-1)}(1)}{2} \right) \right)^{(d_c-1)} + (-1)^{x} \left( \Gamma \left( \frac{P^{(l-1)}(0) - P^{(l-1)}(1)}{2} \right) \right)^{(d_c-1)}
\end{align*}
\]

A simplified expression

\[
\begin{align*}
\because \text{Parity-check equation,} \\
x &= 0, \quad x_1x_2 = 00, 11 \\
x &= 1, \quad x_1x_2 = 01, 10
\end{align*}
\]
Three Theoretical Foundations

- Existence of the cycle-free support tree.~ [Richardson et al. 01]
Three Theoretical Foundations

- Existence of the cycle-free support tree. [Richardson et al. 01]
- **Theorem (Convergence to Perfect Projection in Probability)**

Consider the regular code ensemble $C^n(d_v, d_c)$ with codeword length $n$.

$$\mathbb{P}(\mathcal{N}^{2l_0} \text{ is perfectly projected}) = 1 - \mathcal{O}(n^{-0.1}).$$

*Proof:* Asymptotic codeword weight distribution, the rank of random matrices, and the constraint propagation argument.
Three Theoretical Foundations

- Existence of the cycle-free support tree. [Richardson et al. 01]
- **Theorem (Convergence to Perfect Projection in Probability)**

Consider the regular code ensemble $C^n(d_v, d_c)$ with codeword length $n$.

$$P \left( \mathcal{N}^{2l_0} \text{ is perfectly projected} \right) = 1 - \mathcal{O}(n^{-0.1}).$$

*Proof:* Asymptotic codeword weight distribution, the rank of random matrices, and the constraint propagation argument.

- Graphical properties vs. Algebraic properties
Three Theoretical Foundations

- Existence of the cycle-free support tree. \( \sim \) [Richardson et al. 01]

- **Theorem (Convergence to Perfect Projection in Probability)**
  Consider the regular code ensemble \( C_n^{d_v, d_c} \) with codeword length \( n \).

  \[
P ( N^{2l_0} \text{ is perfectly projected} ) = 1 - \mathcal{O}(n^{-0.1}).
\]

*Proof:* Asymptotic codeword weight distribution, the rank of random matrices, and the constraint propagation argument.

- Graphical properties vs. Algebraic properties

- Performance concentration. \( \sim \) [Richardson et al. 01]
Generalized DE

For non-symmetric channels, the codeword-averaged performance is predicted by our generalized DE when the codeword length $n$ is sufficiently large.

$$\forall x \in \{0, 1\}, \quad P^{(l)}(x) = P^{(0)}(x) \otimes \left( Q^{(l-1)}(x) \right)^{\otimes (d_c - 1)}$$

$$Q^{(l-1)}(x) = \Gamma^{-1} \left( \Gamma \left( \frac{P^{(l-1)}(0) + P^{(l-1)}(1)}{2} \right) \right)^{\otimes (d_c - 1)} \left( \Gamma \left( \frac{P^{(l-1)}(0) - P^{(l-1)}(1)}{2} \right) \right)^{\otimes (d_c - 1)} + (-1)^x \left( \Gamma \left( \frac{P^{(l-1)}(0) - P^{(l-1)}(1)}{2} \right) \right)^{\otimes (d_c - 1)} \left( \Gamma \left( \frac{P^{(l-1)}(0) + P^{(l-1)}(1)}{2} \right) \right)^{\otimes (d_c - 1)}$$
For non-symmetric channels, the codeword-averaged performance is predicted by our generalized DE when the codeword length $n$ is sufficiently large.

For $\forall x \in \{0, 1\}$, $P^{(l)}(x) = P^{(0)}(x) \otimes \left( Q^{(l-1)}(x) \right)^{\otimes (d_c-1)}$

$Q^{(l-1)}(x) = \Gamma^{-1} \left( \left( \Gamma \left( \frac{P^{(l-1)}(0) + P^{(l-1)}(1)}{2} \right) \right)^{\otimes (d_c-1)} \right)$

$+ (-1)^x \left( \Gamma \left( \frac{P^{(l-1)}(0) - P^{(l-1)}(1)}{2} \right) \right)^{\otimes (d_c-1)}$

The same computational complexity as the classical DE.
<table>
<thead>
<tr>
<th>BP Decoder</th>
<th>Optimality</th>
<th>Analysis Tools</th>
<th>Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cycle-free + Non-Sym. Chs.</td>
<td>✓</td>
<td>✓</td>
<td>optimal</td>
</tr>
<tr>
<td>LDPC Codes + Sym. Chs.</td>
<td>?</td>
<td>Dens. Evo.</td>
<td>outstanding</td>
</tr>
<tr>
<td>LDPC Codes + Non-Sym. Chs.</td>
<td>?</td>
<td>?</td>
<td>very good</td>
</tr>
<tr>
<td>BP Decoder</td>
<td>Optimality</td>
<td>Analysis Tools</td>
<td>Simulation</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>------------</td>
<td>----------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Cycle-free + Non-Sym. Chs.</td>
<td>✓</td>
<td>✓</td>
<td>optimal</td>
</tr>
<tr>
<td>LDPC Codes + Sym. Chs.</td>
<td>?</td>
<td>Dens. Evo.</td>
<td>outstanding</td>
</tr>
<tr>
<td>LDPC Codes + Non-Sym. Chs.</td>
<td>?</td>
<td>?</td>
<td>very good</td>
</tr>
<tr>
<td>BP Decoder</td>
<td>Optimality</td>
<td>Analysis Tools</td>
<td>Simulation</td>
</tr>
<tr>
<td>-----------------------------------</td>
<td>------------</td>
<td>----------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Cycle-free + Non-Sym. Chs.</td>
<td>✓</td>
<td>✓</td>
<td>optimal</td>
</tr>
<tr>
<td>LDPC Codes + Sym. Chs.</td>
<td>?</td>
<td>Dens. Evo.</td>
<td>outstanding</td>
</tr>
<tr>
<td>LDPC Codes + Non-Sym. Chs.</td>
<td>?</td>
<td>New Dens. Evo.</td>
<td>very good</td>
</tr>
<tr>
<td>BP Decoder</td>
<td>Optimality</td>
<td>Analysis Tools</td>
<td>Simulation</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>------------</td>
<td>----------------</td>
<td>--------------</td>
</tr>
<tr>
<td>Cycle-free + Non-Sym. Chs.</td>
<td>✓</td>
<td>✓</td>
<td>optimal</td>
</tr>
<tr>
<td>LDPC Codes + Sym. Chs.</td>
<td>?</td>
<td>Dens. Evo.</td>
<td>outstanding</td>
</tr>
<tr>
<td>LDPC Codes + Non-Sym. Chs.</td>
<td>?</td>
<td>New Dens. Evo.</td>
<td>very good</td>
</tr>
</tbody>
</table>
12A: [Richardson01], optimized for BiAWGNC, maxDv=12.

