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.

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

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

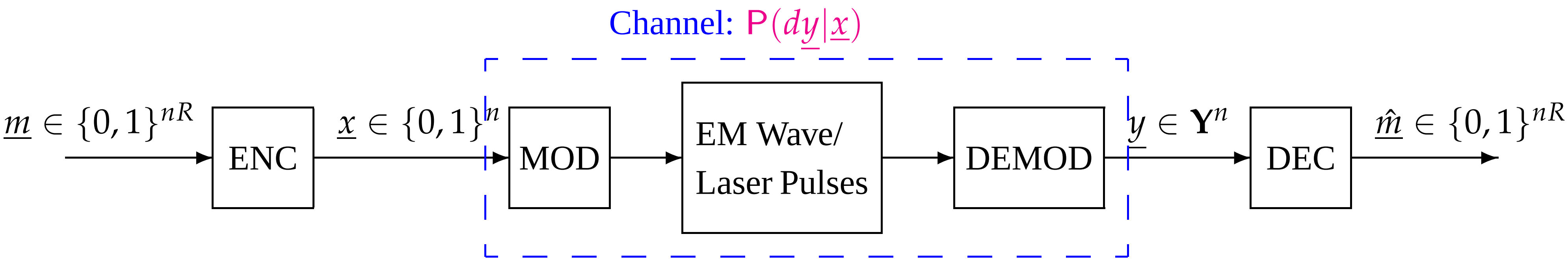
Content

- 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.

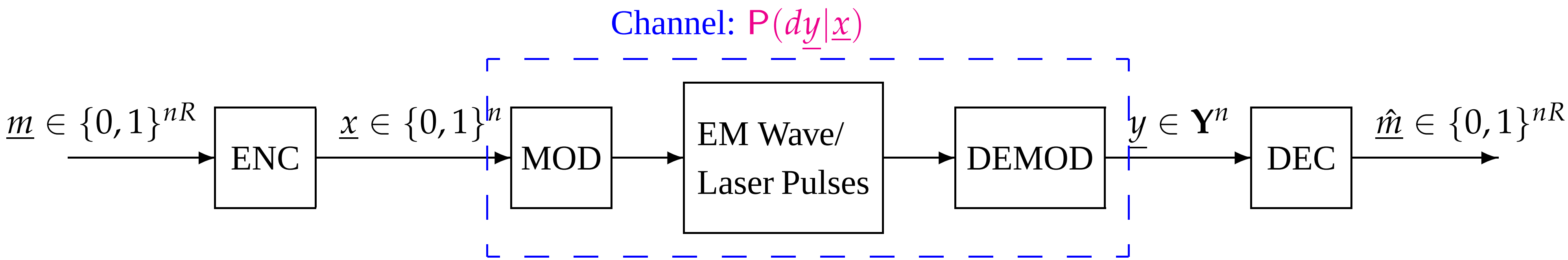
Content

- 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
    - The local optimality of the belief propagation decoder
- Part II — EXtrinsic InformaTion (EXIT) chart & Finite-dim. bounds for LDPC codes.

Error Correcting Codes

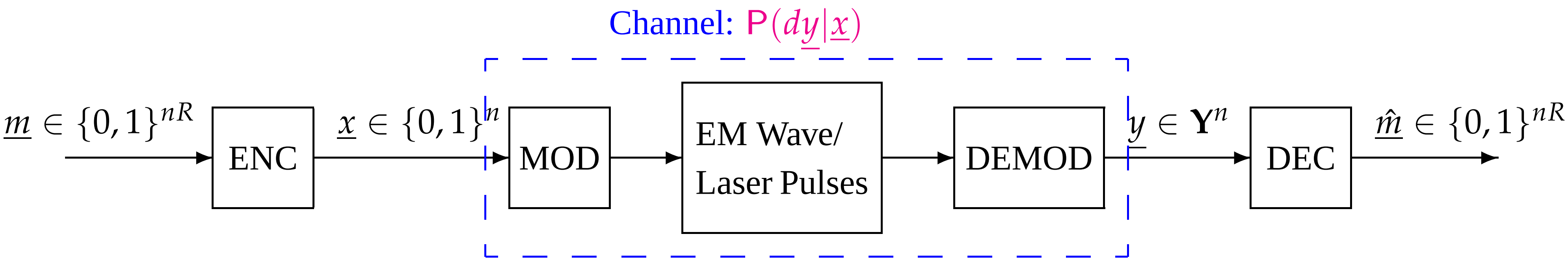


Error Correcting Codes



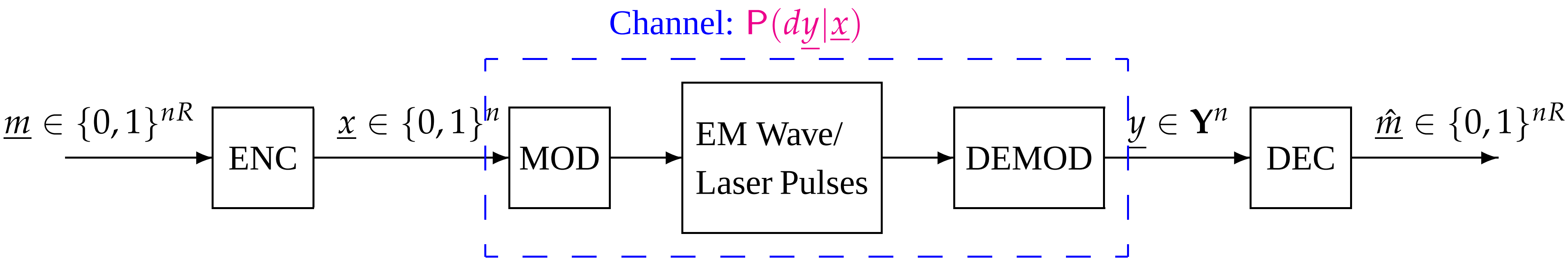
- Memoryless channels: $\mathsf{P}(d\underline{y}|\underline{x}) = \prod_{i=1}^{n} \mathsf{P}(dy_i|x_i)$.
- Symmetric channels: $\mathsf{P}(y|x=0) = \mathsf{P}(-y|x=1)$.

Error Correcting Codes



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

Error Correcting Codes



- Memoryless channels: $\mathsf{P}(d\underline{y}|\underline{x}) = \prod_{i=1}^{n} \mathsf{P}(dy_i|x_i)$.
- Symmetric channels: $\mathsf{P}(y|x=0) = \mathsf{P}(-y|x=1)$.
- 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∼1.5dB away from capacity.

Ultra high performance on almost all symmetric channels.

Ultra high performance on almost all symmetric channels.

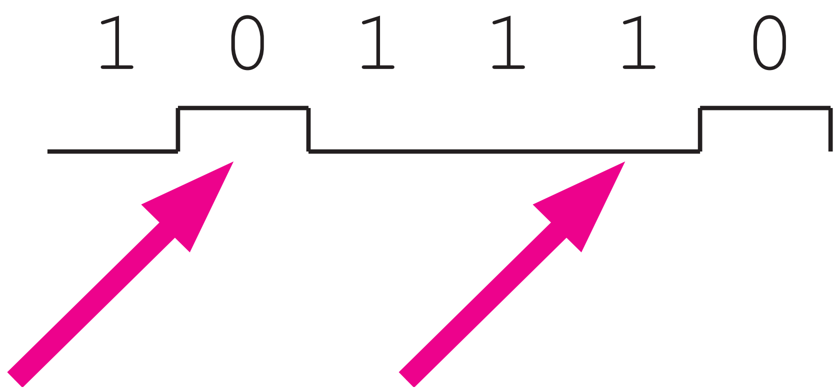
Question: How about non-symmetric memoryless channels?

Ultra high performance on almost all symmetric channels.

Question: How about non-symmetric memoryless channels?

Examples:

Z-Channels

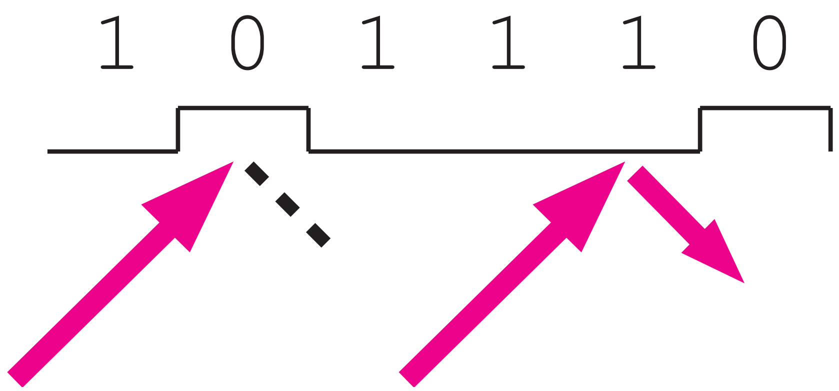


Ultra high performance on almost all symmetric channels.

Question: How about non-symmetric memoryless channels?

Examples:

Z-Channels

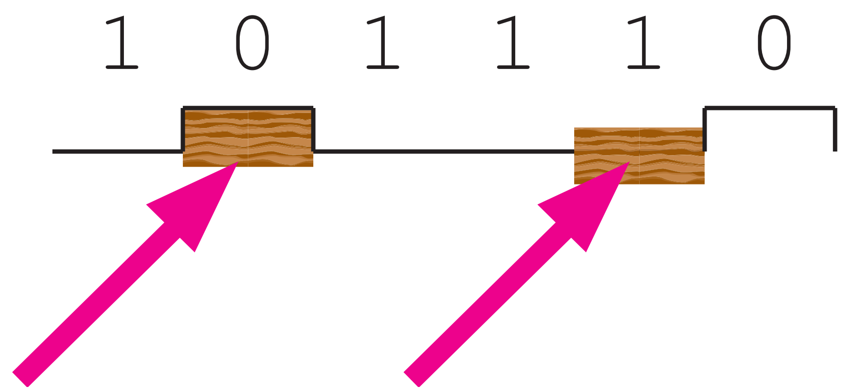


Ultra high performance on almost all symmetric channels.

Question: How about non-symmetric memoryless channels?

Examples:

Z-Channels

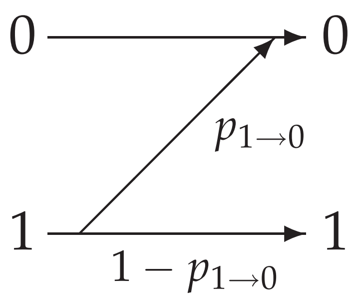


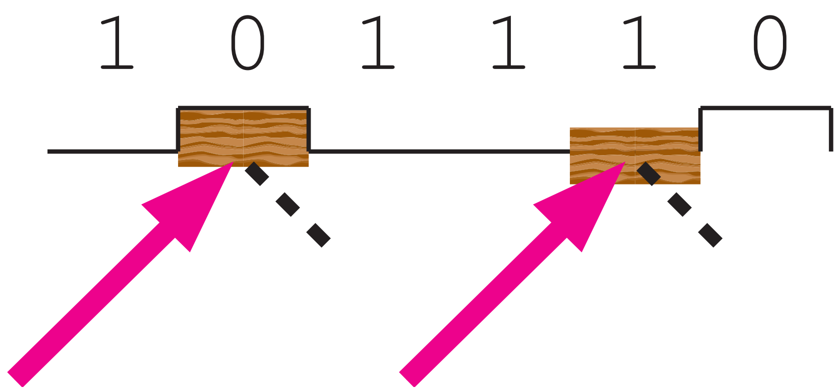
Ultra high performance on almost all symmetric channels.

Question: How about non-symmetric memoryless channels?

Examples:

Z-Channels



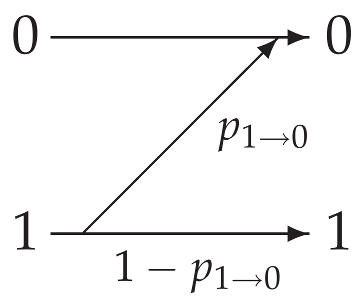


Ultra high performance on almost all symmetric channels.

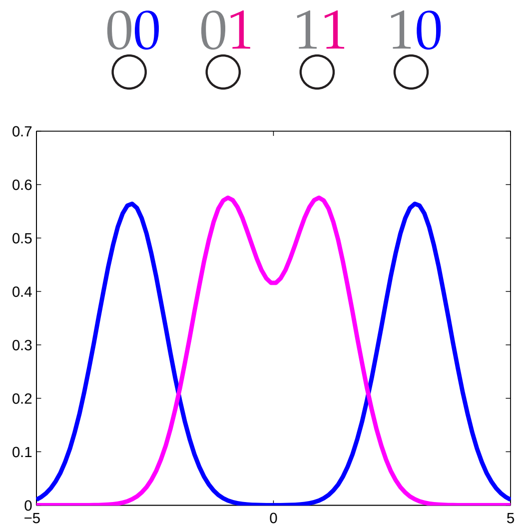
Question: How about non-symmetric memoryless channels?

Examples:

Z-Channels



BICM



LDPC Codes & BP Decoders

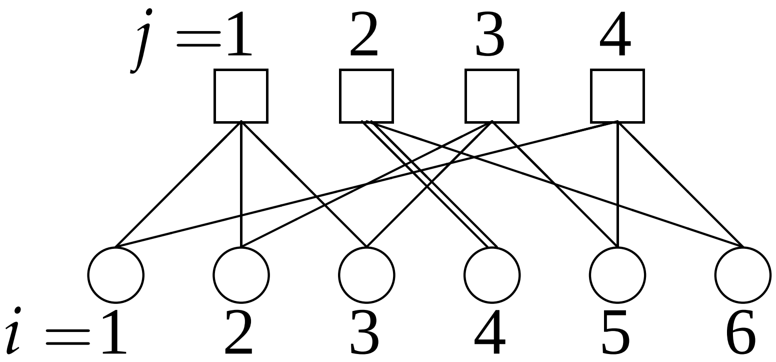
$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$\boxed{\mathbf{Hx} = \mathbf{0}}$$

LDPC Codes & BP Decoders

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

$$\boxed{\mathbf{Hx} = \mathbf{0}}$$

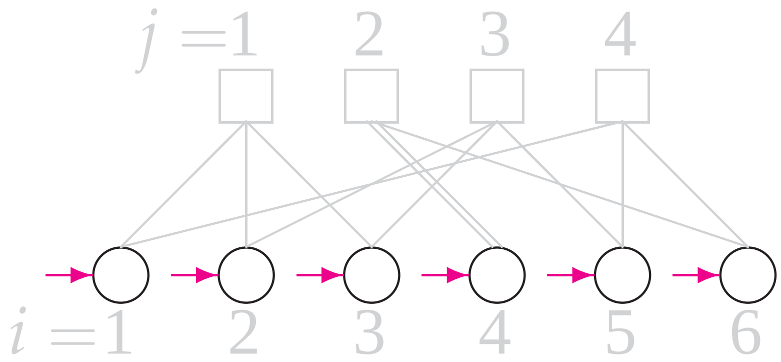


$\mathcal{C}^6(d_v, d_c), d_v = 2, d_c = 3$

LDPC Codes & BP Decoders

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

$$\boxed{\mathbf{Hx} = \mathbf{0}}$$



$\mathcal{C}^6(d_v, d_c), d_v = 2, d_c = 3$

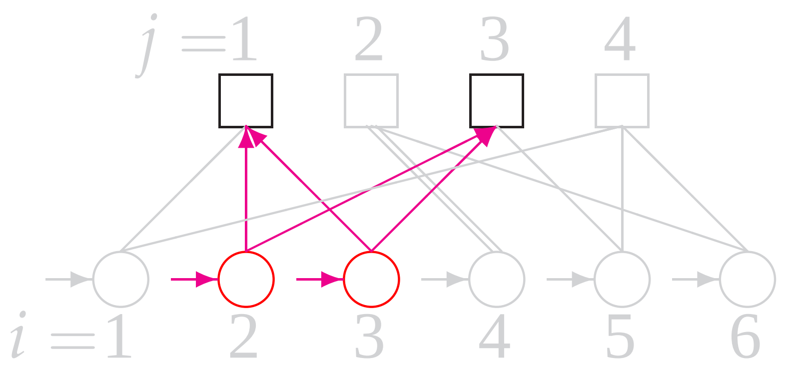
Message Passing Algorithms

(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

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

$$\mathbf{Hx} = \mathbf{0}$$



$\mathcal{C}^6(d_v, d_c), d_v = 2, d_c = 3$

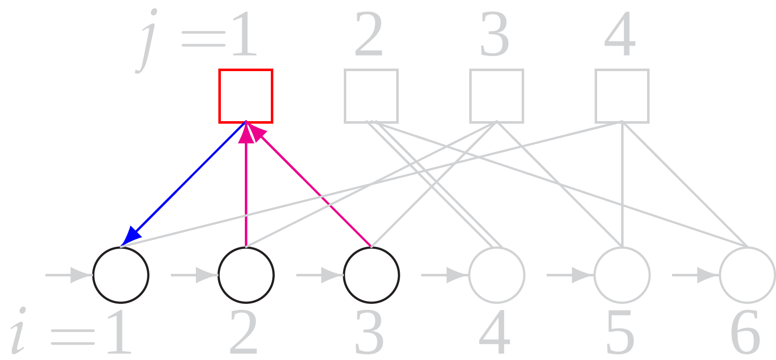
Message Passing Algorithms

(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

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

$$\boxed{\mathbf{Hx} = \mathbf{0}}$$



$\mathcal{C}^6(d_v, d_c)$, $d_v = 2$, $d_c = 3$

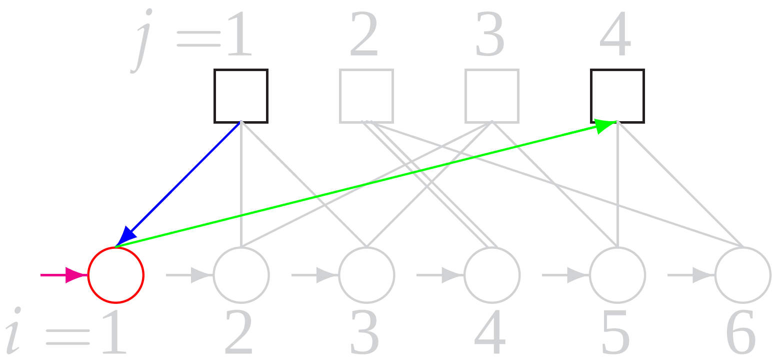
Message Passing Algorithms

(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

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

$$\mathbf{Hx} = \mathbf{0}$$



$\mathcal{C}^6(d_v, d_c)$, $d_v = 2$, $d_c = 3$

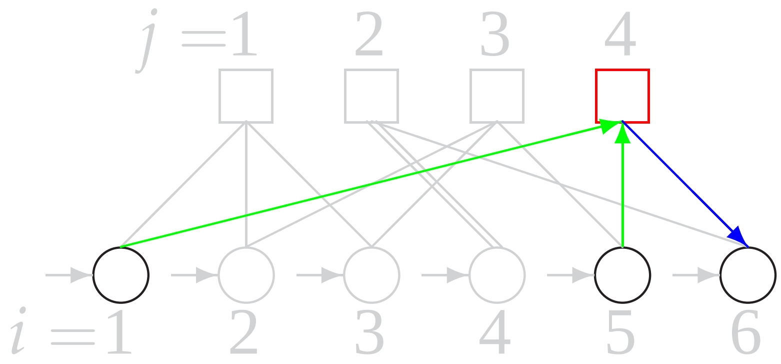
Message Passing Algorithms

(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

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

$$\mathbf{Hx} = \mathbf{0}$$



$\mathcal{C}^6(d_v, d_c), d_v = 2, d_c = 3$

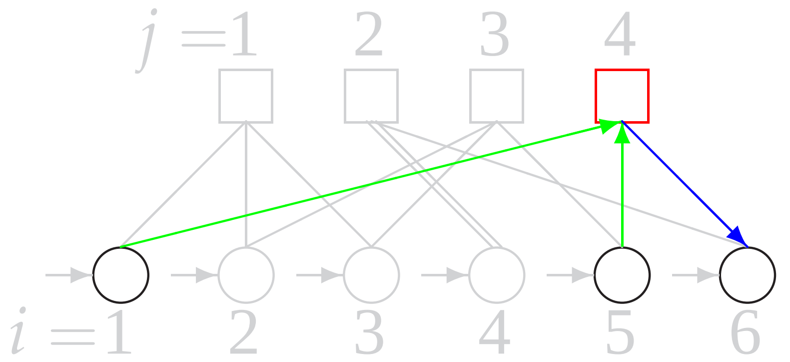
Message Passing Algorithms

(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

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

$$\boxed{\mathbf{Hx} = \mathbf{0}}$$



$\mathcal{C}^6(d_v, d_c)$, $d_v = 2$, $d_c = 3$

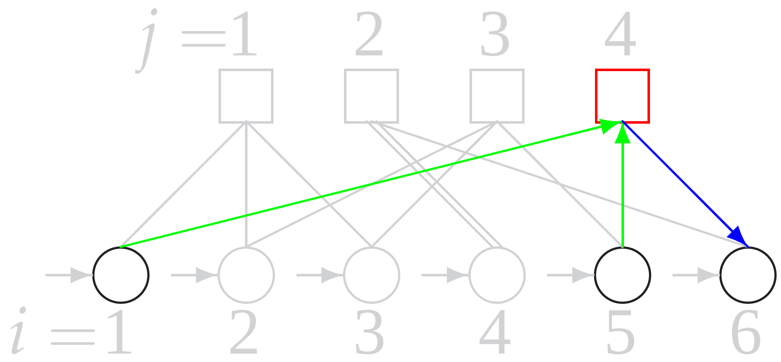
Message Passing Algorithms

(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

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

$$\boxed{\mathbf{Hx} = \mathbf{0}}$$



$\mathcal{C}^6(d_v, d_c), d_v = 2, d_c = 3$

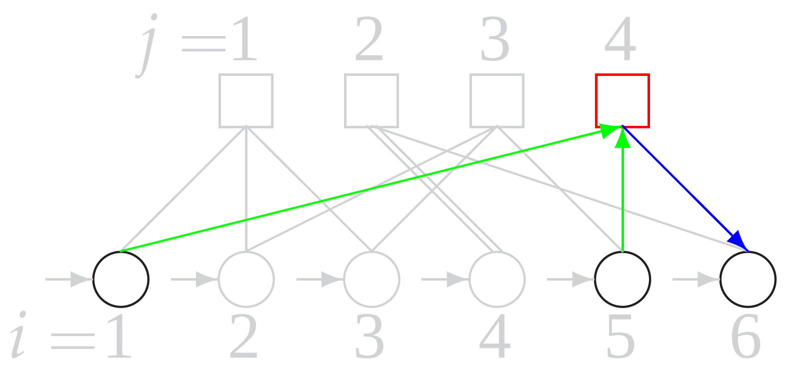
Belief Propagation Decoder:

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

LDPC Codes & BP Decoders

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

$\mathbf{Hx} = \mathbf{0}$



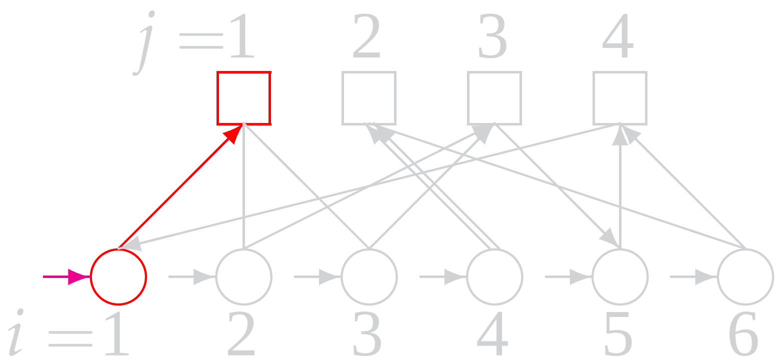
$\mathcal{C}^6(d_v, d_c)$, $d_v = 2$, $d_c = 3$

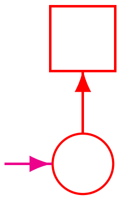
Belief Propagation Decoder:

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

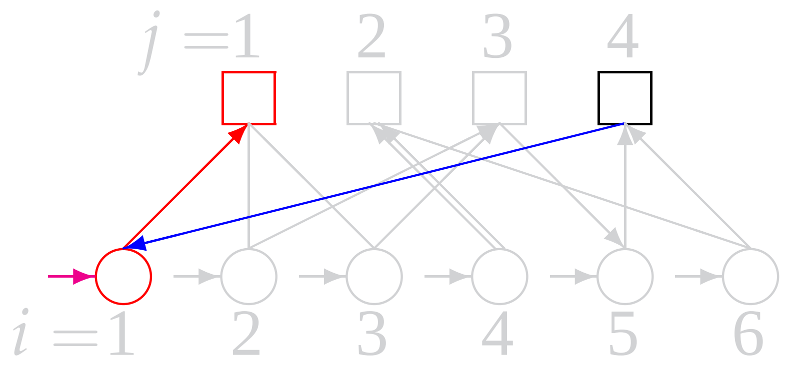
BP Decoder	Optimality	Analysis Tools	Simulation
Cycle-free + Non-Sym. Chs.	✓	✓	optimal
LDPC Codes + Sym. Chs.	?	✓	outstanding
LDPC Codes + Non-Sym. Chs.	?	?	very good

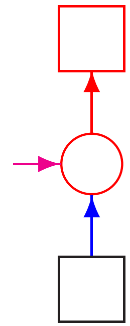
The Density Evolution



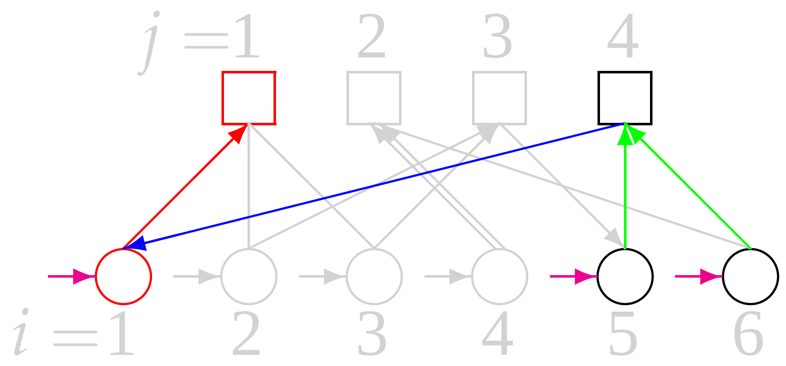


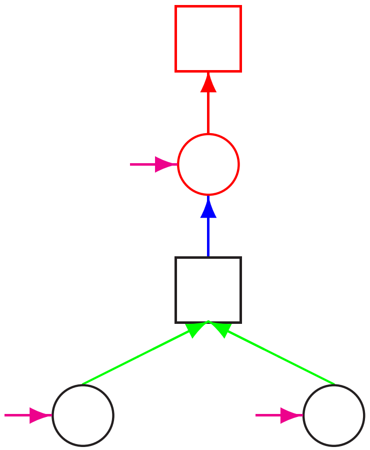
The Density Evolution



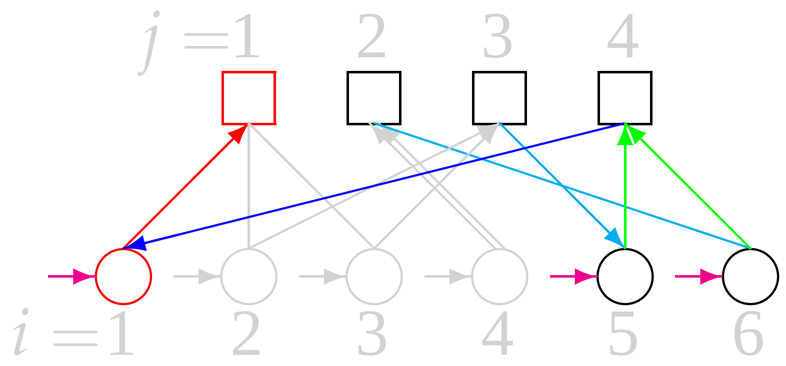


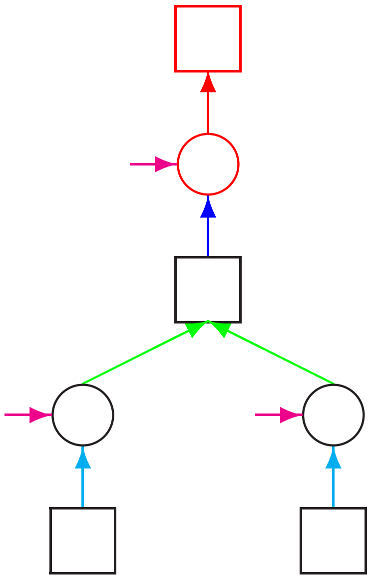
The Density Evolution



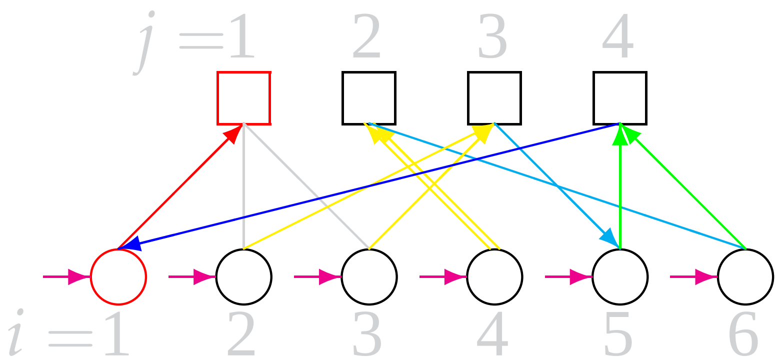


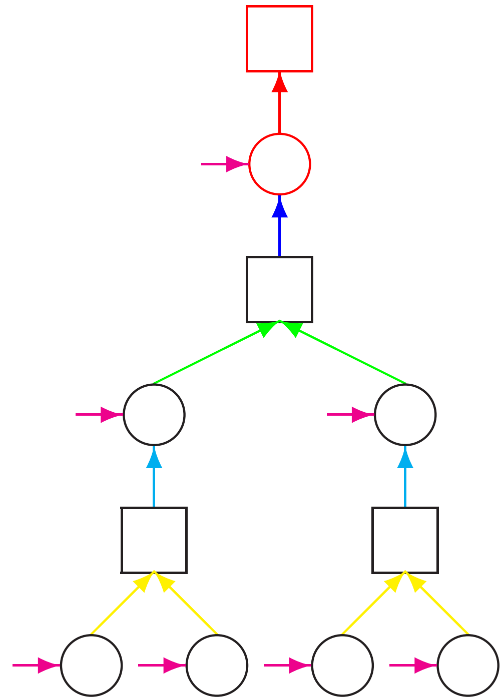
The Density Evolution





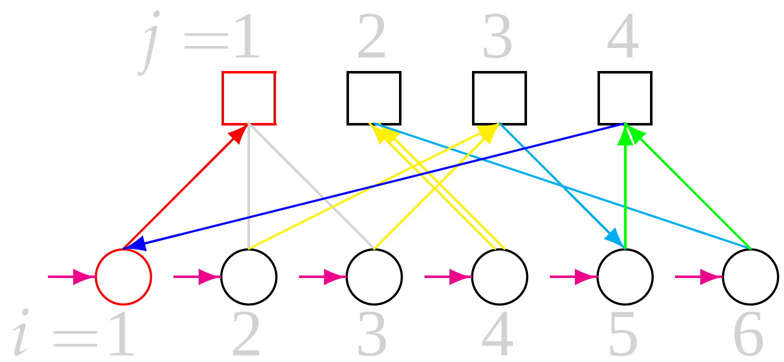
The Density Evolution

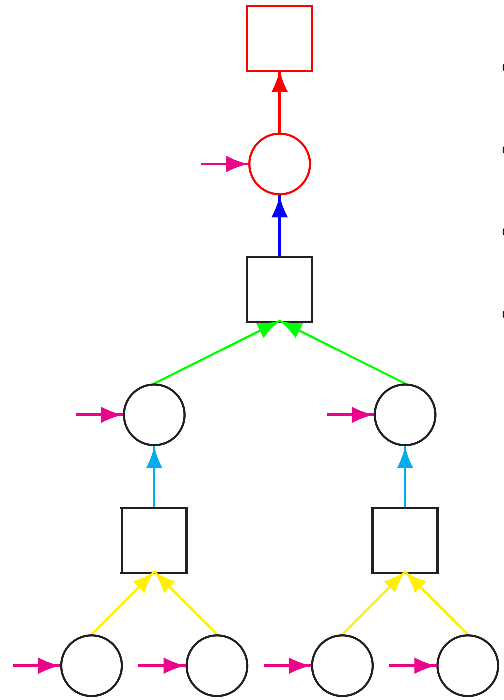


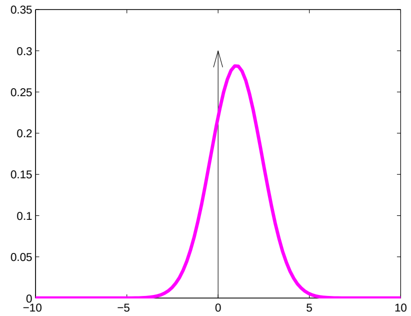


The Density Evolution

Sym. Chs: Assuming $\mathbf{x} = \mathbf{0}$.

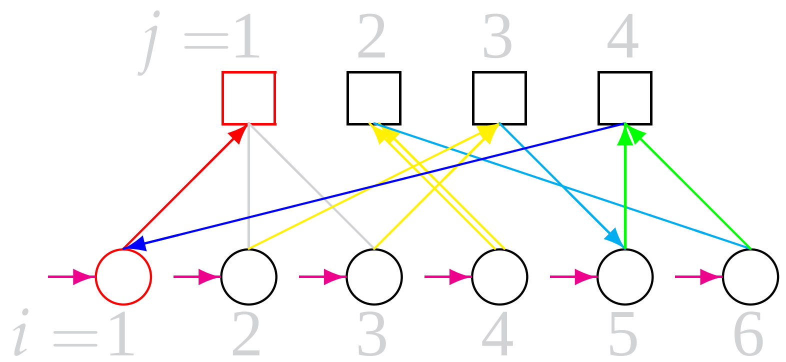


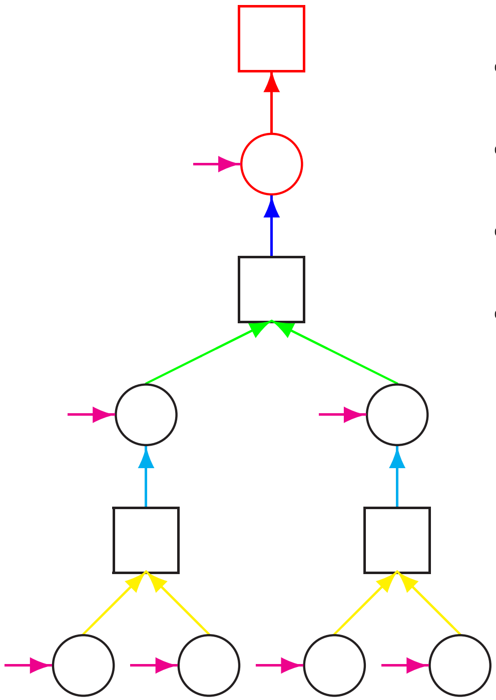


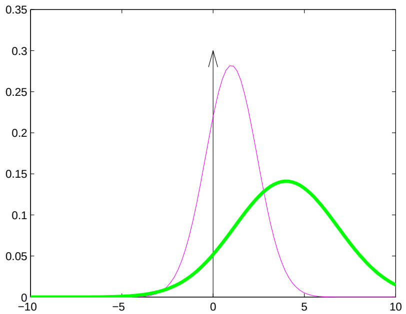


The Density Evolution

Sym. Chs: Assuming $\mathbf{x} = \mathbf{0}$.

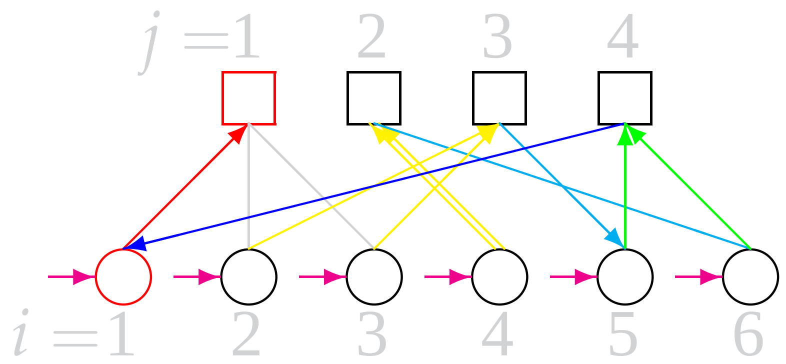


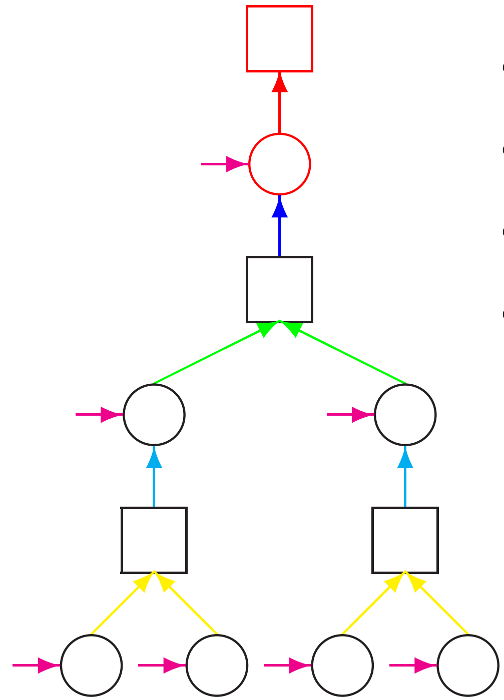


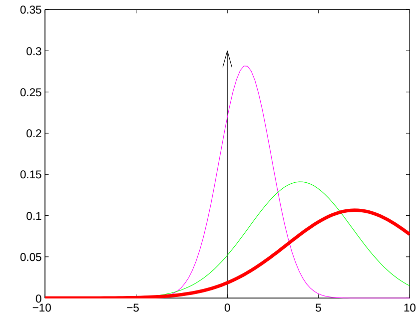


The Density Evolution

Sym. Chs: Assuming $\mathbf{x} = \mathbf{0}$.

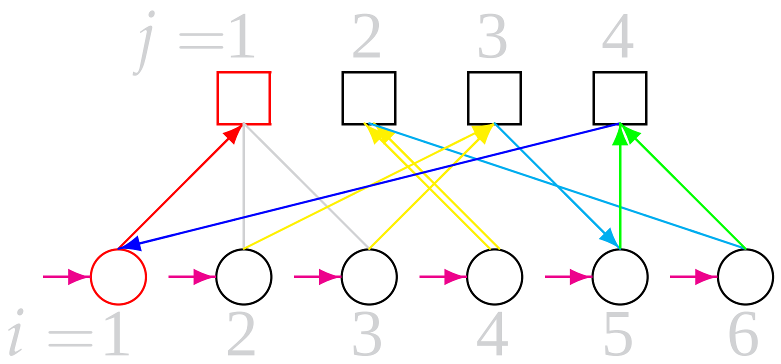


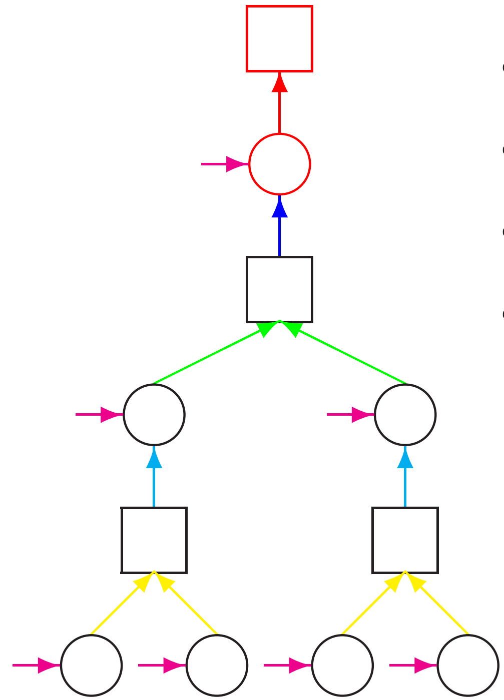


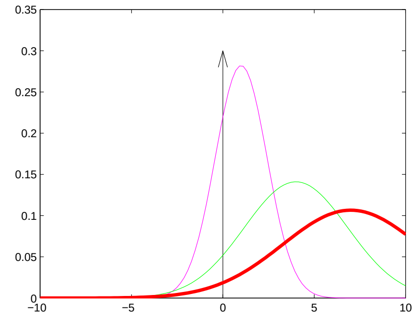


The Density Evolution

Sym. Chs: Assuming $\mathbf{x} = \mathbf{0}$.







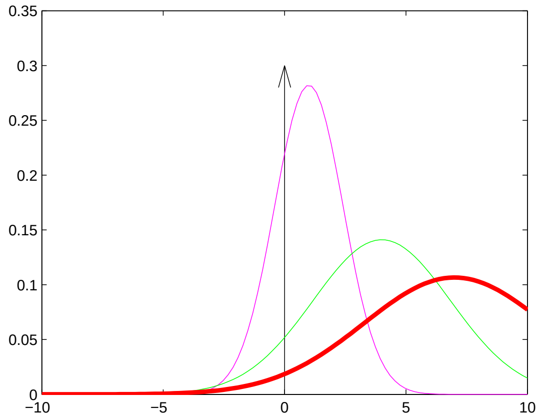
$$
\begin{aligned}
P^{(l)} &= P^{(0)} \otimes \left(Q^{(l-1)}\right)^{\otimes(d_v-1)} \\
Q^{(l-1)} &= \Gamma^{-1}\left(\left(\Gamma\left(P^{(l-1)}\right)\right)^{\otimes(d_c-1)}\right)
\end{aligned}
$$

The Density Evolution (Cont'd)

- [Richardson et al. 01] The DE perfectly describes the decoding behavior for sufficiently large codeword length n.

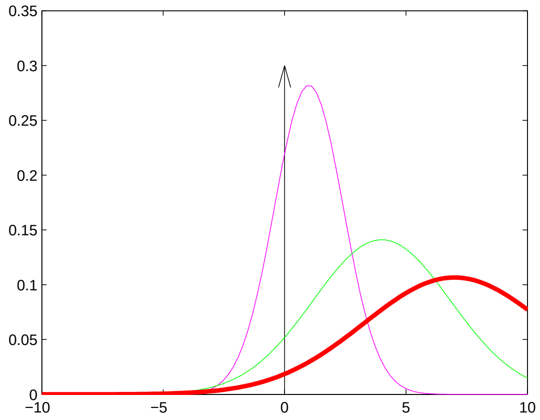
The Density Evolution (Cont'd)

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



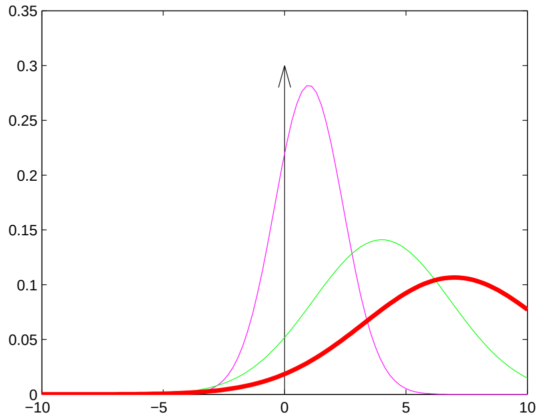
The Density Evolution (Cont'd)

- [Richardson et al. 01] The DE perfectly describes the decoding behavior for sufficiently large codeword length n.
- The tail error probability goes to zero $\Longleftrightarrow$ the channel of interest is decodable.
- In practice, the performance for $n = 10^4 \sim 10^6$ bits is well-predicted.

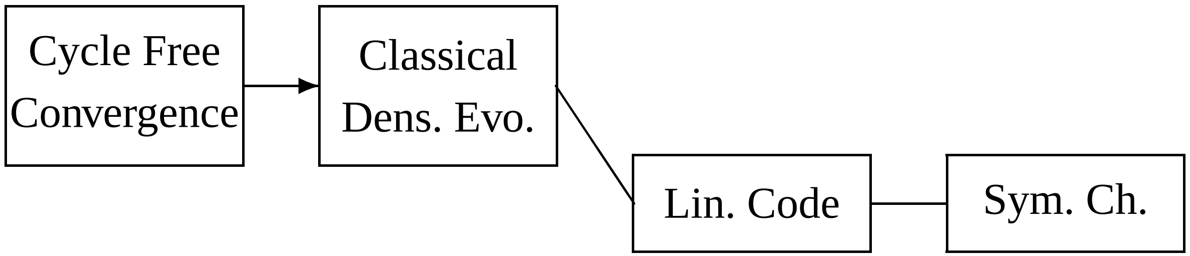


The Density Evolution (Cont'd)

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



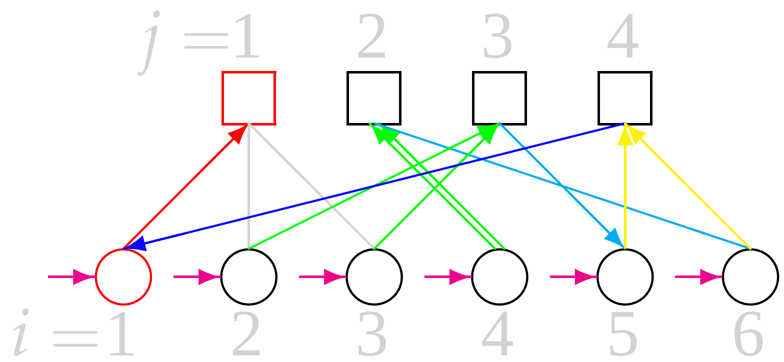
- In practice, the performance for $n = 10^4 \sim 10^6$ bits is well-predicted.
- Advantages:
  - Accurate performance prediction
  - Deterministic, one-time computation instead of random Monte Carlo simulations
  - Can be used as the ultimate performance metric for comparisions/code optimization

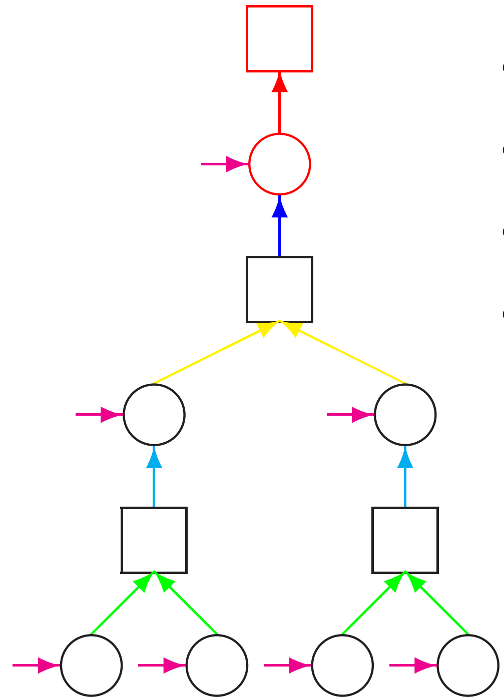


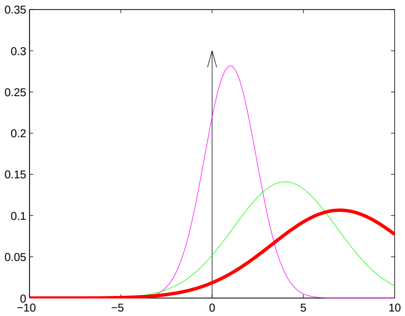
BP Decoder	Optimality	Analysis Tools	Simulation
Cycle-free + Non-Sym. Chs.	✓	✓	optimal
LDPC Codes + Sym. Chs.	?	Dens. Evo.	outstanding
LDPC Codes + Non-Sym. Chs.	?	?	very good

Codeword Dependence of DE

Sym. Chs: Assuming $\mathbf{x} = \mathbf{0}$.





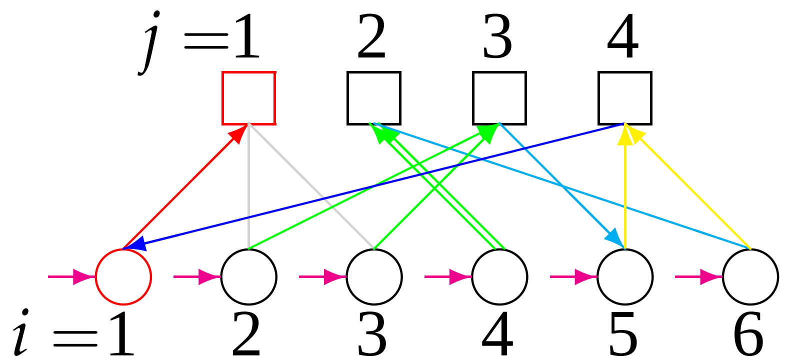


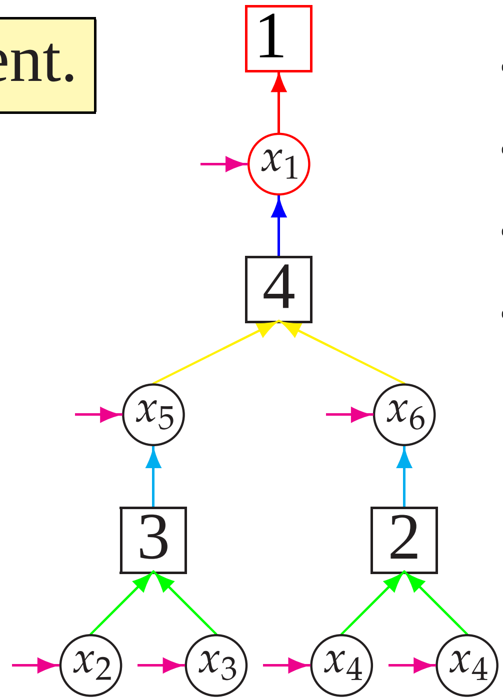
$$P^{(l)} = P^{(0)} \otimes \left(Q^{(l-1)}\right)^{\otimes(d_v-1)}$$

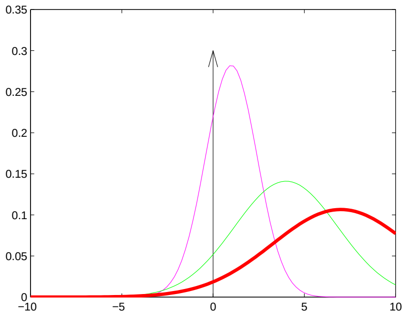
$$Q^{(l-1)} = \Gamma^{-1}\left(\left(\Gamma\left(P^{(l-1)}\right)\right)^{\otimes(d_c-1)}\right)$$

Codeword Dependence of DE

Non-sym. Chs: Codeword-dependent.



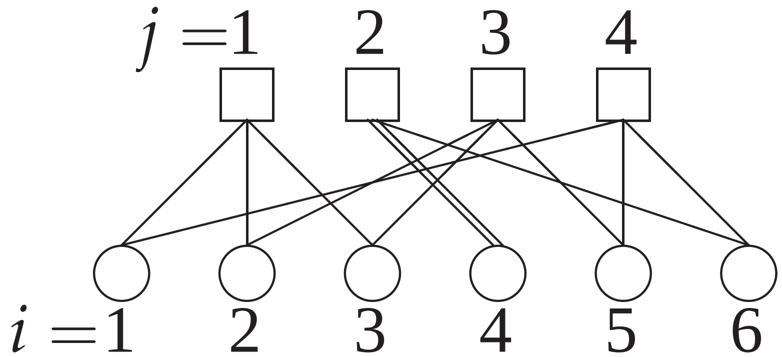


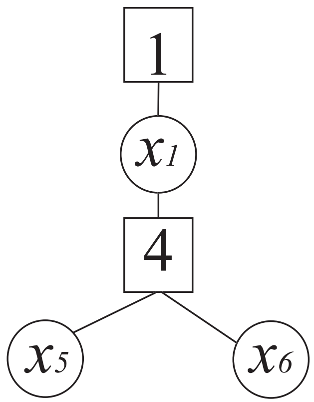


$$P^{(l)}(\mathbf{x}) = P^{(0)}(\mathbf{x}) \otimes \left(Q^{(l-1)}(\mathbf{x})\right)^{\otimes(d_v-1)}$$

$$Q^{(l-1)}(\mathbf{x}) = \Gamma^{-1}\left(\left(\Gamma\left(P^{(l-1)}(\mathbf{x})\right)\right)^{\otimes(d_c-1)}\right)$$

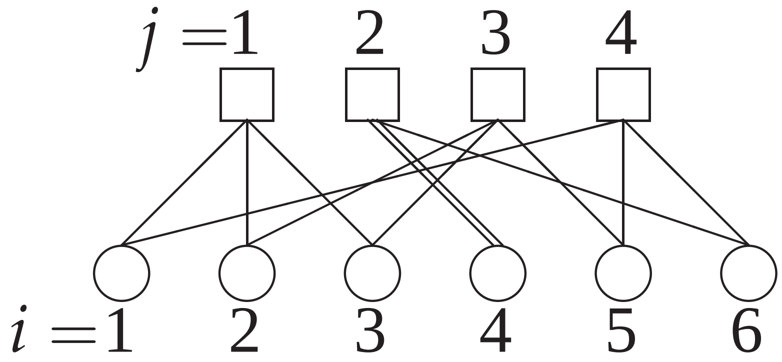
Codeword Averaging?





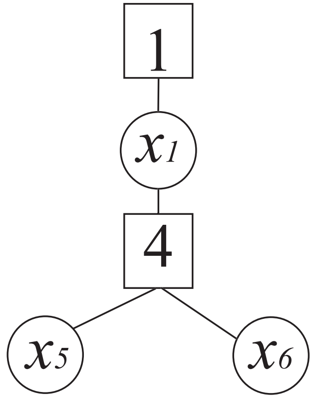
$$\mathbf{X}^1_{(1,1)} := \{x_1x_5x_6 : x_1x_5x_6 = 000, 011, 101, 110\}$$

Codeword Averaging?



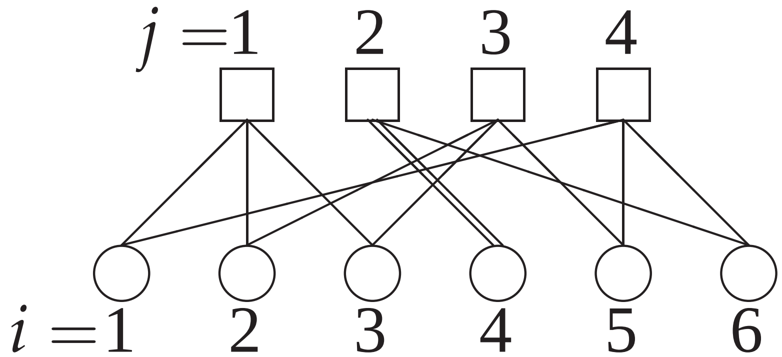
$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$\mathbf{Hx} = \mathbf{0} \Longleftrightarrow x_6 = 0$$



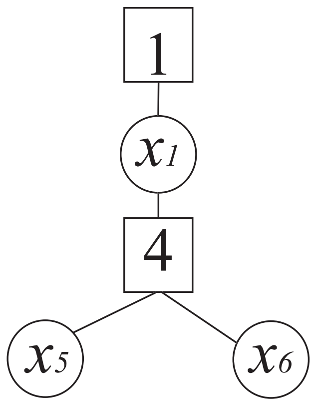
$$\mathbf{X}^1_{(1,1)} := \{x_1x_5x_6 : x_1x_5x_6 = 000, 011, 101, 110\}$$

Codeword Averaging?



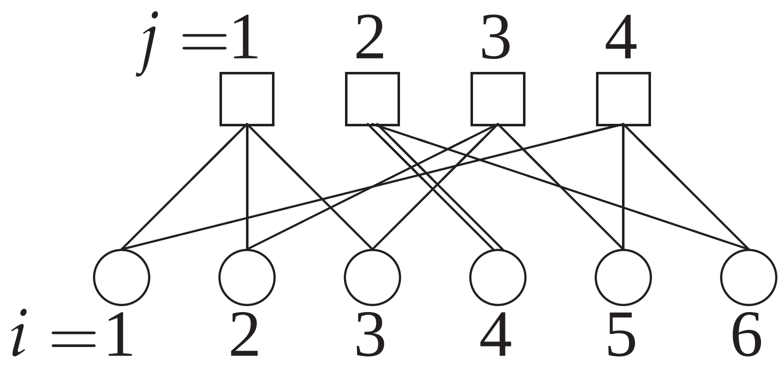
$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$\mathbf{Hx} = \mathbf{0} \Longleftrightarrow x_6 = 0$$



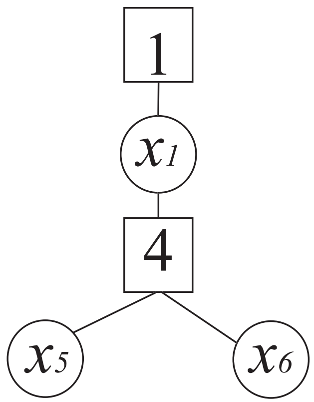
$$\mathbf{X}^1_{(1,1)} := \{x_1x_5x_6 :$$
$$x_1x_5x_6 = 000, 011, 101, 110\}$$

Codeword Averaging?



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

$$\mathbf{Hx} = \mathbf{0} \Longleftrightarrow x_6 = 0$$



$$\mathbf{X}^1_{(1,1)} := \{x_1x_5x_6 : x_1x_5x_6 = 000, 011, 101, 110\}$$

No matter how large the tree is, averaging the trimmed tree code may not be equivalent to averaging the original code.

Perfect Projection Condition

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

$$\frac{\left|\{\mathbf{x}\in\mathbf{X}:\mathbf{x}|_{\text{tree}}=\mathbf{x}_t\}\right|}{|\mathbf{X}|}=\frac{1}{|\mathbf{X}_t|}.$$

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

Perfect Projection Condition

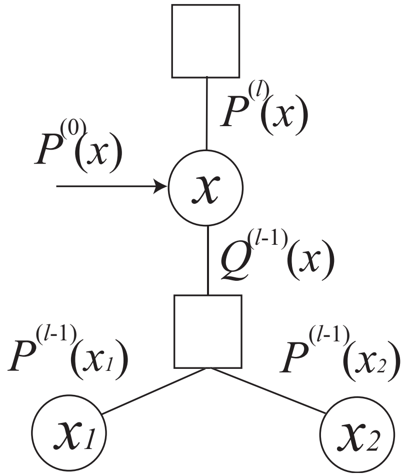
Definition (Perfect Projection) The supporting tree $\mathcal{N}^{2l}$ is perfectly projected, if

$$\frac{\left|\{\mathbf{x}\in\mathbf{X}:\mathbf{x}|_{\text{tree}}=\mathbf{x}_t\}\right|}{|\mathbf{X}|}=\frac{1}{|\mathbf{X}_t|}.$$

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

$$\begin{aligned} P^{(l)}(x) &:= \left\langle P^{(l)}(\mathbf{x})\right\rangle_{\{\mathbf{x}\in\mathbf{X}:\mathbf{x}|_0=x\}} = \left\langle P^{(l)}(\mathbf{x}_t)\right\rangle_{\{\mathbf{x}_t\in\mathbf{X}_t:\mathbf{x}_t|_0=x\}} \\ p_e^{(l)} &= \frac{1}{2}\left(\int_{m=-\infty}^{0} P^{(l)}(0)(dm) + \int_{m=-\infty}^{0} P^{(l)}(1)(dm)\right) \end{aligned}$$

New Iterative Formula for DE



$\because$ Parity-check equation,

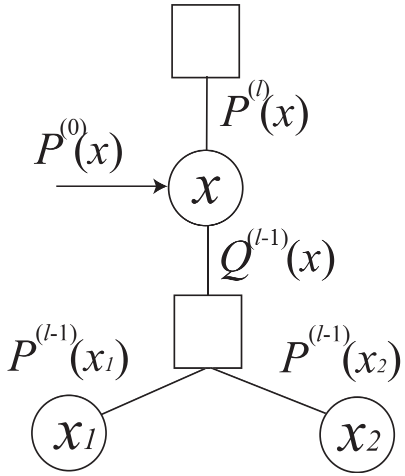
$$x = 0, \;\; x_1 x_2 = 00, 11$$

$$x = 1, \;\; x_1 x_2 = 01, 10$$

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

$$Q^{(l-1)}(x) = \Gamma^{-1}\left(\frac{1}{2^{d_c-2}} \sum_{\mathbf{x}^1 \in \mathbf{X}^1(x)} \bigotimes_{v=1}^{d_c-1} \Gamma\left(P^{(l-1)}(x_v)\right)\right)$$

New Iterative Formula for DE



$\because$ Parity-check equation,

$$x = 0, \quad x_1 x_2 = 00, 11$$

$$x = 1, \quad x_1 x_2 = 01, 10$$

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

$$Q^{(l-1)}(x) = \Gamma^{-1}\left(\frac{1}{2^{d_c-2}} \sum_{\mathbf{x}^1 \in \mathbf{X}^1(x)} \bigotimes_{v=1}^{d_c-1} \Gamma\left(P^{(l-1)}(x_v)\right)\right)$$

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

A simplified expression

Three Theoretical Foundations

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

Three Theoretical Foundations

- Existence of the cycle-free support tree. $\sim$ [Richardson *et al.* 01]
- **Theorem** (<u>Convergence to Perfect Projection in Probability</u>)
  Consider the regular code ensemble $\mathcal{C}^n(d_v, d_c)$ with codeword length n.