12B: Optimized for z-channels, maxDv=12.
Optimized Rate 1/2 Codes

- Code complexity: $\max d_v = 20$ and $\max d_c = 10$.
  - Allowing 100 iterations: $p_{1 \rightarrow 0}^* = 0.2741$.
  - Allowing 100 iterations: $p_{1 \rightarrow 0}^* = 0.2796$.  

\[ \begin{array}{c}
0 \quad \rightarrow \quad 0 \\
\downarrow \\
1 \quad \rightarrow \quad 1 \\
\end{array} \]

\[ p_{1 \rightarrow 0} \]

\[ 1 - p_{1 \rightarrow 0} \]
Optimized Rate 1/2 Codes

- Code complexity: $\max d_v = 20$ and $\max d_c = 10$.
  - Allowing 100 iterations: $p_{1 \rightarrow 0}^* = 0.2741$.
  - Allowing 100 iterations: $p_{1 \rightarrow 0}^* = 0.2796$.
- Symmetric information rate bound: $p_{1 \rightarrow 0}^* \leq 0.2932$. 

![Diagram showing the transition probabilities](attachment:image.png)
Optimized Rate 1/2 Codes

- Code complexity: $\max d_v = 20$ and $\max d_c = 10$.
  - Allowing 100 iterations: $p_{1\to0}^* = 0.2741$.
  - Allowing 100 iterations: $p_{1\to0}^* = 0.2796$.
- Symmetric information rate bound:
  $p_{1\to0}^* \leq 0.2932$.
- Away from the capacity bound: $p_{1\to0}^* \leq 0.3035$. 
Optimized Rate 1/2 Codes

- Code complexity: $\max d_v = 20$ and $\max d_c = 10$.
  - Allowing 100 iterations: $p_{1 \rightarrow 0}^* = 0.2741$.
  - Allowing 100 iterations: $p_{1 \rightarrow 0}^* = 0.2796$.
- Symmetric information rate bound:
  $p_{1 \rightarrow 0}^* \leq 0.2932$.
- Away from the capacity bound: $p_{1 \rightarrow 0}^* \leq 0.3035$.
- Not far
  - [Majani & Rumsey 91] showed that the ratio between the symmetric mutual information rate and the capacity is lower bounded by $\frac{e \ln 2}{2} \approx 94\%$.
  - [Shulman & Feder 04] further proved that the absolute difference is upper bounded by 0.011 bit/sym.
When to Stop DE?

Decodable $\iff \lim p_e^{(l)} = 0$.

Computationally, how to check $\lim p_e^{(l)} = 0$?
When to Stop DE?

- Decodable $\iff \lim p_e^{(l)} = 0$.
  - Computationally, how to check $\lim p_e^{(l)} = 0$?
- An alternative one dimensional index of $P^{(l)}(x)$:
  - Bhattacharyya Noise Parameter (BNP):
Decodable $\iff \lim p_{e}^{(l)} = 0$.

Computationally, how to check $\lim p_{e}^{(l)} = 0$?

An alternative one dimensional index of $P^{(l)}(x)$:

Bhattacharyya Noise Parameter (BNP):

$000 \cdots 0$ vs. $111 \cdots 1$. 
When to Stop DE?

- Decodable $\iff \lim p_e^{(l)} = 0$.

- Computationally, how to check $\lim p_e^{(l)} = 0$?

- An alternative one dimensional index of $P^{(l)}(x)$:
  - Bhattacharyya Noise Parameter (BNP):
    $000 \cdots 0$ vs. $111 \cdots 1$, $\implies p_{e,n} \approx BNP^n$
When to Stop DE?

- Decodable $\iff \lim p_e^{(l)} = 0$.
  - Computationally, how to check $\lim p_e^{(l)} = 0$?
- An alternative one dimensional index of $P^{(l)}(x)$:
  - Bhattacharyya Noise Parameter (BNP):
    - 000 $\cdots$ 0 vs. 111 $\cdots$ 1, $\implies p_{e,n} \approx BNP^n$
    - $BNP^{(l)} := \int e^{-m/2}\langle P^{(l)} \rangle (dm)$. 

When to Stop DE?

- Decodable $\iff \lim pe_l^{(l)} = 0$.
  - Computationally, how to check $\lim pe_l^{(l)} = 0$?

- An alternative one dimensional index of $P_l^{(l)}(x)$:
  - **Bhattacharyya Noise Parameter** (BNP):
    - $000 \ldots 0$ vs. $111 \ldots 1 \implies pe_n \approx BNP^n$
    - $BNP_l^{(l)} := \int e^{-m/2}\langle P^{(l)}(l) \rangle (dm)$.

- **Lemma**: $2pe_l^{(l)} \leq BNP_l^{(l)} \leq 2\sqrt{pe_l^{(l)}(1 - pe_l^{(l)})}$, even for non-symmetric channels.
When to Stop DE?

- Decodable $\iff \lim p_e^{(l)} = 0$.
- Computationally, how to check $\lim p_e^{(l)} = 0$?
- An alternative one dimensional index of $P^{(l)}(x)$:
  - Bhattacharyya Noise Parameter (BNP):
    - $000 \cdots 0$ vs. $111 \cdots 1 \implies p_{e,n} \approx BNP^n$
    - $\text{BNP}^{(l)} := \int e^{-m/2} \langle P^{(l)} \rangle (dm)$.
  - **Lemma:** $2p_e^{(l)} \leq \text{BNP}^{(l)} \leq 2\sqrt{p_e^{(l)} (1 - p_e^{(l)})}$, even for non-symmetric channels.
- Describing the bipartite graph: **Degree distribution** polynomials,
  - Variable nodes: $\lambda(x) := \sum \lambda_k x^{k-1}$
  - Check nodes: $\rho(x) := \sum \rho_k x^{k-1}$.
A Stopping Criterion