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

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

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 $\mathcal{C}^n(d_v, d_c)$ with codeword length n.

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

  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 $\mathcal{C}^n(d_v, d_c)$ with codeword length n.

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

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

  - Graphical properties vs. Algebraic properties
- 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.

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

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

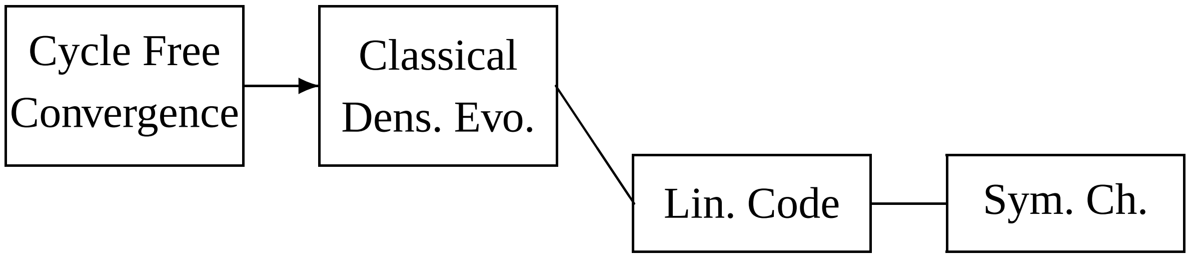
Generalized DE

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

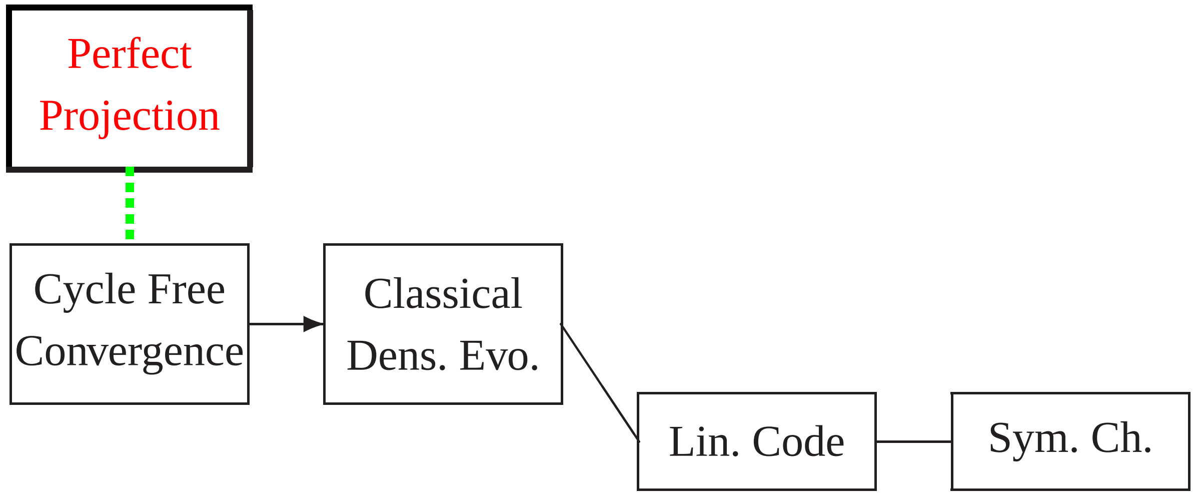
$$\forall x \in \{0,1\}, \quad P^{(l)}(x) = P^{(0)}(x) \otimes \left(Q^{(l-1)}(x)\right)^{\otimes(d_v-1)}$$

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

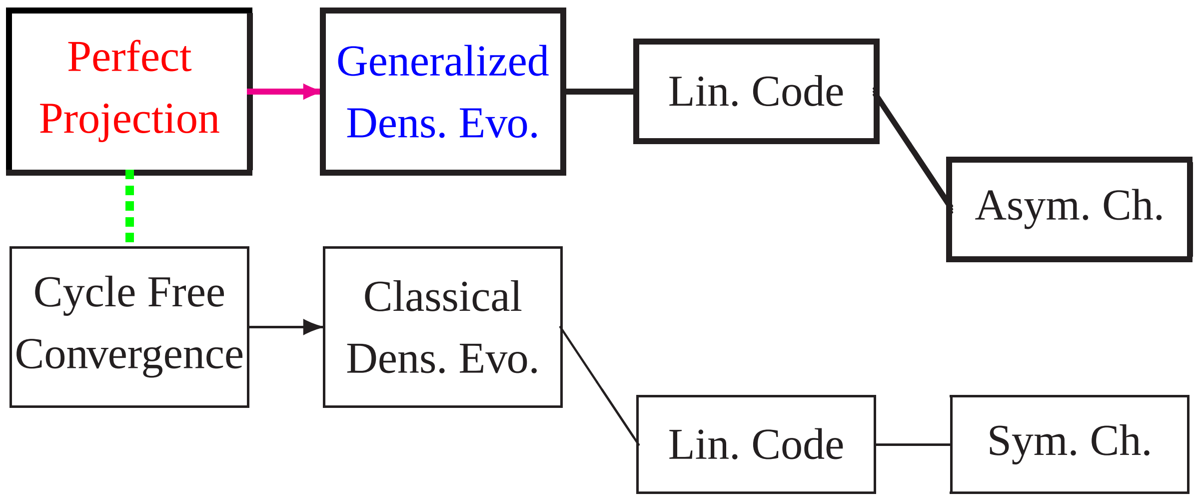
- The same computational complexity as the classical DE.



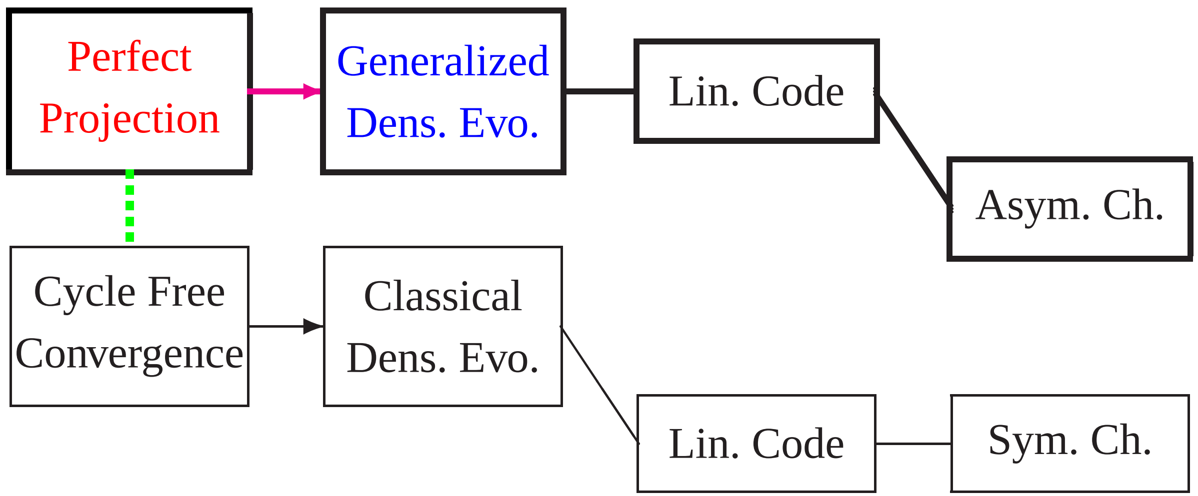
BP Decoder	Optimality	Analysis Tools	Simulation
Cycle-free + Non-Sym. Chs.	✓	✓	optimal
LDPC Codes + Sym. Chs.	?	Dens. Evo.	outstanding
LDPC Codes + Non-Sym. Chs.	?	?	very good



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



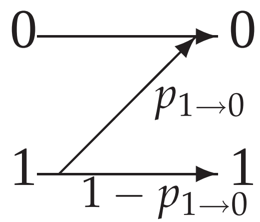
BP Decoder	Optimality	Analysis Tools	Simulation
Cycle-free + Non-Sym. Chs.	✓	✓	optimal
LDPC Codes + Sym. Chs.	?	Dens. Evo.	outstanding
LDPC Codes + Non-Sym. Chs.	?	New Dens. Evo.	very good

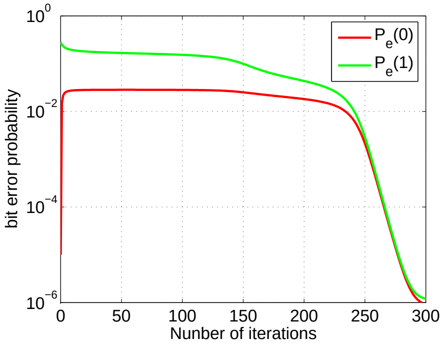


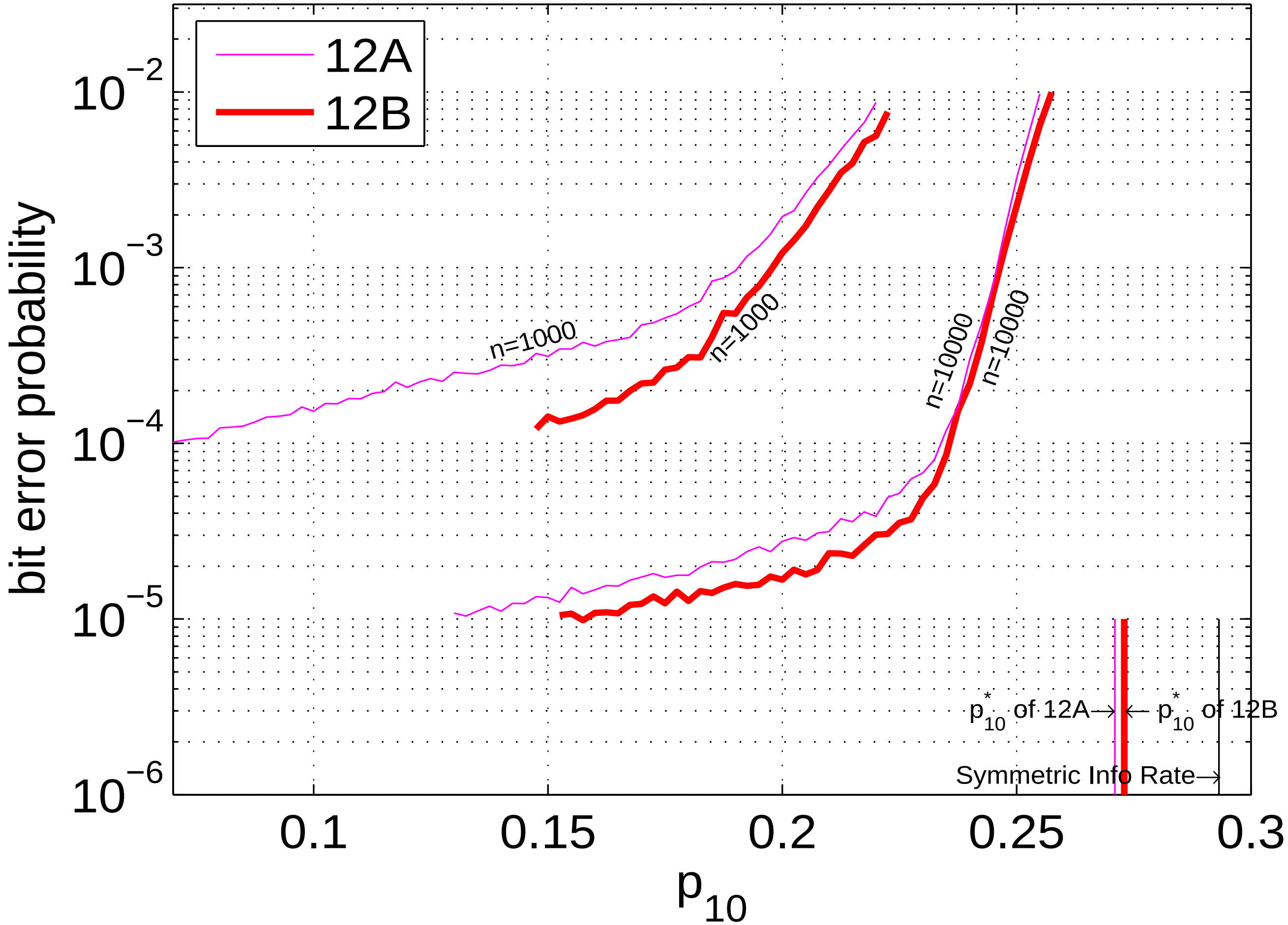
BP Decoder	Optimality	Analysis Tools	Simulation
Cycle-free + Non-Sym. Chs.	✓	✓	optimal
LDPC Codes + Sym. Chs.	?	Dens. Evo.	outstanding
LDPC Codes + Non-Sym. Chs.	?	New Dens. Evo.	very good

Code Optimization

Z-channels





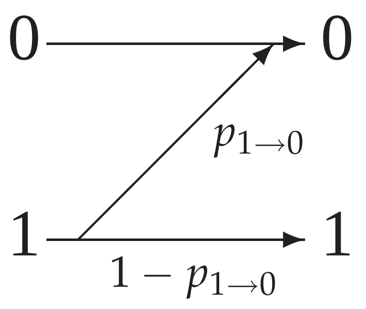


12A: [Richardson01], optimized for BiAWGNC, maxDv=12.

12B: Optimized for z-channels, maxDv=12.

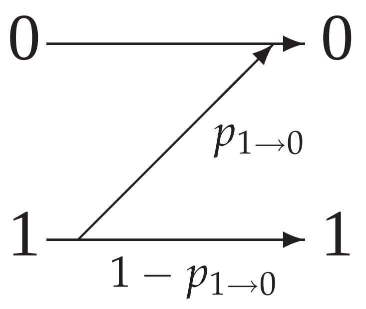
Optimized Rate 1/2 Codes

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



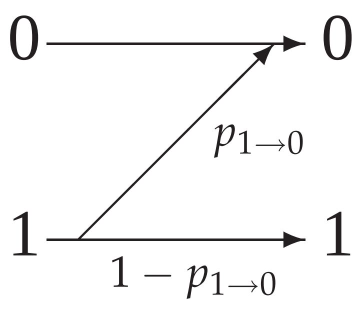
Optimized Rate 1/2 Codes

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



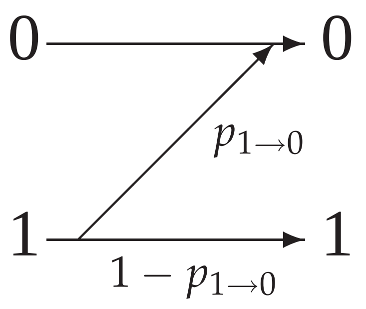
Optimized Rate 1/2 Codes

- Code complexity: $\max d_v = 20$ and $\max d_c = 10$.
  - Allowing 100 iterations: $p^*_{1\to 0} = 0.2741$.
  - Allowing 100 iterations: $p^*_{1\to 0} = 0.2796$.
- Symmetric information rate bound: $p^*_{1\to 0} \leq 0.2932$.
- Away from the capacity bound: $p^*_{1\to 0} \leq 0.3035$.



Optimized Rate 1/2 Codes

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



- Symmetric information rate bound: $p^*_{1\to 0} \leq 0.2932$.
- Away from the capacity bound: $p^*_{1\to 0} \leq 0.3035$.
- 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 $\Longleftrightarrow \lim p_e^{(l)} = 0$.
  - Computationally, how to check $\lim p_e^{(l)} = 0$?

When to Stop DE?

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

When to Stop DE?

- Decodable $\Longleftrightarrow \lim p_e^{(l)} = 0$.
  - Computationally, how to check $\lim p_e^{(l)} = 0$?
- An alternative one dimensional index of $P^{(l)}(x)$:
  - Bhattacharyya Noise Parameter (BNP):
    $000\cdots0$ vs. $111\cdots1$,

When to Stop DE?

- Decodable $\Longleftrightarrow \lim p_e^{(l)} = 0$.
  - Computationally, how to check $\lim p_e^{(l)} = 0$?
- An alternative one dimensional index of $P^{(l)}(x)$:
  - Bhattacharyya Noise Parameter (BNP):
    $000\cdots0$ vs. $111\cdots1$, $\Longrightarrow p_{e,n} \approx \mathrm{BNP}^n$

When to Stop DE?

- Decodable $\Longleftrightarrow \lim p_e^{(l)} = 0$.
  - Computationally, how to check $\lim p_e^{(l)} = 0$?
- An alternative one dimensional index of $P^{(l)}(x)$:
  - Bhattacharyya Noise Parameter (BNP):
    $000\cdots0$ vs. $111\cdots1$, $\Longrightarrow p_{e,n} \approx \mathrm{BNP}^n$
  - $\mathrm{BNP}^{(l)} := \int e^{-m/2} \langle P^{(l)} \rangle (dm)$.

When to Stop DE?

- Decodable $\Longleftrightarrow \lim p_e^{(l)} = 0$.
  - 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. $111\cdots 1$, $\Longrightarrow p_{e,n} \approx \text{BNP}^n$
  - $\text{BNP}^{(l)} := \int e^{-m/2} \langle P^{(l)} \rangle (dm)$.
  - **Lemma:** $2p_e^{(l)} \leq \text{BNP}^{(l)} \leq 2\sqrt{p_e^{(l)}(1-p_e^{(l)})}$, even for non-symmetric channels.

When to Stop DE?

- Decodable $\Longleftrightarrow \lim p_e^{(l)} = 0$.
  - Computationally, how to check $\lim p_e^{(l)} = 0$?
- An alternative one dimensional index of $P^{(l)}(x)$:
  - Bhattacharyya Noise Parameter (BNP):
    $000\cdots0$ vs. $111\cdots1$, $\Longrightarrow p_{e,n} \approx \mathrm{BNP}^n$
  - $\mathrm{BNP}^{(l)} := \int e^{-m/2}\langle P^{(l)}\rangle(dm)$.
  - **Lemma:** $2p_e^{(l)} \leq \mathrm{BNP}^{(l)} \leq 2\sqrt{p_e^{(l)}(1-p_e^{(l)})}$,
    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

if $\text{BNP}^{(0)} \geq \frac{1}{\lambda_2 \rho'(1)}$ **then**

    $\lim_{l \to \infty} p_e^{(l)} > 0 \Longleftrightarrow$ Undecodable.

else

    repeat

        Perform DE iteration

    until $\text{BNP}^{(l)} < \epsilon^*$

    $\lim_{l \to} p_e^{(l)} = 0 \Longleftrightarrow$ Decodable.

end if

A Stopping Criterion

1: **if** $\text{BNP}^{(0)} \geq \frac{1}{\lambda_2 \rho'(1)}$ **then**

2: $\lim_{l\to\infty} p_e^{(l)} > 0 \Longleftrightarrow$ Undecodable.