1: if $\text{BNP}^{(0)} \geq \frac{1}{\lambda_2 \rho'(1)}$ then
2: $\lim_{l \to \infty} p_e^{(l)} > 0 \iff \text{Undecodable.}$
3: else
4: repeat
5: Perform DE iteration
6: until $\text{BNP}^{(l)} < \epsilon^*$
7: $\lim_{l \to \infty} p_e^{(l)} = 0 \iff \text{Decodable.}$
8: end if
A Stopping Criterion

1: if $\text{BNP}^{(0)} \geq \frac{1}{\lambda_2 \rho'(1)}$ then
2: $\lim_{l \to \infty} p_e^{(l)} > 0 \iff$ Undecodable.
3: else
4: repeat
5: Perform DE iteration
6: until $\text{BNP}^{(l)} < \epsilon^*$
7: $\lim_{l \to \infty} p_e^{(l)} = 0 \iff$ Decodable.
8: end if

Example: Code 12B

\[
\frac{1}{\lambda_2 \rho'(1)} = 0.6430 \\
\text{BNP}^{(0)} = 0.5253 \\
\epsilon^* = 0.0675
\]
A Stopping Criterion

1: if $\text{BNP}^{(0)} \geq \frac{1}{\lambda_2\rho'(1)}$ then
2: $\lim_{l \to \infty} p_e^{(l)} > 0 \iff \text{Undecodable.}$
3: else
4: repeat
5: Perform DE iteration
6: until $\text{BNP}^{(l)} < \epsilon^*$
7: $\lim_{l \to \infty} p_e^{(l)} = 0 \iff \text{Decodable.}$
8: end if

Stability Conditions

Example: Code 12B

$$\frac{1}{\lambda_2\rho'(1)} = 0.6430$$
$$\text{BNP}^{(0)} = 0.5253$$
$$\epsilon^* = 0.0675$$
A Stopping Criterion

1: if $\text{BNP}(0) \geq \frac{1}{\lambda_2 \rho'(1)}$ then
2: $\lim_{l \to \infty} p_e^{(l)} > 0 \iff$ Undecodable.
3: else
4: repeat
5: Perform DE iteration
6: until $\text{BNP}^{(l)} < \epsilon^*$
7: $\lim_{l \to \infty} p_e^{(l)} = 0 \iff$ Decodable.
8: end if

Stability Conditions

Example: Code 12B
\[
\frac{1}{\lambda_2 \rho'(1)} = 0.6430
\]
\[
\text{BNP}(0) = 0.5253
\]
\[
\epsilon^* = 0.0675
\]

Non-symmetric channels:

- lin. LDPC ENC
- Non-sym. CH.
- lin. LDPC DEC
LDPC Coset Code Ensemble

Non-symmetric channels:

1. lin. LDPC ENC
2. Non-sym. CH.
3. lin. LDPC DEC

Symmetrized channels:

1. lin. LDPC ENC
2. Rand. Bits
3. Non-sym. CH.
4. lin. LDPC DEC
**LDPC Coset Code Ensemble**

**Non-symmetric channels:**

1. **lin. LDPC ENC** → **Non-sym. CH.** → **lin. LDPC DEC**

**Symmetrized channels:**

- **lin. LDPC ENC** → **Rand. Bits** → **Non-sym. CH.** → **lin. LDPC DEC**

**Symmetric Channel**

**LDPC coset code ensemble:** \( Hx = s \) and \( s \in_{\text{rand.}} \{0, 1\}^{n(1-R)}. \)
**LDPC Coset Code Ensemble**

Non-symmetric channels:
- lin. LDPC ENC → Non-sym. CH. → lin. LDPC DEC

Symmetrized channels:
- Randomly Chosen LDPC Coset ENC
  - Rand. Bits
  - Lin. LDPC ENC → Non-sym. CH. → Lin. LDPC DEC
- Matched LDPC Coset ENC
  - Lin. LDPC ENC + Non-sym. CH. + Lin. LDPC DEC

**LDPC coset code ensemble:** $Hx = s$ and $s \in_{\text{rand.}} \{0, 1\}^{n(1-R)}$. 

Chih-Chun Wang – p.20/41
LDPC Coset Code Ensemble

Non-symmetric channels:

- lin. LDPC ENC → Non-sym. CH. → lin. LDPC DEC

Symmetrized channels:

- Randomly Chosen LDPC Coset ENC
- Lin. LDPC ENC + Rand. Bits → Non-sym. CH. → Lin. LDPC DEC

LDPC coset code ensemble: $Hx = s$ and $s \in_{\text{rand.}} \{0, 1\}^{n(1-R)}$.

- Their stability conditions coincide.
- Nearly identical performance under Monte-Carlo simulations.
LDPC Coset Code Ensemble

Non-symmetric channels:
- lin. LDPC ENC -> Non-sym. CH. -> lin. LDPC DEC

Symmetrized channels:
- lin. LDPC ENC + Rand. Bits + Non-sym. CH. + lin. LDPC DEC

Randomly Chosen LDPC Coset ENC

Matched LDPC Coset DEC

LDPC coset code ensemble: $Hx = s$ and $s \in_{\text{rand.}} \{0, 1\}^{n(1-R)}$.

- Their stability conditions coincide.
- Nearly identical performance under Monte-Carlo simulations.
The Answer

With the help of our new DE,

- Not equivalent.
- But close.
The Answer

With the help of our new DE,

- Not equivalent.
- But close. Even for Z-channels, the discrepancy is $< 0.05\%$ for $d_c \geq 6$. 
The Answer

With the help of our new DE,

- Not equivalent.
- But close. Even for Z-channels, the discrepancy is \(< 0.05\%\) for \(d_c \geq 6\).
- A new phenomenon: The larger \(d_c\), the smaller the discrepancy.
The Answer

With the help of our new DE,

- Not equivalent.
- But close. Even for Z-channels, the discrepancy is < 0.05% for \( d_c \geq 6 \).
- A new phenomenon: The larger \( d_c \), the smaller the discrepancy.

\[
\lim_{d_{c,\min} \to \infty} \left\langle P^{(l_0)}_{\text{linear}} \right\rangle \overset{D}{=} P^{(l_0)}_{\text{coset}} \quad \text{exponentially}
\]
The Answer

With the help of our new DE,

- Not equivalent.
- But close. Even for Z-channels, the discrepancy is $< 0.05\%$ for $d_c \geq 6$.
- A new phenomenon: The larger $d_c$, the smaller the discrepancy.

\[
\lim_{d_{c,\text{min}} \to \infty} \left< P_{\text{linear}}^{(l_0)} \right> \overset{D}{=} P_{\text{coset}}^{(l_0)} \quad \text{exponentially}
\]

\[
\lim_{d_{c,\text{min}} \to \infty} \left| p_{1 \to 0,\text{linear}}^* - p_{1 \to 0,\text{coset}}^* \right| = 0
\]
The Answer

With the help of our new DE,

- Not equivalent.
- But close. Even for Z-channels, the discrepancy is < 0.05% for $d_c \geq 6$.
- A new phenomenon: The larger $d_c$, the smaller the discrepancy.

\[
\lim_{d_{c,\text{min}} \to \infty} \left\langle P_{\text{linear}}^{(l_0)} \right\rangle \overset{\mathcal{D}}{=} P_{\text{coset}}^{(l_0)} \quad \text{exponentially}
\]

\[
\lim_{d_{c,\text{min}} \to \infty} \left| p_{1 \to 0,\text{linear}}^* - p_{1 \to 0,\text{coset}}^* \right| = 0
\]

Applications

Non-symmetric channels:

- lin. LDPC ENC
- Non-sym. CH.
- lin. LDPC DEC

Symmetrized channels:

- LDPC Coset ENC
- Rand. Bits
- Non-sym. CH.
- lin. LDPC DEC

⇑ almost equivalent for moderate $d_c$
Applications

Non-symmetric channels:
- lin. LDPC ENC
- Non-sym. CH.
- lin. LDPC DEC

Symmetrized channels:
- LDPC Coset ENC
- Rand. Bits
- Non-sym. CH.
- Lin. LDPC DEC

Almost equivalent for moderate $d_c$
Applications

Non-symmetric channels:

- lin. LDPC ENC
- Non-sym. CH.
- lin. LDPC DEC

Symmetrized channels:

LDPC Coset ENC
- Lin. LDPC ENC
- Non-sym. CH.
- Lin. LDPC DEC

LDPC Coset DEC
- Fixed b

几乎等价于适度的 $d_c$ 时

[Kavčić 03]: 几乎所有 b 都是典型的。
Applications

Non-symmetric channels:

- lin. LDPC ENC
- Non-sym. CH.
- lin. LDPC DEC

Symmetrized channels:

- LDPC Coset ENC
- Fixed b
- LDPC Coset DEC

Almost equivalent for moderate $d_c$

- [Kavčić 03]: Almost all $b$ are typical.

- Applications: Analysis — EXIT chart/fast DE,
Applications

Non-symmetric channels:

lin. LDPC ENC \rightarrow \text{Non-sym. CH.} \rightarrow \text{lin. LDPC DEC}

Symmetrized channels:

LDPC Coset ENC \leftrightarrow \text{almost equivalent for moderate } d_c \leftrightarrow \text{LDPC Coset DEC}

\[\text{Fixed } b\]

[\text{Kavčić 03}]: Almost all \(b\) are typical.

Applications: Analysis — EXIT chart/fast DE, Simulations — using the all-zero codeword,
Applications

Non-symmetric channels:

- lin. LDPC ENC
- Non-sym. CH.
- lin. LDPC DEC

Symmetrized channels:

- LDPC Coset ENC
- Fixed b
- Non-sym. CH.
- lin. LDPC DEC

LDPC Coset ENC

LDPC Coset DEC

almost equivalent for moderate $d_c$

- [Kavčić 03]: Almost all $b$ are typical.

- Applications: Analysis — EXIT chart/fast DE, Simulations — using the all-zero codeword, & Implementation — linear codes.
Perfect Projection

Cycle Free Convergence

Generalized Dens. Evo.

Lin. Code

Asym. Ch.

Classical Dens. Evo.

Lin. Code

Sym. Ch.
Perfect Projection

Cycle Free Convergence

Generalized Dens. Evo.

Classical Dens. Evo.

Lin. Code

Coset Code

Lin. Code

Asym. Ch.

Sym. Ch.

Practical Sys.
Perfect Projection

Cycle Free Convergence

Generalized Dens. Evo.

Lin. Code

Asym. Ch.

Classical Dens. Evo.

Coset Code

Lin. Code

Sym. Ch.

Kavčić’s Typ. Thm.

Practical Sys.

Chih-Chun Wang – p.23/41
Perfect Projection

Generalized Dens. Evo.

Classical Dens. Evo.

Lin. Code

Typicality

Asym. Ch.

Lin. Code

Coset Code

Sym. Ch.

Practical Sys.

Kavčič’s Typ. Thm.

Cycle Free Convergence
**BP Decoder** | **Optimality** | **Analysis Tools** | **Simulation**
--- | --- | --- | ---
Cycle-free + Non-Sym. Chs. | ✓ | ✓ | optimal
LDPC Codes + Sym. Chs. | ? | Dens. Evo. | outstanding
LDPC Codes + Non-Sym. Chs. | ? | New Dens. Evo. | outstanding
Local Optimality of BP

- BP is optimal when applied to cycle-free networks.
- Exceptional performance when the network has cycles.
Local Optimality of BP

- BP is optimal when applied to cycle-free networks.
- Exceptional performance when the network has cycles.
- Bethe–Kikuchi free energy approximation.
Local Optimality of BP

- BP is optimal when applied to cycle-free networks.
- Exceptional performance when the network has cycles.
- Bethe–Kikuchi free energy approximation.

Is BP locally optimal? Namely, given local observations on $\mathcal{N}^{(2l)}$ and complete knowledge of the codebook, can a maximum a posteriori probability (MAP) decoder do better?
Local Optimality of BP

Is BP locally optimal? Namely, given local observations on \( \mathcal{N}^{(2l)} \) and complete knowledge of the codebook, can a maximum a posteriori probability (MAP) decoder do better?

BP Decoder

\[ x_1 \]

\[ x_5 \]

\[ x_6 \]
Is BP locally optimal? Namely, given local observations on $\mathcal{N}^{(2l)}$ and complete knowledge of the codebook, can a maximum a posteriori probability (MAP) decoder do better?

$$H = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$Hx = 0 \iff x_6 = 0$$
Is BP locally optimal? Namely, given local observations on $\mathcal{N}^{(2l)}$ and complete knowledge of the codebook, can a maximum a posteriori probability (MAP) decoder do better?

$$H x = 0 \iff x_6 = 0$$

Where $H$ is the parity check matrix:

$$H = \begin{pmatrix}
1 & 1 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 1 & 1
\end{pmatrix}$$
Local Optimality of BP

Is BP locally optimal? Namely, given local observations on $\mathcal{N}^{(2l)}$ and complete knowledge of the codebook, can a maximum a posteriori probability (MAP) decoder do better?