3: **else**

4: **repeat**

5: Perform DE iteration

6: **until** $\text{BNP}^{(l)} < \epsilon^*$

7: $\lim_{l\to} p_e^{(l)} = 0 \Longleftrightarrow$ 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

if $\text{BNP}^{(0)} \geq \frac{1}{\lambda_2 \rho'(1)}$ **then**
    $\lim_{l \to \infty} p_e^{(l)} > 0 \Longleftrightarrow$ Undecodable.
else
    repeat
        Perform DE iteration
    until $\text{BNP}^{(l)} < \epsilon^*$
    $\lim_{l \to} p_e^{(l)} = 0 \Longleftrightarrow$ Decodable.
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

```
if $\mathrm{BNP}^{(0)} \geq \frac{1}{\lambda_2 \rho'(1)}$ then
    $\lim_{l\to\infty} p_e^{(l)} > 0 \iff$ Undecodable.
else
    repeat
        Perform DE iteration
    until $\mathrm{BNP}^{(l)} < \epsilon^*$
    $\lim_{l\to} p_e^{(l)} = 0 \iff$ Decodable.
end if
```

Stability Conditions

Example: Code 12B

$$\frac{1}{\lambda_2 \rho'(1)} = 0.6430$$

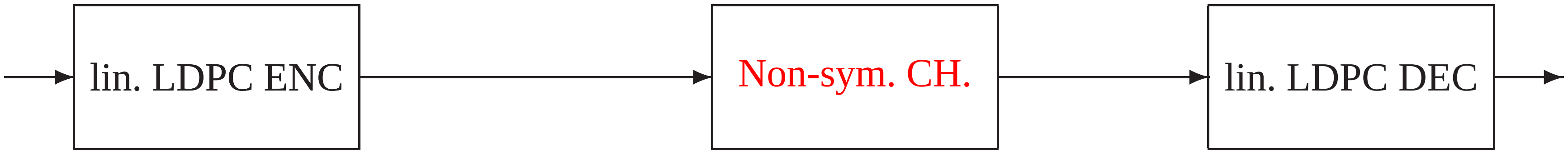
$$\mathrm{BNP}^{(0)} = 0.5253$$

$$\epsilon^* = 0.0675$$

C.-C. Wang, S.R. Kulkarni, and H.V. Poor, "Density Evolution for Asymmetric Memoryless Channels," to appear in *IEEE Transactions on Information Theory*.

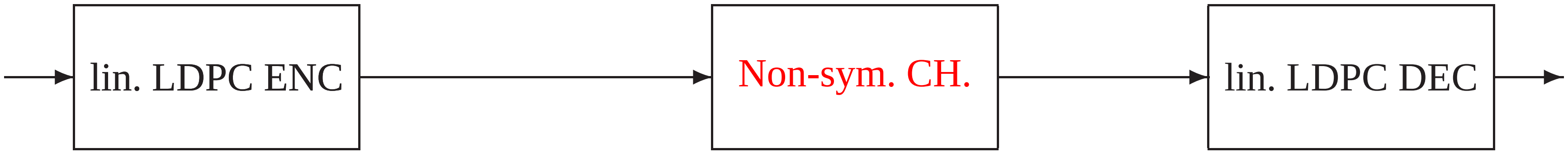
LDPC Coset Code Ensemble

Non-symmetric channels:

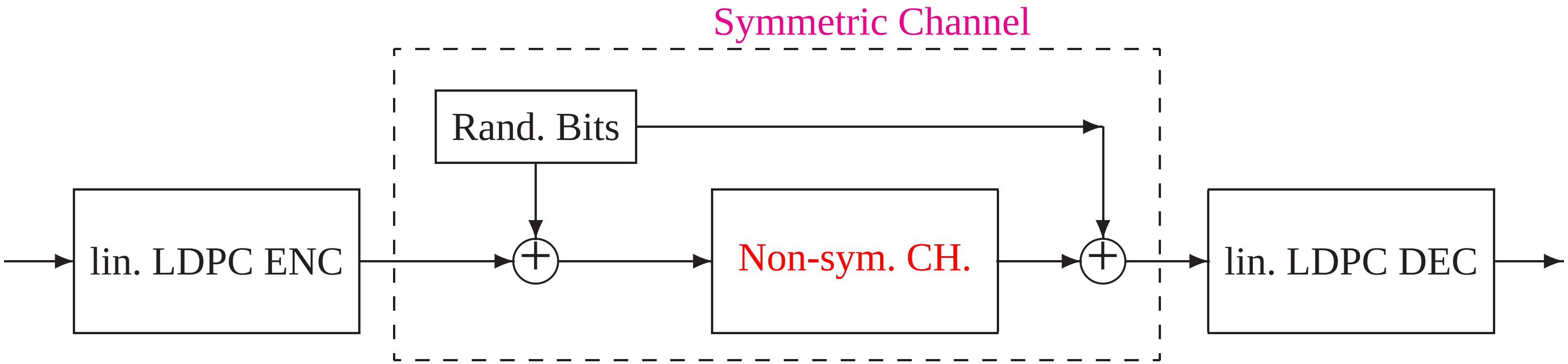


LDPC Coset Code Ensemble

Non-symmetric channels:

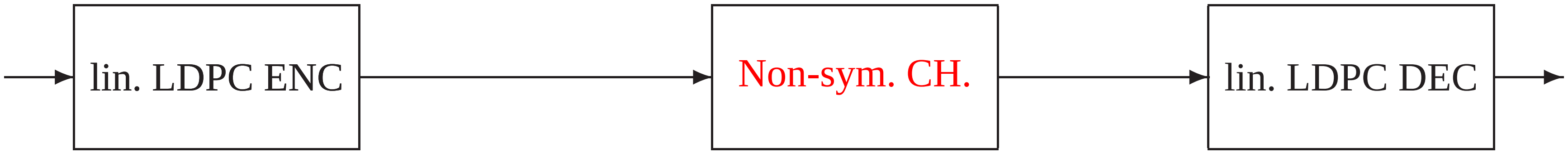


Symmetrized channels:

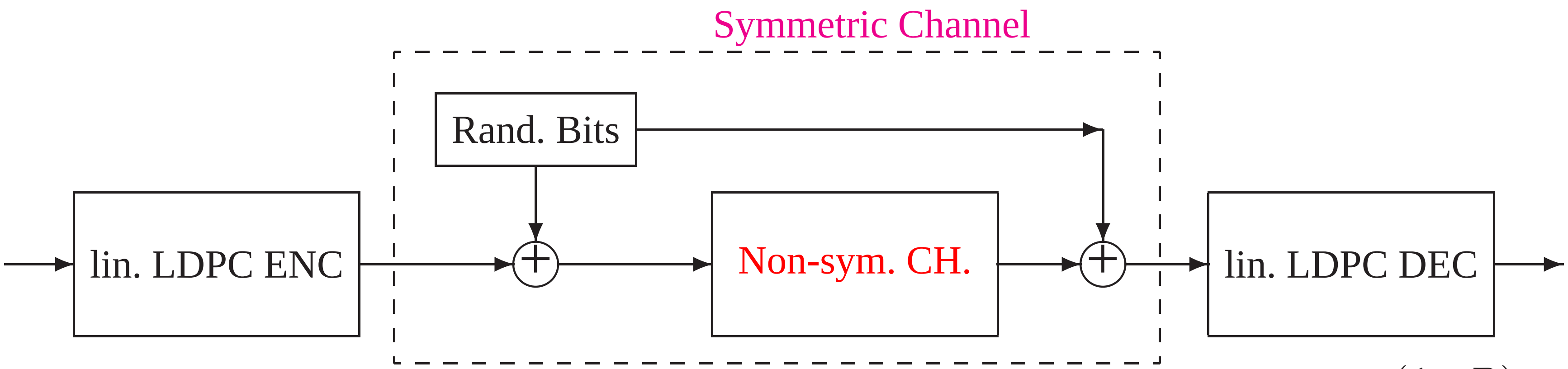


LDPC Coset Code Ensemble

Non-symmetric channels:



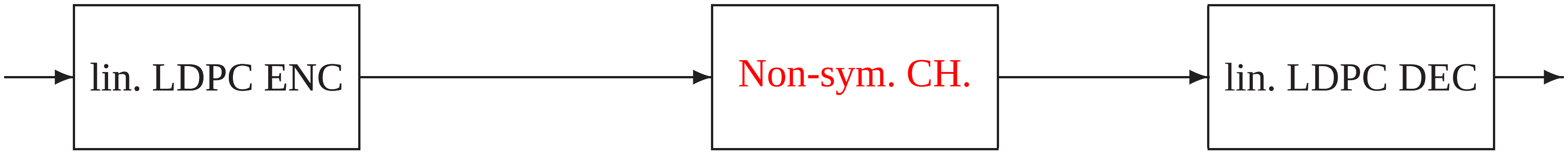
Symmetrized channels:



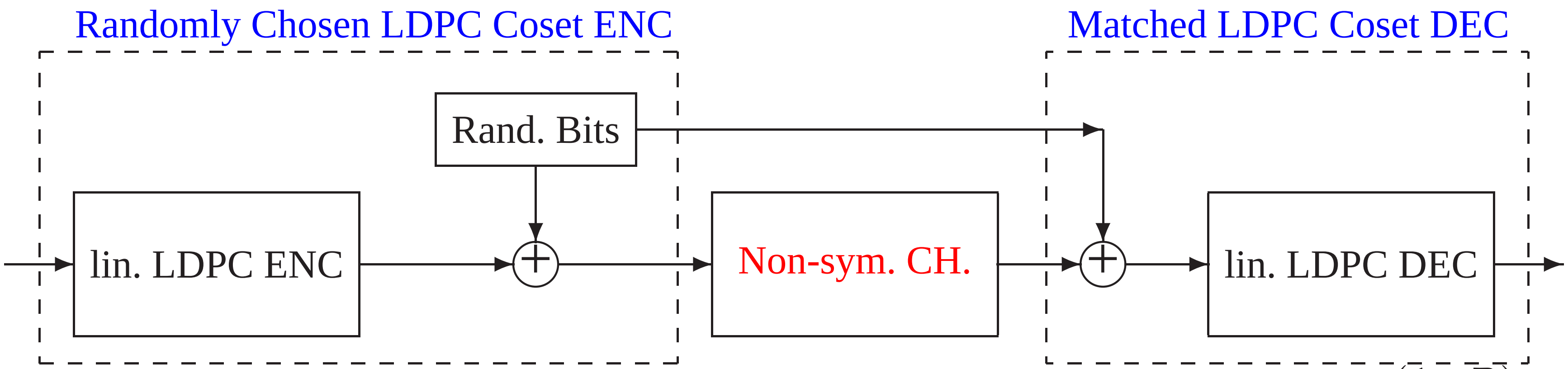
LDPC coset code ensemble: $\mathbf{H}\mathbf{x} = \mathbf{s}$ and $\mathbf{s} \in_{\text{rand.}} \{0,1\}^{n(1-R)}$.

LDPC Coset Code Ensemble

Non-symmetric channels:



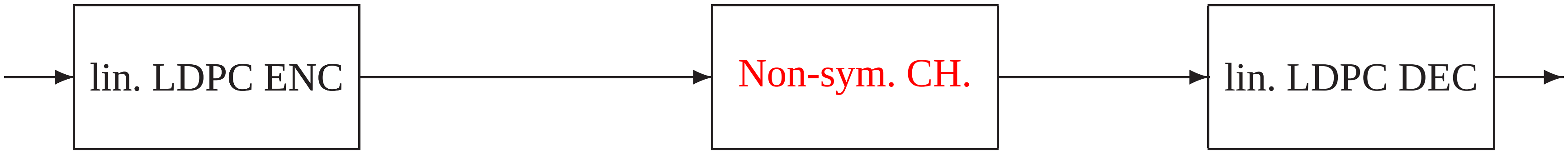
Symmetrized channels:



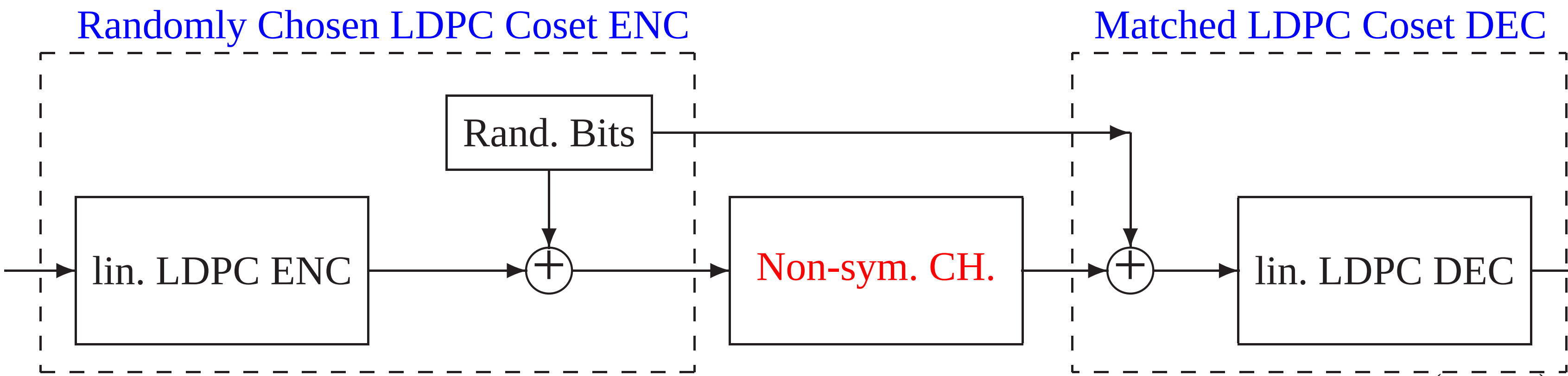
LDPC coset code ensemble: $\mathbf{Hx} = \mathbf{s}$ and $\mathbf{s} \in_{\text{rand.}} \{0,1\}^{n(1-R)}$.

LDPC Coset Code Ensemble

Non-symmetric channels:



Symmetrized channels:

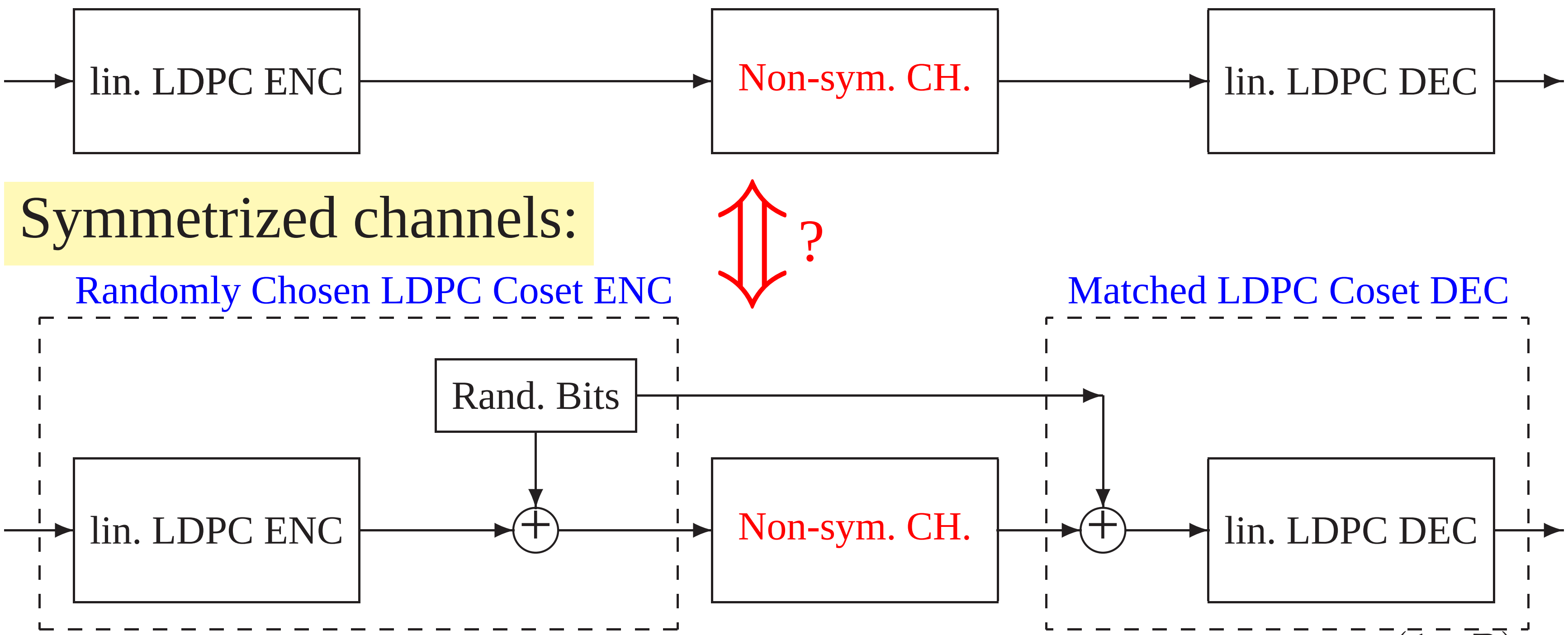


LDPC coset code ensemble: $\mathbf{Hx} = \mathbf{s}$ and $\mathbf{s} \in_{\text{rand.}} \{0, 1\}^{n(1-R)}$.

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

LDPC Coset Code Ensemble

Non-symmetric channels:



Symmetrized channels:

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

- 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.
- $\lim_{d_{c,min}\to\infty} \left\langle P^{(l_0)}_{linear} \right\rangle \stackrel{\mathcal{D}}{=} P^{(l_0)}_{coset}$ 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.
- $\lim_{d_{c,min}\to\infty} \left\langle P^{(l_0)}_{linear} \right\rangle \stackrel{\mathcal{D}}{=} P^{(l_0)}_{coset}$ exponentially
- $\lim_{d_{c,min}\to\infty} \left| p^*_{1\to 0,linear} - p^*_{1\to 0,coset} \right| = 0$

The Answer

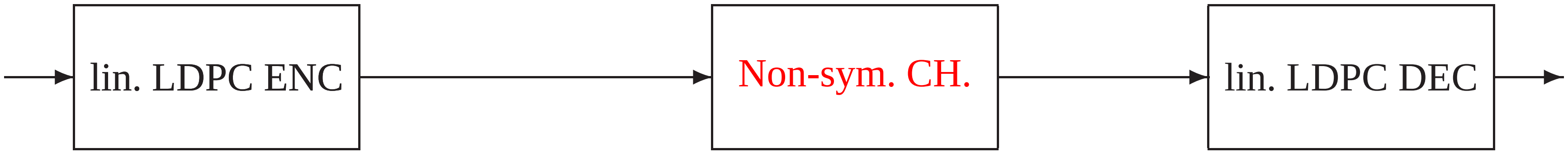
With the help of our new DE,

- 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.
- $\lim_{d_{c,min}\to\infty} \left\langle P^{(l_0)}_{linear} \right\rangle \stackrel{\mathcal{D}}{=} P^{(l_0)}_{coset}$ exponentially
- $\lim_{d_{c,min}\to\infty} \left| p^*_{1\to 0,linear} - p^*_{1\to 0,coset} \right| = 0$

C.-C. Wang, S.R. Kulkarni, and H.V. Poor, "Typicality of Linear Codes among the LDPC Coset Code Ensemble,"
in *Proc. 39th Conference on Information Sciences and Systems*, Baltimore, MD, 2005.