The cycle-free assumption is not enough. The condition that the support tree $\mathcal{N}^{(2l)}$ is perfectly projected guarantees the local optimality of BP.
<table>
<thead>
<tr>
<th>BP Decoder</th>
<th>Optimality</th>
<th>Analysis Tools</th>
<th>Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cycle-free + Non-Sym. Chs.</td>
<td>✓</td>
<td>✓</td>
<td>optimal</td>
</tr>
<tr>
<td>LDPC Codes + Sym. Chs.</td>
<td>?</td>
<td>Dens. Evo.</td>
<td>outstanding</td>
</tr>
<tr>
<td>LDPC Codes + Non-Sym. Chs.</td>
<td>?</td>
<td>New Dens. Evo.</td>
<td>outstanding</td>
</tr>
</tbody>
</table>
BP Decoder | Optimality | Analysis Tools | Simulation
---|---|---|---
Cycle-free + Non-Sym. Chs. | ✓ | ✓ | optimal
LDPC Codes + Sym. Chs. | ? | Dens. Evo. | outstanding
LDPC Codes + Non-Sym. Chs. | ? | New Dens. Evo. | outstanding


Typicality - Coset Code


Local Opt. of BP

Perfect Projection

Classical Dens. Evo.

Lin. Code

Typicality

Asym. Ch.

Kavčić’s Typ. Thm.

BP Decoder

<table>
<thead>
<tr>
<th>Optimality</th>
<th>Analysis Tools</th>
<th>Simulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>✓</td>
<td>optimal</td>
</tr>
<tr>
<td>Local</td>
<td>Dens. Evo.</td>
<td>outstanding</td>
</tr>
<tr>
<td>Local</td>
<td>New Dens. Evo.</td>
<td>outstanding</td>
</tr>
</tbody>
</table>
Part I — Linear LDPC codes on symmetric & non-symmetric channels

Part II — EXtrinsic Information (EXIT) chart & Finite-dim. bounds for LDPC codes
Part I — Linear LDPC codes on symmetric & non-symmetric channels

Part II — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes
  Binary-input/non-symmetric-output channels
  Bhattacharyya-noise-parameter-based (BNP-based) bound for non-symmetric channels
Content

Part I — Linear LDPC codes on symmetric & non-symmetric channels

Part II — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes

- **Binary-input/non-symmetric-output** channels
  - Bhattacharyya-noise-parameter-based (BNP-based) bound for non-symmetric channels

- **Binary-input/symmetric-output** channels
  - A two-dimensional bound
  - A non-iterative soft-bit-based bound $\Rightarrow$ Tight for BSCs
Part I — Linear LDPC codes on symmetric & non-symmetric channels

Part II — EXtrinsic Information (EXIT) chart & Finite-dim. bounds for LDPC codes

- **Binary-input/non-symmetric-output channels**
  - Bhattacharyya-noise-parameter-based (BNP-based) bound for non-symmetric channels

- **Binary-input/symmetric-output channels**
  - A two-dimensional bound
  - A non-iterative soft-bit-based bound $\Rightarrow$ Tight for BSCs

- **$\mathbb{Z}_m$-input/symmetric-output channels**
  - An $m$-ary BNP-based bound
  - The necessary and sufficient stability conditions
  - $\mathbb{Z}_m$ LDPC coded modulation
Density Evolution

- Iteratively trace the infinite-dim. distribution of the log-likelihood ratio (LLR) message.

Finite-Dim. Bounds
DE vs. Finite-Dim. Bounds

**Density Evolution**
- Iteratively trace the infinite-dim. distribution of the log-likelihood ratio (LLR) message.
- Precisely predicts asymptotic performance.

**Finite-Dim. Bounds**
DE vs. Finite-Dim. Bounds

Density Evolution

- Iteratively trace the infinite-dim. distribution of the log-likelihood ratio (LLR) message.
- Precisely predicts asymptotic performance.
- DE is required on every channel model of interest.

Finite-Dim. Bounds
DE vs. Finite-Dim. Bounds

**Density Evolution**
- Iteratively trace the \textit{infinite-dim.} distribution of the log-likelihood ratio (LLR) message.
- Precisely predicts asymptotic performance.
- DE is required on every channel model of interest.
- Higher computational complexity.

**Finite-Dim. Bounds**
DE vs. Finite-Dim. Bounds

**Density Evolution**
- Iteratively trace the infinite-dim. distribution of the log-likelihood ratio (LLR) message.
- Precisely predicts asymptotic performance.
- DE is required on every channel model of interest.
- Higher computational complexity.

**Finite-Dim. Bounds**
- Iteratively trace a finite-dim. index of the LLR distribution.
- Relatively loosely bound the asymptotic performance.
- **Uniform performance** on all types of channels can be inferred.
- Simpler iteration formula admits more analysis.
DE vs. Finite-Dim. Bounds

**Density Evolution**
- Iteratively trace the infinite-dim. distribution of the log-likelihood ratio (LLR) message.
- Precisely predicts asymptotic performance.
- DE is required on every channel model of interest.
- Higher computational complexity.

**Finite-Dim. Bounds**
- Iteratively trace a finite-dim. index of the LLR distribution
- Relatively loosely bound the asymptotic performance
- **Uniform performance** on all types of channels can be inferred.
- Simpler iteration formula admits more analysis.

Reference: [Burshtein, Miller 01], [Khandekar 02], [Land et al. 03], and [Sutskover, Shamai 03].
The support tree is a \(\{0, 1\} \mapsto \mathbf{Y}\) channel, where \(\mathbf{Y} = \mathbb{R}^{\text{\# involved var. nodes}}\).
Iterative Bounding Technique

The support tree is a $\{0, 1\} \rightarrow Y$ channel, where $Y = R^\#\{\text{involved var. nodes}\}$. 
Iterative Bounding Technique

The support tree is a $\{0, 1\} \rightarrow Y$ channel, where $Y = \mathbb{R}^{\#\text{involved var. nodes}}$. 

[Diagram of a tree structure with nodes labeled as "Ch"]
The support tree is a $\{0, 1\} \rightarrow Y$ channel, where $Y = \mathbb{R}^{\#\text{involved var. nodes}}$. 
Assumption: binary-input/symmetric-output channels.

Iteratively trace the evolution of a one-dimensional index.
Assumption: binary-input/symmetric-output channels.

Iteratively trace the evolution of a one-dimensional index.

\[ p_e := P\{X \neq \hat{X}_{ML}(Y)\}. \]

[Burshtein and Miller 02]: \( SB := 2E_{XY}\{P(\bar{X}|Y)\}. \) Note: \( SB \neq 2p_e \)

[Khandekar and McEliece 01]: \( CB := E_{XY}\{\sqrt{\frac{P(\bar{X}|Y)}{P(X|Y)}}\}. \)

[Land et al. 03], [Sutskover et al. 03]: Mutual information, or equivalently, conditional entropy \( h := H_2(X|Y). \)
Existing Finite Dim. Bnds.

- Assumption: binary-input/symmetric-output channels.
- Iteratively trace the evolution of a one-dimensional index.

\[ p_e := P\{X \neq \hat{X}_{ML}(Y)\} \]

[Burshtein and Miller 02]: \( SB := 2E_{XY}\{P(\hat{X}|Y)\} \). Note: \( SB \neq 2p_e \)

[Khandekar and McEliece 01]: \( CB := E_{XY}\{\sqrt{\frac{P(\hat{X}|Y)}{P(X|Y)}}\} \).

[Land et al. 03], [Sutskover et al. 03]: Mutual information, or equivalently, conditional entropy \( h := H_2(X|Y) \).

- All are based on the convexity/concavity of their transfer functions.
\[ \text{INFO}_{out} \geq V_{\text{BSC}}(\text{INFO}_{in,1}, \text{INFO}_{in,1}) \]

\[ \text{INFO}_{out} \leq V_{\text{BEC}}(\text{INFO}_{in,1}, \text{INFO}_{in,1}) \]
INFO_{out} \geq V_{BSC}(INFO_{in,1}, INFO_{in,1})
INFO_{out} \leq V_{BEC}(INFO_{in,1}, INFO_{in,1})
INFO_{out} \geq C_{BEC}(INFO_{in,1}, INFO_{in,1})
INFO_{out} \leq C_{BSC}(INFO_{in,1}, INFO_{in,1})
Application 1: Iterative Bounds

Var: $\text{INFO}_{out} \geq V_{BSC}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Chk: $\text{INFO}_{out} \geq C_{BEC}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Iterative Bound: $\text{INFO}^{(l+1)} \geq V_{BSC} \left( \text{INFO}^{(0)}, C_{BEC} \left( \text{INFO}^{(l)} \right) \right)$

Reference: [Land et al. 03], and [Sutskover, Shamai 03]
Application 2: EXIT Chart

Var: $\text{INFO}_{out} \approx V_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Chk: $\text{INFO}_{out} \approx C_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Iterative Equation:

$\text{INFO}^{(l+1)} \approx V_{\text{Gaussian}} \left( \text{INFO}^{(0)}, C_{\text{Gaussian}} \left( \text{INFO}^{(l)} \right) \right)$
Application 2: EXIT Chart

Var: $\text{INFO}_{out} \approx V_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Chk: $\text{INFO}_{out} \approx C_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Iterative Equation:

$\text{INFO}^{(l+1)} \approx V_{\text{Gaussian}} \left( \text{INFO}^{(0)}, C_{\text{Gaussian}} \left( \text{INFO}^{(l)} \right) \right)$
Application 2: EXIT Chart

Var: $\text{INFO}_{out} \approx V_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Chk: $\text{INFO}_{out} \approx C_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Iterative Equation:

$\text{INFO}^{(l+1)} \approx V_{\text{Gaussian}} \left( \text{INFO}^{(0)}, C_{\text{Gaussian}} \left( \text{INFO}^{(l)} \right) \right)$
Application 2: EXIT Chart

Var: $\text{INFO}_{out} \approx V_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Chk: $\text{INFO}_{out} \approx C_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Iterative Equation:

$\text{INFO}^{(l+1)} \approx V_{\text{Gaussian}} \left( \text{INFO}^{(0)}, C_{\text{Gaussian}} \left( \text{INFO}^{(l)} \right) \right)$
Application 2: EXIT Chart

Var: $\text{INFO}_{out} \approx V_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Chk: $\text{INFO}_{out} \approx C_{\text{Gaussian}}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$

Iterative Equation:

$\text{INFO}^{(l+1)} \approx V_{\text{Gaussian}} \left( \text{INFO}^{(0)}, C_{\text{Gaussian}} \left( \text{INFO}^{(l)} \right) \right)$
The First Bound for Non-Symmetric Channels

\[ \text{BNP}^{(l+1)} \leq \text{BNP}^{(0)} \left( 1 - \left( 1 - \text{BNP}^{(l)} \right)^{d_c - 1} \right)^{d_v - 1} \]
The First Bound for Non-Symmetric Channels

\[ BNP^{(l+1)} \leq BNP^{(0)} \left( 1 - \left( 1 - BNP^{(l)} \right)^{d_e-1} \right)^{d_v-1} \]

Uniformly good performance for general fading channels.
\( UB_{\text{BNP}, \text{SB}} \) given \((\text{BNP}_{in}, \text{SB}_{in})\)

Equivalent to finding the universal optimizer for all \( p_2 \in [0, 1/2] \):

\[
\forall p_2 \in [0, 1/2], \max \int BNP_{out}(p_1, p_2) dP_1(p_1)
\]

and \( \int SB_{out}(p_1, p_2) dP_1(p_1) \)

subject to \( \int BNP(p_1) dP_1(p_1) \leq BNP_{in,1} \)

\( \int SB(p_1) dP_1(p_1) \leq SB_{in,1} \)
$UB_{BNP,SB}$ given $(BNP_{in,1}, SB_{in,1})$

Equivalent to finding the universal optimizer for all $p_2 \in [0, 1/2]$:

\[
\forall p_2 \in [0, 1/2], \max \quad \int BNP_{out}(p_1, p_2) dP_1(p_1)
\]

and

\[
\int SB_{out}(p_1, p_2) dP_1(p_1)
\]

subject to

\[
\int BNP(p_1) dP_1(p_1) \leq BNP_{in,1}
\]

\[
\int SB(p_1) dP_1(p_1) \leq SB_{in,1}
\]