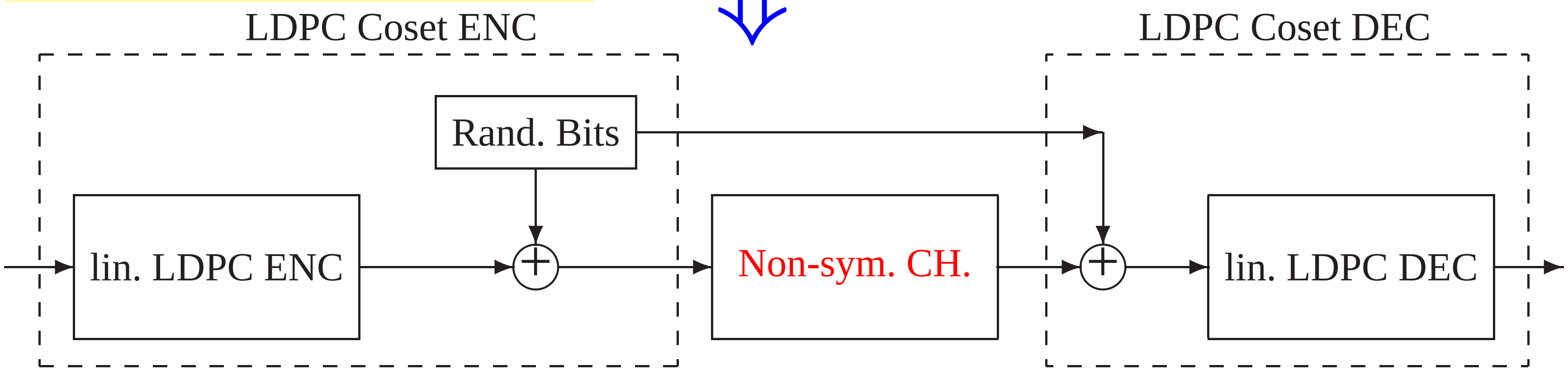
Applications

Non-symmetric channels:



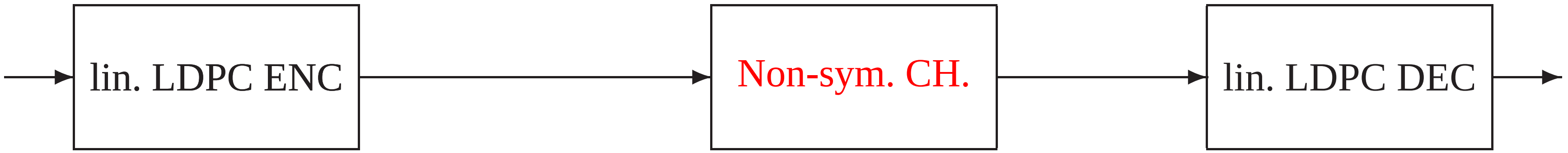
Symmetrized channels:

⇕ almost equivalent for moderate d_c



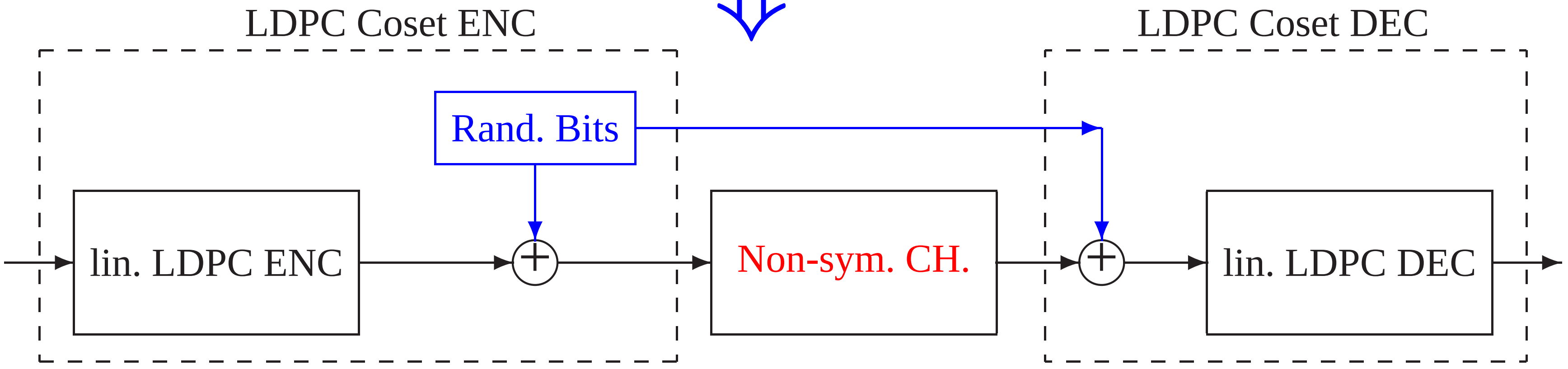
Applications

Non-symmetric channels:



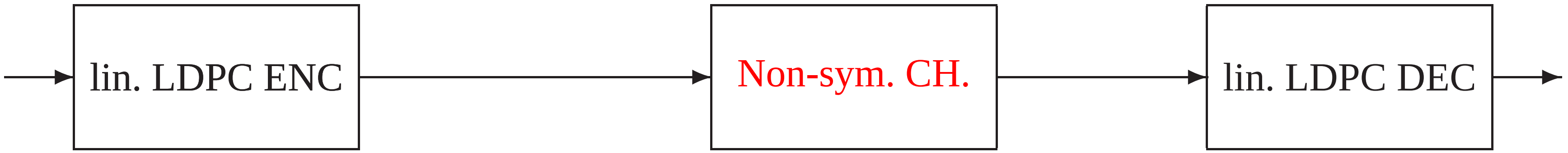
Symmetrized channels:

$\Updownarrow$ almost equivalent for moderate d_c



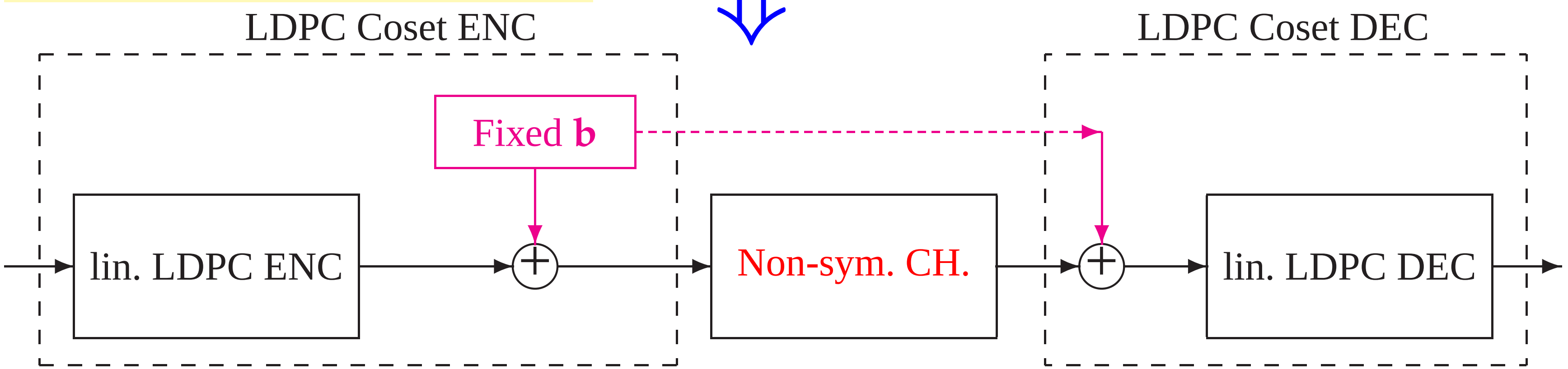
Applications

Non-symmetric channels:



Symmetrized channels:

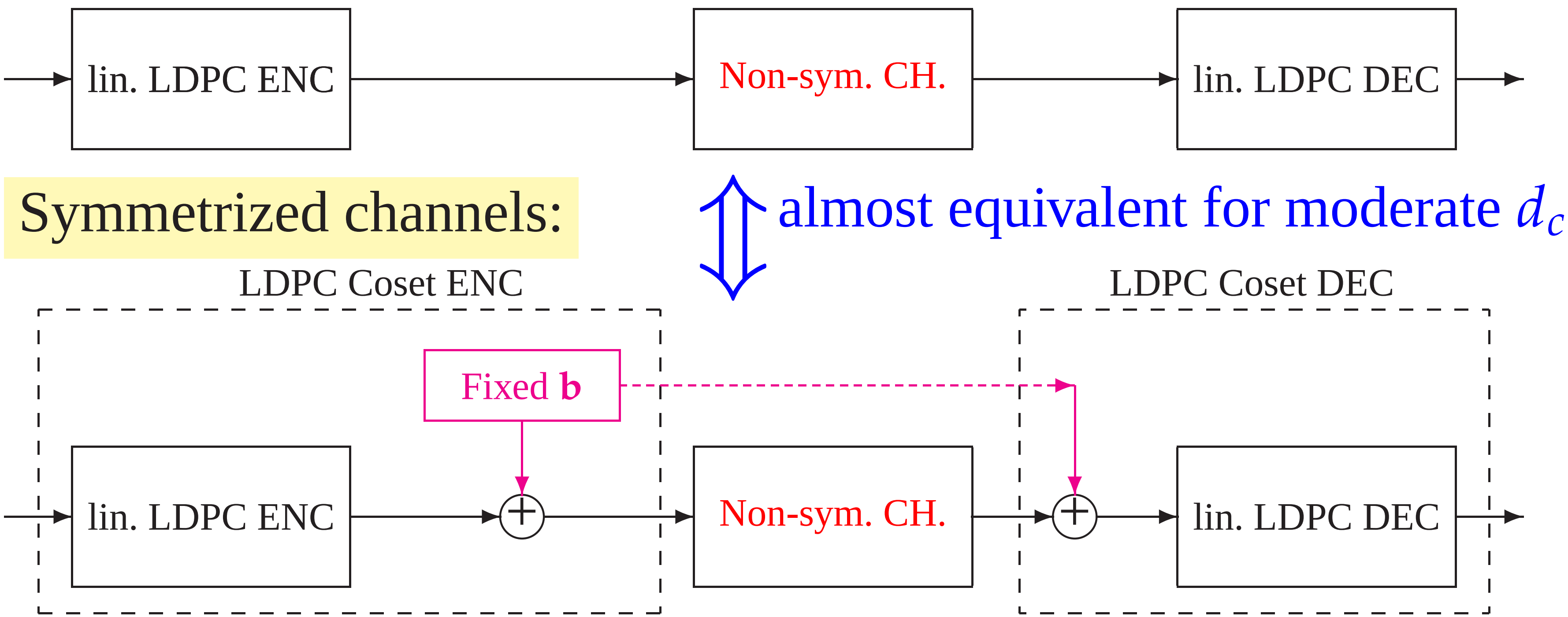
$\Updownarrow$ almost equivalent for moderate d_c



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

Applications

Non-symmetric channels:

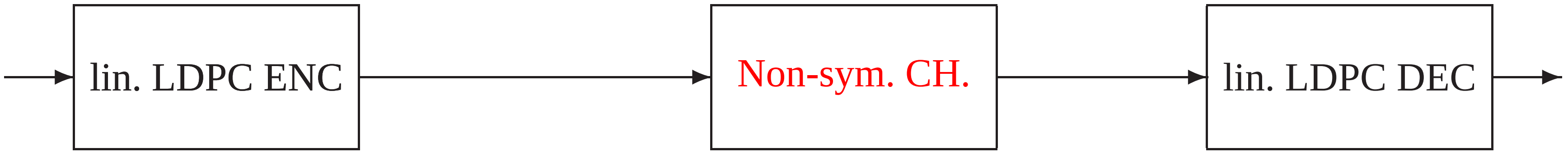


Symmetrized channels:

- [Kavčić 03]: Almost all **b** are typical.
- **Applications:** Analysis — EXIT chart/fast DE,

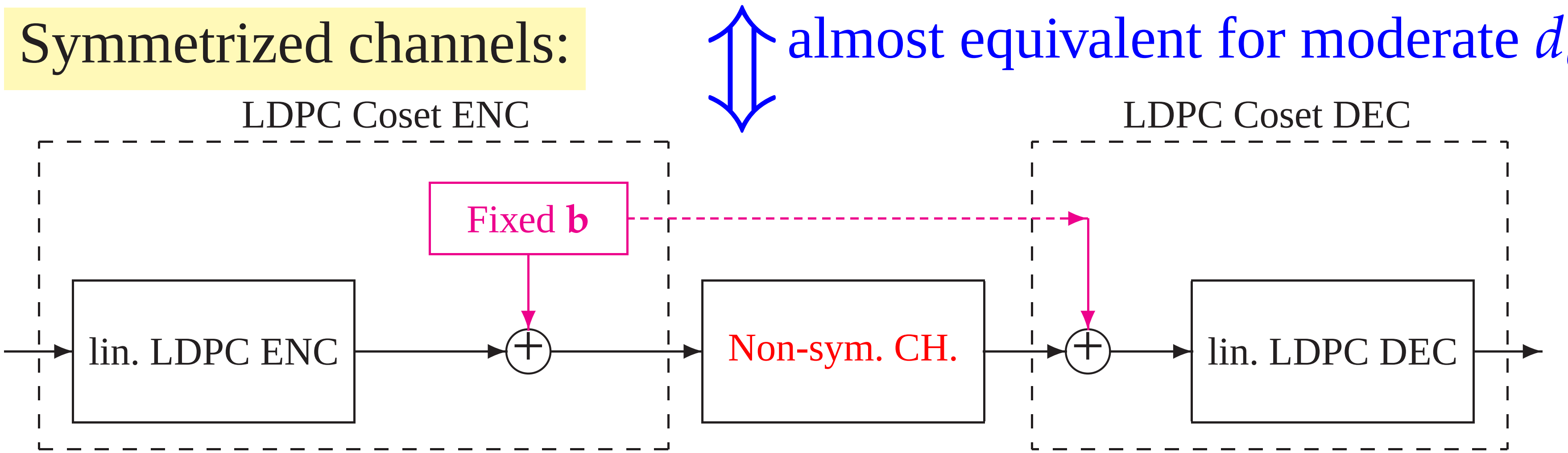
Applications

Non-symmetric channels:



Symmetrized channels:

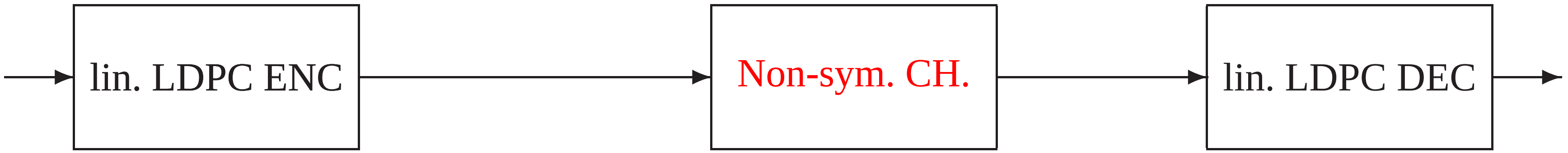
almost equivalent for moderate d_c



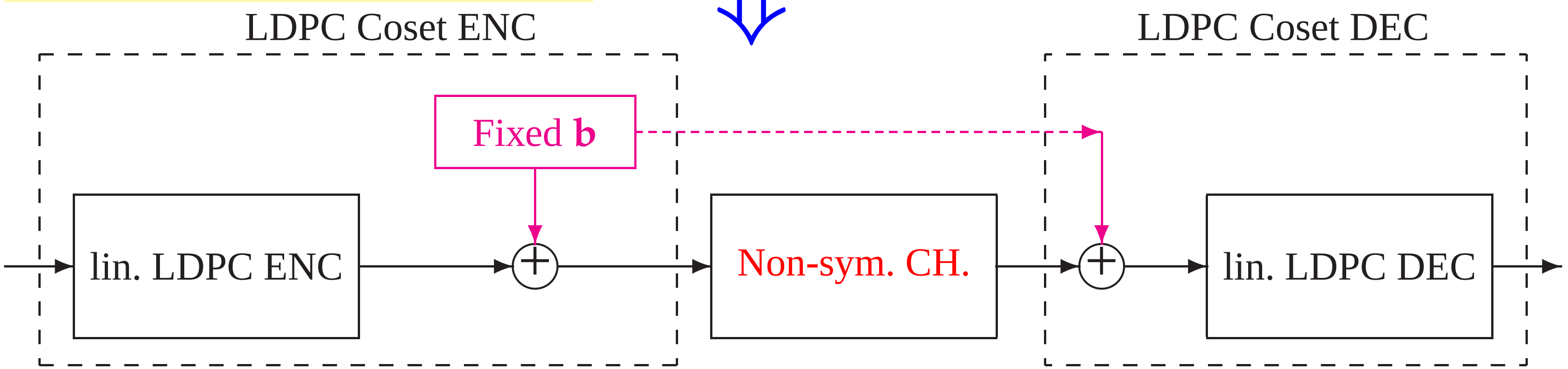
- [Kavčić 03]: Almost all **b** are typical.
- **Applications:** Analysis — EXIT chart/fast DE, Simulations — using the all-zero codeword,

Applications

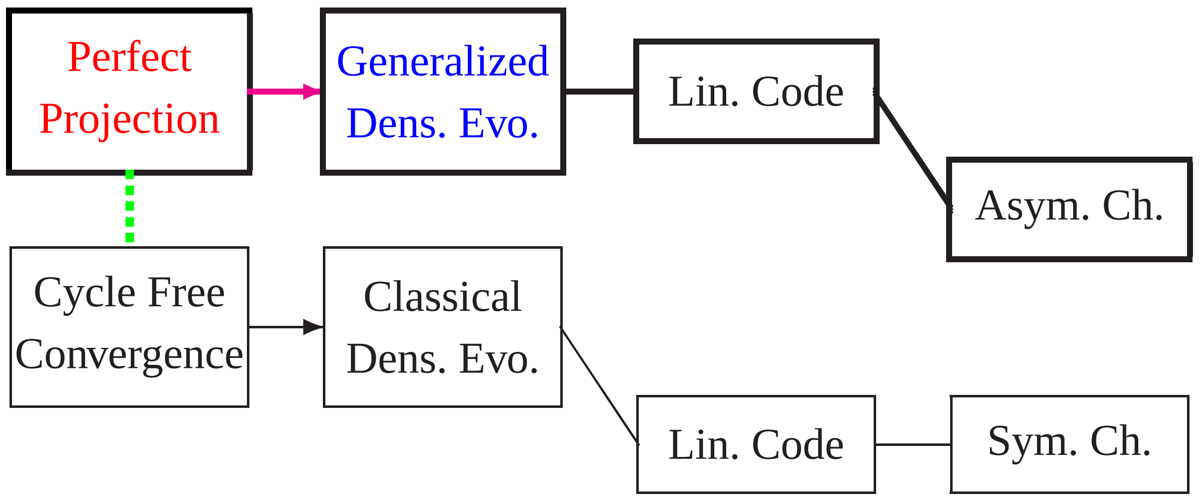
Non-symmetric channels:

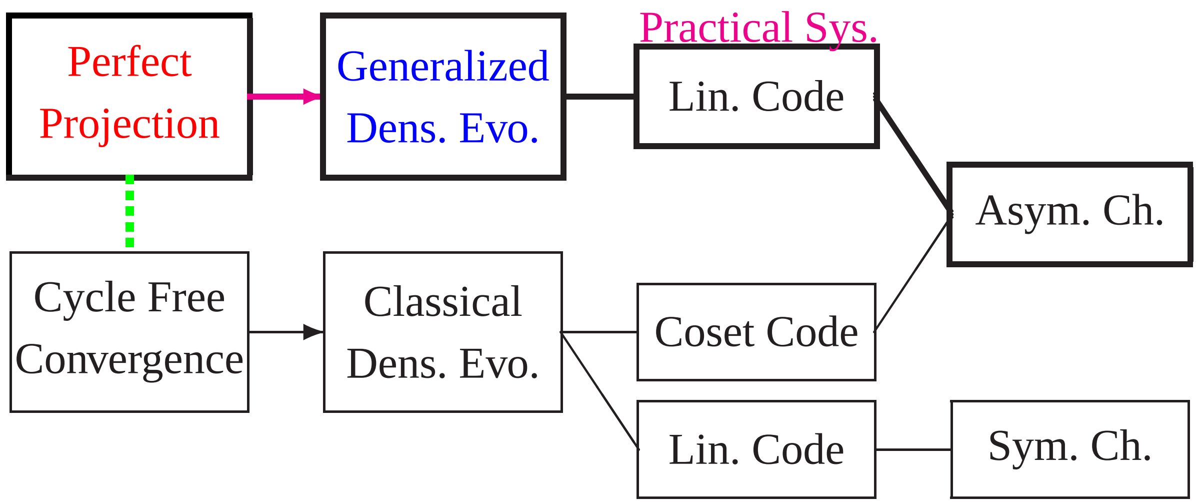


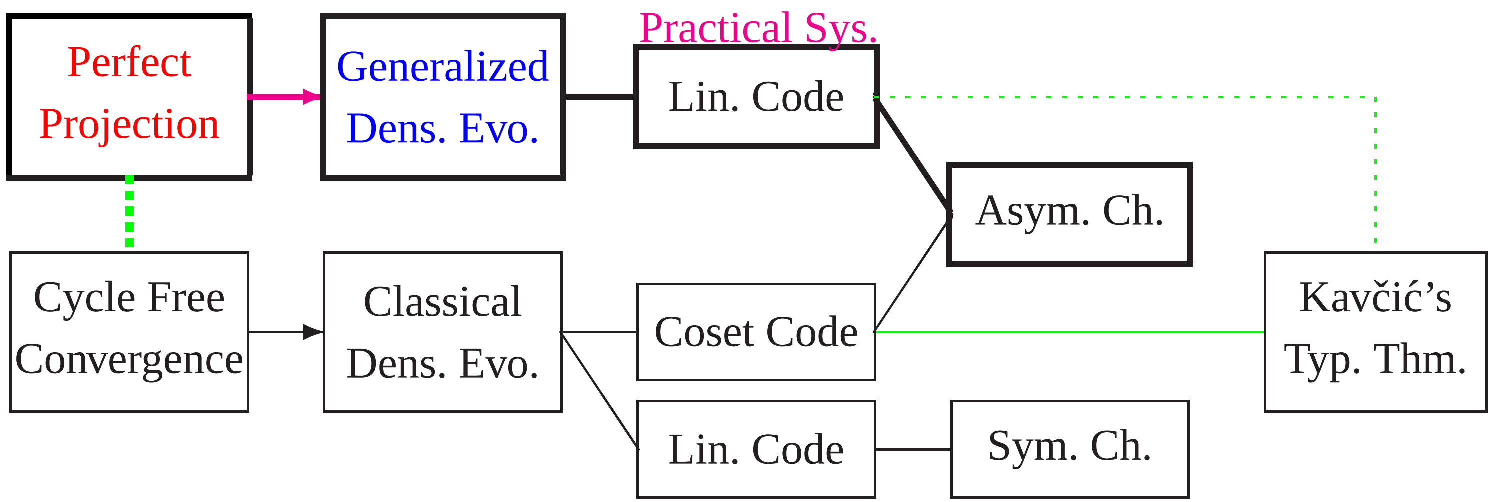
Symmetrized channels: $\Updownarrow$ almost equivalent for moderate d_c

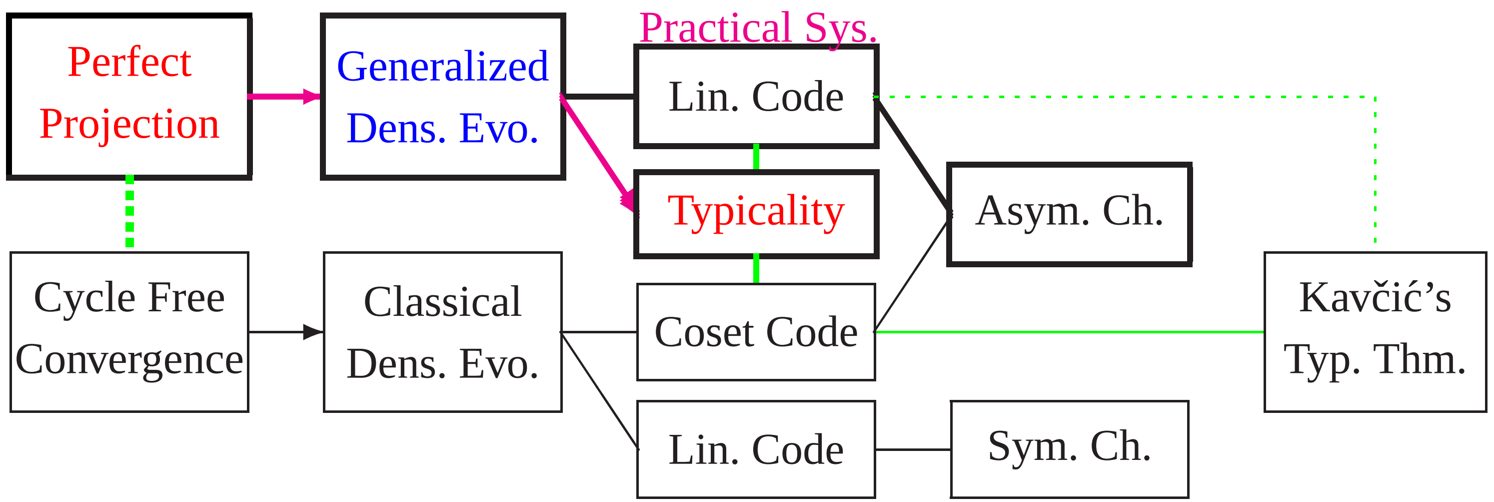


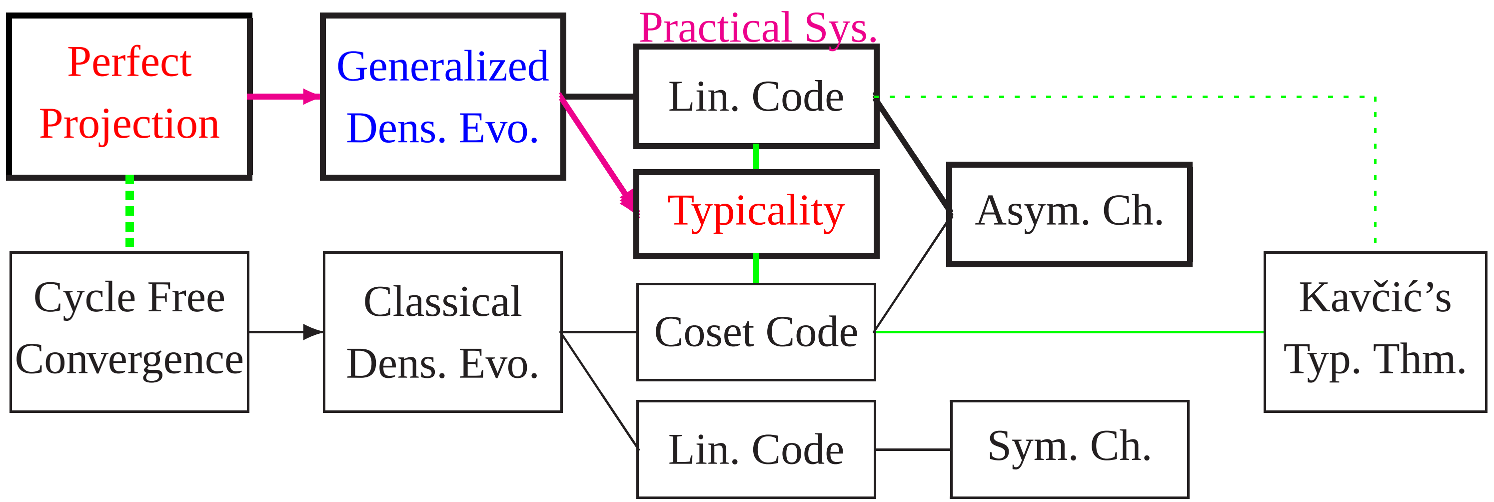
- [Kavčić 03]: Almost all **b** are typical.
- **Applications:** Analysis — EXIT chart/fast DE, Simulations — using the all-zero codeword, & Implementation — linear codes.


Perfect Projection
Generalized Dens. Evo.
Lin. Code
Asym. Ch.
Cycle Free Convergence
Classical Dens. Evo.
Lin. Code
Sym. Ch.


Perfect Projection
Generalized Dens. Evo.
Practical Sys.
Lin. Code
Asym. Ch.
Cycle Free Convergence
Classical Dens. Evo.
Coset Code
Lin. Code
Sym. Ch.


Perfect Projection
Generalized Dens. Evo.
Practical Sys.
Lin. Code
Asym. Ch.
Cycle Free Convergence
Classical Dens. Evo.
Coset Code
Kavčić's Typ. Thm.
Lin. Code
Sym. Ch.


Practical Sys.
Perfect Projection
Generalized Dens. Evo.
Lin. Code
Typicality
Asym. Ch.
Cycle Free Convergence
Classical Dens. Evo.
Coset Code
Kavčić's Typ. Thm.
Lin. Code
Sym. Ch.



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

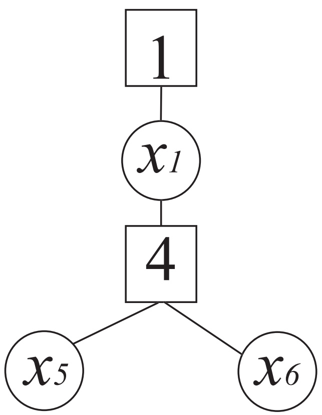
- BP
- Exc ... es.
- Bet

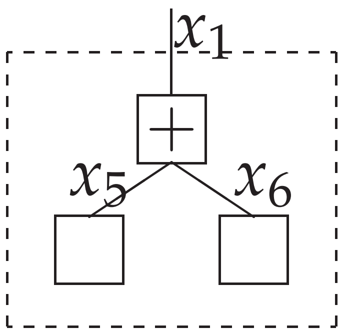
Is BP locally optimal? Namely, given local observations on $\mathcal{N}^{(2l)}$ and complete knowledge of the codebook, can a maximum *a posteriori* probability (MAP) decoder do better?

Local Optimality of BP

- BP
- Exc... es.
- Bet

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?



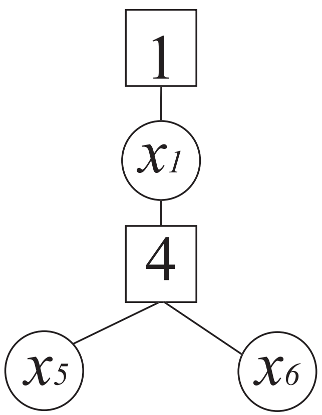


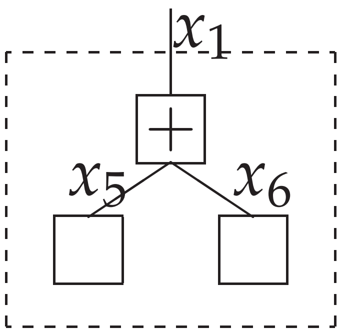
BP Decoder

Local Optimality of BP

- BP
- Exc … es.
- Bet

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

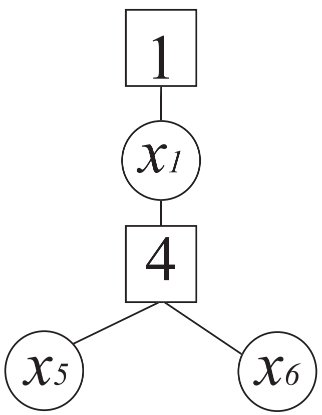
$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

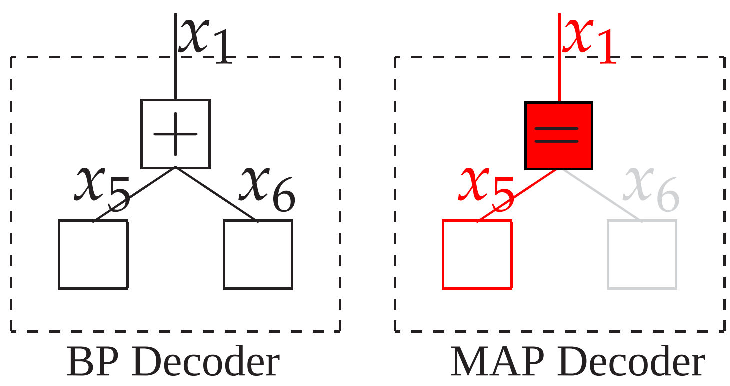
$$\mathbf{H}\mathbf{x} = \mathbf{0} \Longleftrightarrow x_6 = 0$$

Local Optimality of BP

- BP
- Exc … es.
- Bet

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?





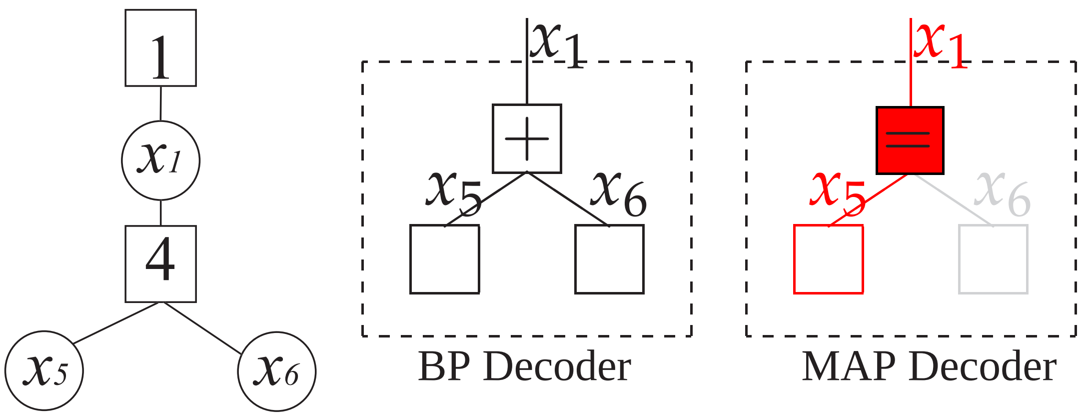
$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$\mathbf{Hx} = \mathbf{0} \Longleftrightarrow x_6 = 0$$

Local Optimality of BP

- BP
- Exc… es.
- Bet

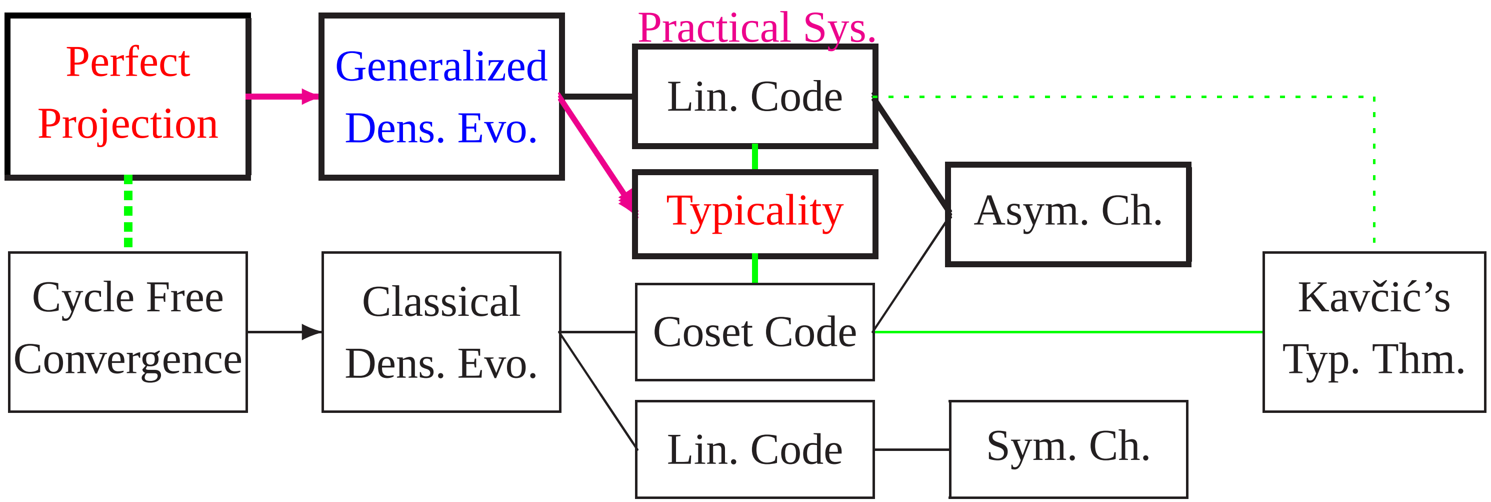
Is BP locally optimal? Namely, given local observations on $\mathcal{N}^{(2l)}$ and complete knowledge of the codebook, can a maximum *a posteriori* probability (MAP) decoder do better?



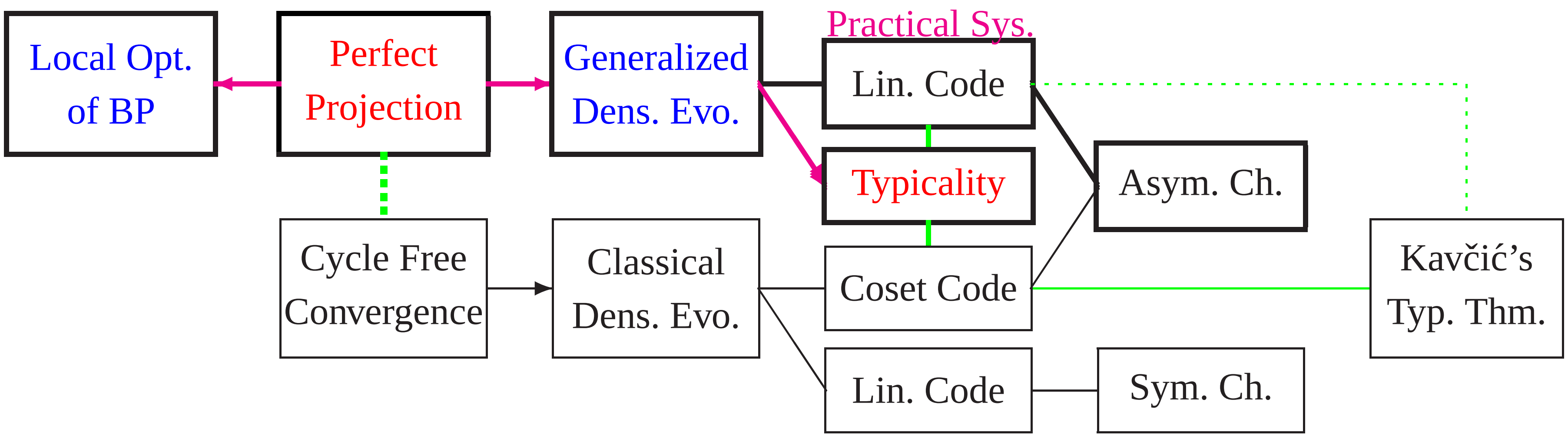
$$\mathbf{H} = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}$$

$$\mathbf{Hx} = \mathbf{0} \Longleftrightarrow x_6 = 0$$

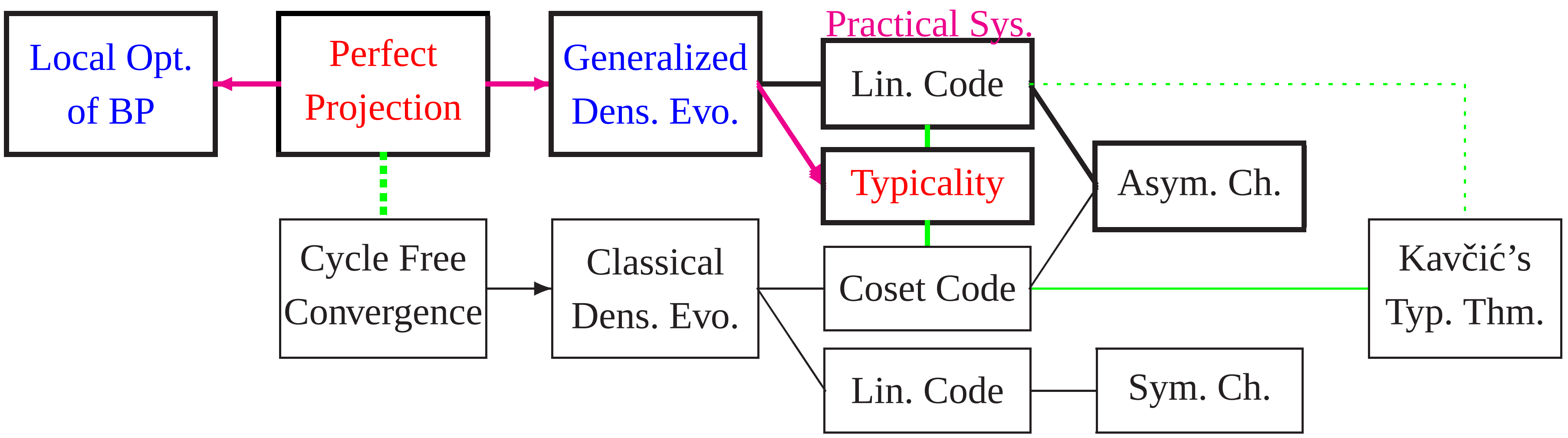
The cycle-free assumption is not enough. The condition that the support tree $\mathcal{N}^{(2l)}$ is perfectly projected guarantees the local optimality of BP.