For check nodes, such a universal maximizer $dP^*_1(p_1)$ exists.
Equivalent to finding the universal optimizer for all \( p_2 \in [0, 1/2] \):

\[
\forall p_2 \in [0, 1/2], \quad \int BNP_{out}(p_1, p_2) dP_1^*(p_1)
\]

and

\[
\int SB_{out}(p_1, p_2) dP_1^*(p_1)
\]

satisfying

\[
\int BNP(p_1) dP_1^*(p_1) \leq BNP_{in,1}
\]

\[
\int SB(p_1) dP_1^*(p_1) \leq SB_{in,1}
\]

For check nodes, such a universal maximizer \( dP_1^*(p_1) \) exists.
$UB_{BNP,SB}$ given $(BNP_{in}, SB_{in})$

Equivalent to finding the universal optimizer for all $p_2 \in [0, 1/2]$:

$$\forall p_2 \in [0, 1/2], \max \int BNP_{out}(p_1, p_2) dP_1(p_1)$$

and

$$\int SB_{out}(p_1, p_2) dP_1(p_1)$$

subject to

$$\int BNP(p_1) dP_1(p_1) \leq BNP_{in,1}$$

$$\int SB(p_1) dP_1(p_1) \leq SB_{in,1}$$

For check nodes, such a universal maximizer $dP_1^*(p_1)$ exists.

For variable nodes, no such universal maximizer exists.

Alternatively, we find a universal bounding distr., $dP_1^+(p_1)$. 
Given \( UB_{\text{BNP,SB}} \), we are interested in finding the universal optimizer for all \( p_2 \in [0, 1/2] \):

\[
\int \text{BNP}_{\text{out}}(p_1, p_2) dP_1^+(p_1) \geq \int \text{BNP}_{\text{out}}(p_1, p_2) dP_1(p_1), \text{ and}
\]

\[
\int \text{SB}_{\text{out}}(p_1, p_2) dP_1^+(p_1) \geq \int \text{SB}_{\text{out}}(p_1, p_2) dP_1(p_1), \forall p_2 \in [0, 1/2]
\]

subject to

\[
\int \text{BNP}(p_1) dP_1(p_1) \leq \text{BNP}_{\text{in,1}}
\]

\[
\int \text{SB}(p_1) dP_1(p_1) \leq \text{SB}_{\text{in,1}}
\]

- For check nodes, such a universal maximizer \( dP_1^*(p_1) \) exists.
- For variable nodes, no such universal maximizer exists.

Alternatively, we find a universal bounding distr., \( dP_1^+(p_1) \).
\[ dP^*(p) \text{ and } dP^+(p) \]

\[
dP_1^*(p_1) = \begin{cases} 
1 - \frac{\text{BNP}_{in,1}}{t} & \text{if } p_1 = 0 \\
\frac{\text{BNP}_{in,1}}{t} & \text{if } 2\sqrt{p_1(1-p_1)} = t \\
0 & \text{otherwise}
\end{cases}
\]

\[
dP_1^+(p_1) = \begin{cases} 
(1 - f_{SB}) \frac{t}{t + \text{BNP}_{in,1}} & \text{if } 2\sqrt{p_1(1-p_1)} = \text{BNP}_{in,1} \\
f_{SB} & \text{if } 2\sqrt{p_1(1-p_1)} = \sqrt{\text{SB}_{in,1}} \\
(1 - f_{SB}) \frac{\text{BNP}_{in,1}}{t + \text{BNP}_{in,1}} & \text{if } 2\sqrt{p_1(1-p_1)} = t \\
0 & \text{otherwise}
\end{cases}
\]

\[
t = \frac{\text{SB}_{in,1}}{\text{BNP}_{in,1}}
\]

\[
f_{SB} = \begin{cases} 
0 & \text{if } 2\sqrt{t\text{BNP}_{in,1}} - t + \sqrt{\text{BNP}_{in,1}(2t - \text{BNP}_{in,1})} \geq 0 \\
\frac{\eta(w^*)}{2(t - \text{BNP}_{in,1})^2} & \text{otherwise}
\end{cases}
\]

\[
\eta(w) = w^3 - 2tw^2 + (t - \text{BNP}_{in,1})^2w
\]

\[
w^* = \begin{cases} 
2\sqrt{t\text{BNP}_{in,1}} & \text{if } \eta'(2\sqrt{t\text{BNP}_{in,1}}) \leq 0 \\
\frac{2t - \sqrt{4t^2 - 3(t - \text{BNP}_{in,1})^2}}{3} & \text{otherwise}
\end{cases}
\]
UB_{BNP,SB}  

Strict improvements over existing one-dimensional bounds.
A Non-iterative Tight $SB$ Bound

Iteration-based approach vs. Non-iteration-based approach
The same $H$ as in the GF(2) codes. But $Hx = 0$ in $\mathbb{Z}_m$. 
$\mathbb{Z}_m$-based LDPC Codes

- The same $H$ as in the GF(2) codes. But $Hx = 0$ in $\mathbb{Z}_m$.
- Consider only circularly sym. $\mathbb{Z}_m$-input channels.
$\mathbb{Z}_m$-based LDPC Codes

- The same $H$ as in the $\mathbb{GF}(2)$ codes. But $Hx = 0$ in $\mathbb{Z}_m$.
- Consider only circularly sym. $\mathbb{Z}_m$-input channels.
- Define the pairwise Bhattacharyya noise parameter as $\text{BNP}(0 \rightarrow x)$, and a vector representation $\text{BNP} = (\text{BNP}(0 \rightarrow x))_{x \in \mathbb{Z}_m}$. \

$\mathbb{Z}_m$-based LDPC Codes

- The same $H$ as in the GF(2) codes. But $Hx = 0$ in $\mathbb{Z}_m$.
- Consider only circularly sym. $\mathbb{Z}_m$-input channels.
- Define the pairwise Bhattacharyya noise parameter as $\text{BNP}(0 \rightarrow x)$, and a vector representation $\text{BNP} = (\text{BNP}(0 \rightarrow x))_{x \in \mathbb{Z}_m}$.

**Theorem (A Pairwise-BNP-Based Bound)**

$$\text{BNP}^{(l+1)} \leq \text{BNP}^{(0)} \prod_{j=1}^{d_v-1} \left( \bigotimes_{i=1}^{d_c-1} \text{BNP}^{(l)} \right)$$
**$\mathbb{Z}_m$-based LDPC Codes**

- The same $H$ as in the GF(2) codes. But $Hx = 0$ in $\mathbb{Z}_m$.
- Consider only **circularly sym.** $\mathbb{Z}_m$-input channels.
- Define the pairwise Bhattacharyya noise parameter as $\text{BNP}(0 \rightarrow x)$, and a vector representation $\text{BNP} = (\text{BNP}(0 \rightarrow x))_{x \in \mathbb{Z}_m}$.

**Theorem** (A Pairwise-BNP-Based Bound)

$$\text{BNP}^{(l+1)} \leq \text{BNP}^{(0)} \prod_{j=1}^{d_v-1} \left( \bigotimes_{i=1}^{d_c-1} \text{BNP}^{(l)} \right)$$

- Matched nec. & suff. stability conditions, depending on

$$\max_{x \in \mathbb{Z}_m \setminus \{0\}} \text{BNP}^{(0)}(0 \rightarrow x)$$
$\mathbb{Z}_m$-based LDPC Codes

- The same $H$ as in the GF(2) codes. But $Hx = 0$ in $\mathbb{Z}_m$.
- Consider only circularly sym. $\mathbb{Z}_m$-input channels.
- Define the pairwise Bhattacharyya noise parameter as $\text{BNP}(0 \rightarrow x)$, and a vector representation $\text{BNP} = (\text{BNP}(0 \rightarrow x))_{x \in \mathbb{Z}_m}$.

**Theorem** *(A Pairwise-BNP-Based Bound)*

$$\text{BNP}^{(l+1)} \leq \text{BNP}^{(0)} \prod_{j=1}^{d_v-1} \left( \bigotimes_{i=1}^{d_c-1} \text{BNP}^{(l)} \right)$$

- Matched nec. & suff. stability conditions, depending on

$$\max_{x \in \mathbb{Z}_m \setminus \{0\}} \text{BNP}^{(0)}(0 \rightarrow x)$$

[MacKay et al. 98]: With the same amount of info. bits, $\mathbb{Z}_m$ codes have better performance at the cost of complexity, $\mathcal{O}(m)$. 
[MacKay et al. 98]: With the same amount of info. bits, $\mathbb{Z}_m$ codes have better performance at the cost of complexity, $\mathcal{O}(m)$.

$$m \in \mathbb{Z}_8 \xrightarrow{\text{LDPC ENC}} x \in \mathbb{Z}_8$$

$$\Psi : \{0, 1, \ldots, 7\} \mapsto S$$

$S = \{s_0, s_1, \ldots, s_7\}$
[MacKay et al. 98]: With the same amount of info. bits, $\mathbb{Z}_m$ codes have better performance at the cost of complexity, $\mathcal{O}(m)$.

$$m \in \mathbb{Z}_8 \xrightarrow{\text{LDPC ENC}} x \in \mathbb{Z}_8$$

$$\Psi : \mathcal{X} \mapsto \mathcal{S}$$

$$\mathcal{S} = \{s_0, s_1, \ldots, s_7\}$$

4 bit/sym 64QAM w. two $\mathbb{Z}_8$ regular (3,9) codes
\[ \mathbb{Z}_m \text{ LDPC Coded 8PAM} \]

- [MacKay et al. 98]: With the same amount of info. bits, \( \mathbb{Z}_m \) codes have better performance at the cost of complexity, \( \mathcal{O}(m) \).

\[ \Psi : X \mapsto S \]

\[ S = \{s_0, s_1, \ldots, s_7\} \]

4 bit/sym 64QAM w. two \( \mathbb{Z}_8 \) regular (3,9) codes

- 0.3dB better than BICM w. bin. regular codes
- 0.3dB worse than BICM w. bin. ir. codes
- Further code optimization is the next step.
- Compatible with almost all equalization schemes
Conclusion

Part I — Linear LDPC codes on symmetric & non-symmetric channels
Conclusion

Part I — Linear LDPC codes on symmetric & non-symmetric channels

- Local Opt. of BP
- Perfect Projection
- Generalized Dens. Evo.
- Practical Sys.
- Lin. Code
- Typicality
- Asym. Ch.
- Coset Code
- Lin. Code
- Sym. Ch.
- Kavčič's Typ. Thm.
Conclusion

- **Part I** — Linear LDPC codes on symmetric & non-symmetric channels
  - Local Opt. of BP
  - Perfect Projection
  - Generalized Dens. Evo.
  - Practical Sys.
  - Lin. Code
  - Typicality
  - Asym. Ch.
  - Coset Code
  - Lin. Code
  - Typicality
  - Asym. Ch.

- **Part II** — EXIT chart & finite-dim. bounds for LDPC codes
Conclusion

Part I — Linear LDPC codes on symmetric & non-symmetric channels

- Local Opt. of BP
- Perfect Projection
- Generalized Dens. Evo.
- Lin. Code
- Practical Sys.
- Typicality
- Asym. Ch.
- Cycle Free Convergence
- Classical Dens. Evo.
- Coset Code
- Sym. Ch.
- Kavčič's Typ. Thm.

Part II — EXIT chart & finite-dim. bounds for LDPC codes

- Regular (3,6) Code
- Outer Bnd by LB
- Inner Bnd by UB
- BEC
- GSN
- LAPLACE
- Rayleigh
- BICM Cap.
Future Work

LDPC Codes

- Improving the performance on finite codes

- BP Algorithms

- Practical schemes
Future Work

LDPC Codes
- Improving the performance on finite codes
  - Algebraic construction
  - Importance sampling
  - BER upper bounds for finite codes
- BP Algorithms

Practical schemes
Future Work

LDPC Codes

- Improving the performance on finite codes
  - Algebraic construction
  - Importance sampling
  - BER upper bounds for finite codes

BP Algorithms

- Pseudo-codeword consideration
- Soft decoding for RS codes

Practical schemes
Future Work

LDPC Codes

- Improving the performance on finite codes
  - Algebraic construction
  - Importance sampling
  - BER upper bounds for finite codes

- BP Algorithms
  - Pseudo-codeword consideration
  - Soft decoding for RS codes

- Practical schemes
  - Coded modulation, dirty paper codes, lossless data compression, etc.