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



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



BP Decoder	Optimality	Analysis Tools	Simulation
Cycle-free + Non-Sym. Chs.	✓	✓	optimal
LDPC Codes + Sym. Chs.	Local	Dens. Evo.	outstanding
LDPC Codes + Non-Sym. Chs.	Local	New Dens. Evo.	outstanding

Content

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

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

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 $\Longrightarrow$ Tight for BSCs

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 $\Longrightarrow$ 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

DE vs. Finite-Dim. Bounds

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 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 imferred.
- 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.

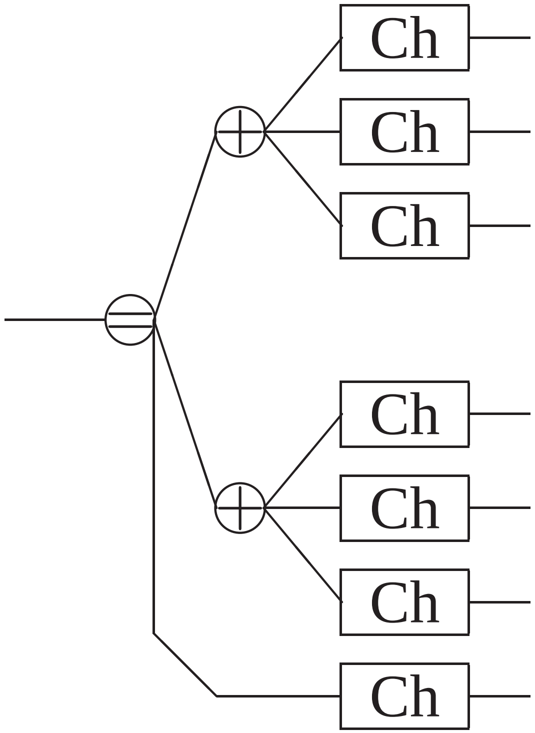
Finite-Dim. Bounds

- 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 imferred.
- Simpler iteration formula admits more analysis.

Reference: [Burshtein, Miller 01], [Khandekar 02], [Land *et al.* 03], and [Sutskover, Shamai 03].

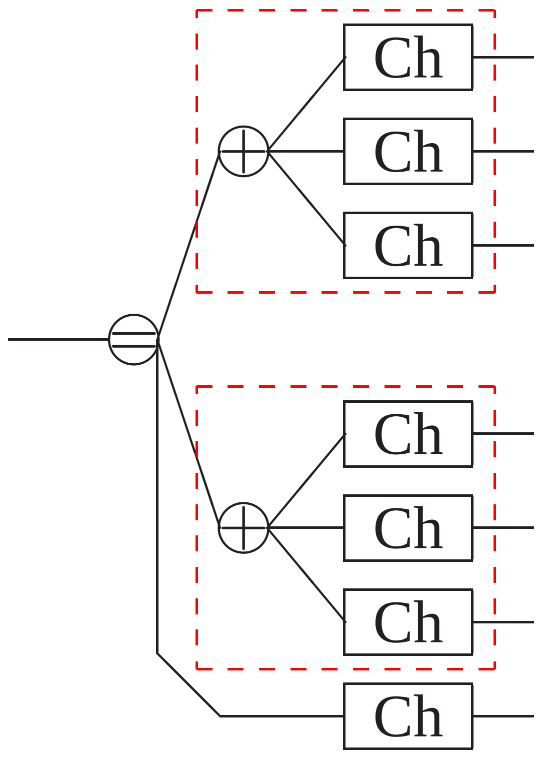
Iterative Bounding Technique

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



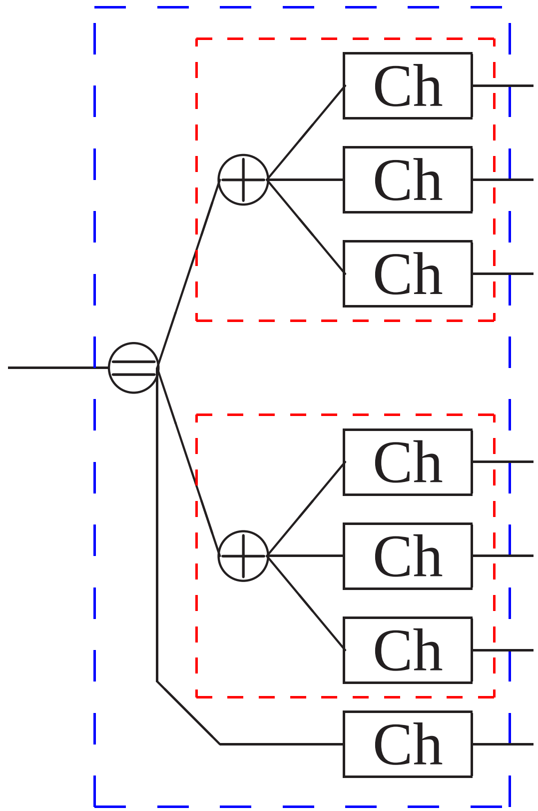
Iterative Bounding Technique

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



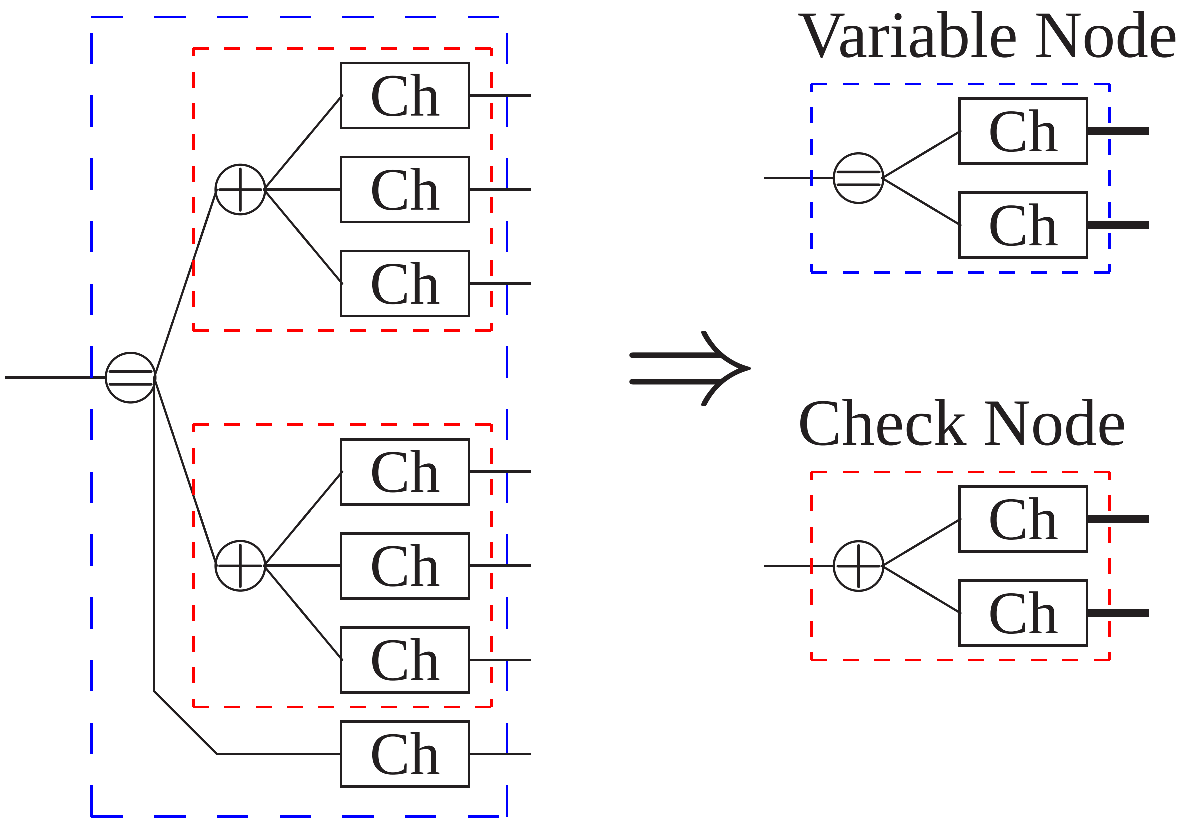
Iterative Bounding Technique

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



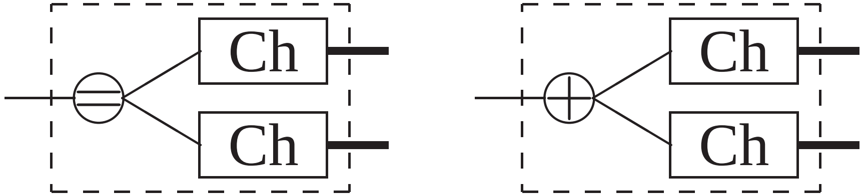
Iterative Bounding Technique

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



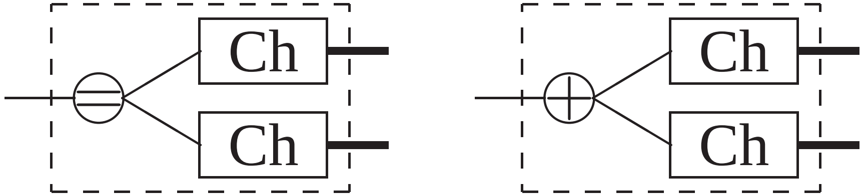
Existing Finite Dim. Bnds.

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



Existing Finite Dim. Bnds.

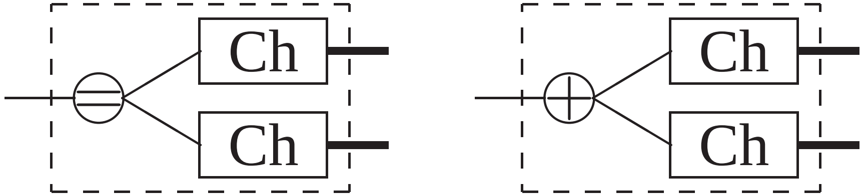
- Assumption: binary-input/symmetric-output channels.
- Iteratively trace the evolution of a one-dimensional index.
  - $p_e := \mathsf{P}\{X \neq \hat{X}_{ML}(Y)\}$.



  - [Burshtein and Miller 02]: $SB := 2\mathsf{E}_{XY}\{\mathsf{P}(\bar{X}|Y)\}$. Note: $SB \neq 2p_e$
  - [Khandekar and McEliece 01]: $CB := \mathsf{E}_{XY}\{\sqrt{\frac{\mathsf{P}(\bar{X}|Y)}{\mathsf{P}(X|Y)}}\}$.
  - [Land *et al.* 03], [Sutskover *et al.* 03]: Mutual information, or equivalently, conditional entropy $h := H_2(X|Y)$.

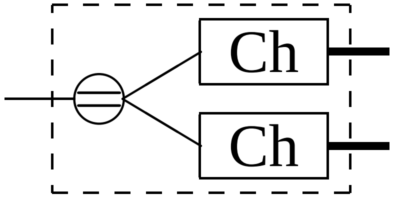
Existing Finite Dim. Bnds.

- Assumption: binary-input/symmetric-output channels.
- Iteratively trace the evolution of a one-dimensional index.
  - $p_e := \mathsf{P}\{X \neq \hat{X}_{ML}(Y)\}$.



  - [Burshtein and Miller 02]: $SB := 2\mathsf{E}_{XY}\{\mathsf{P}(\bar{X}|Y)\}$. Note: $SB \neq 2p_e$
  - [Khandekar and McEliece 01]: $CB := \mathsf{E}_{XY}\{\sqrt{\frac{\mathsf{P}(\bar{X}|Y)}{\mathsf{P}(X|Y)}}\}$.
  - [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.

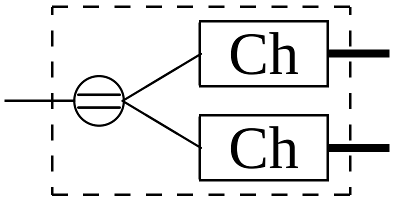
Discovery



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

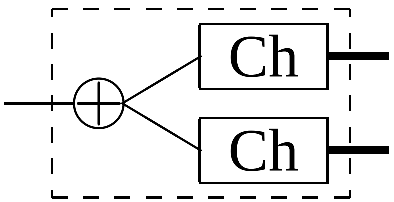
$$\text{INFO}_{out} \leq V_{BEC}(\text{INFO}_{in,1}, \text{INFO}_{in,1})$$

Discovery



$$\mathrm{INFO}_{out} \geq V_{BSC}(\mathrm{INFO}_{in,1}, \mathrm{INFO}_{in,1})$$

$$\mathrm{INFO}_{out} \leq V_{BEC}(\mathrm{INFO}_{in,1}, \mathrm{INFO}_{in,1})$$



$$\mathrm{INFO}_{out} \geq C_{BEC}(\mathrm{INFO}_{in,1}, \mathrm{INFO}_{in,1})$$

$$\mathrm{INFO}_{out} \leq C_{BSC}(\mathrm{INFO}_{in,1}, \mathrm{INFO}_{in,1})$$

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]

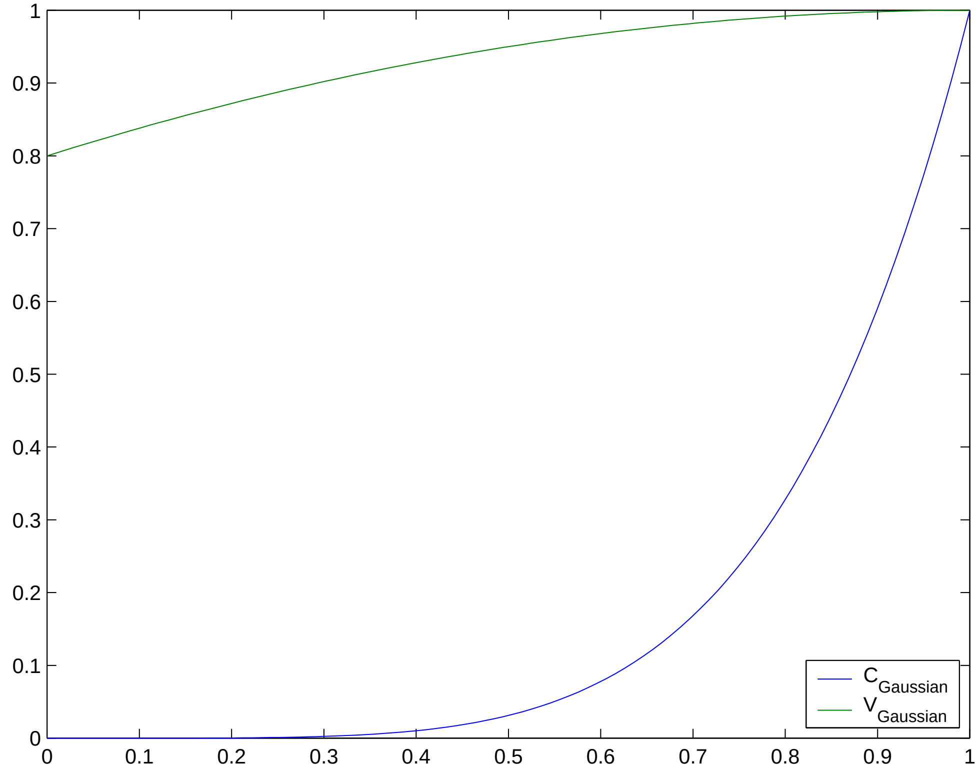
Application 2: EXIT Chart

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

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

Iterative Equation:

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



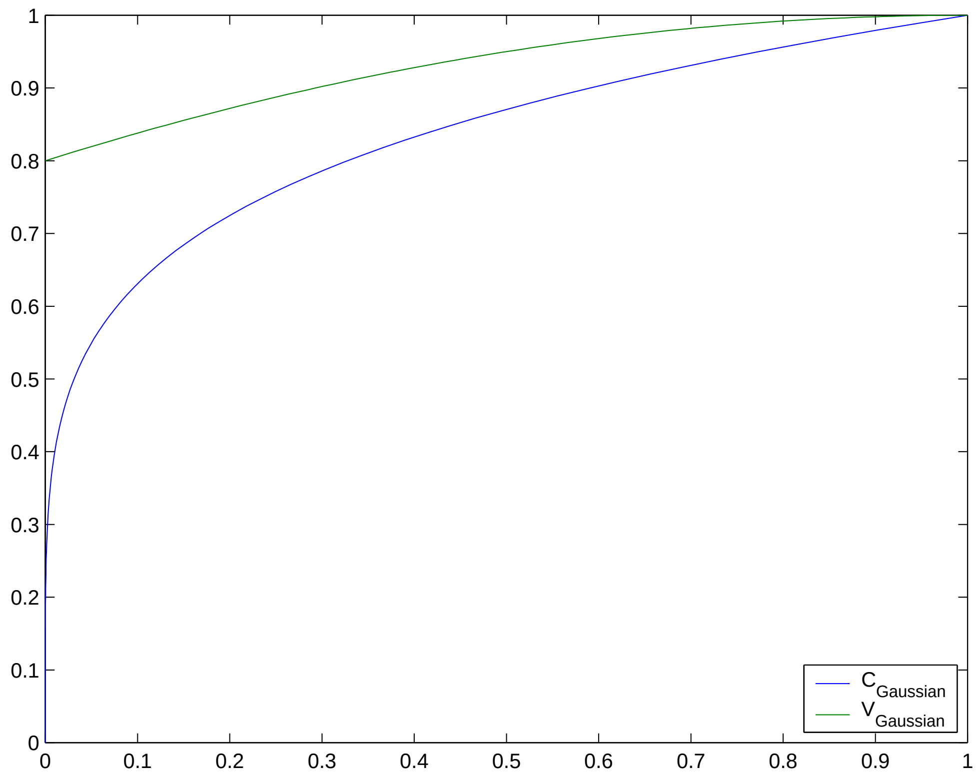
Application 2: EXIT Chart

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

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

Iterative Equation:

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



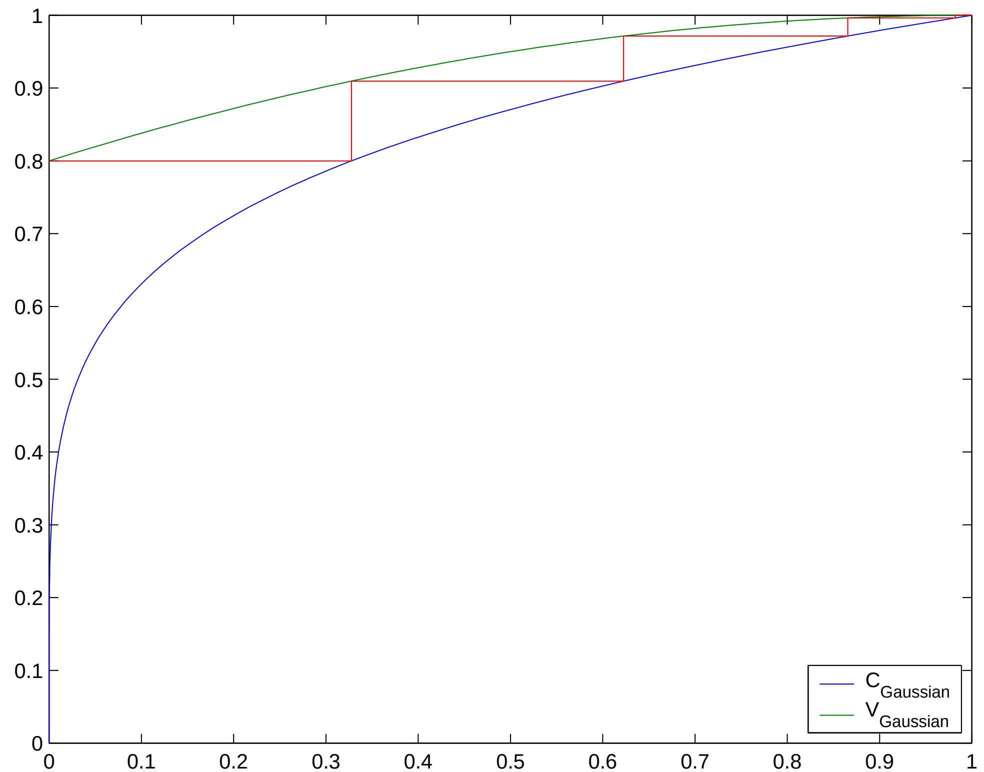
Application 2: EXIT Chart

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

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

Iterative Equation:

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



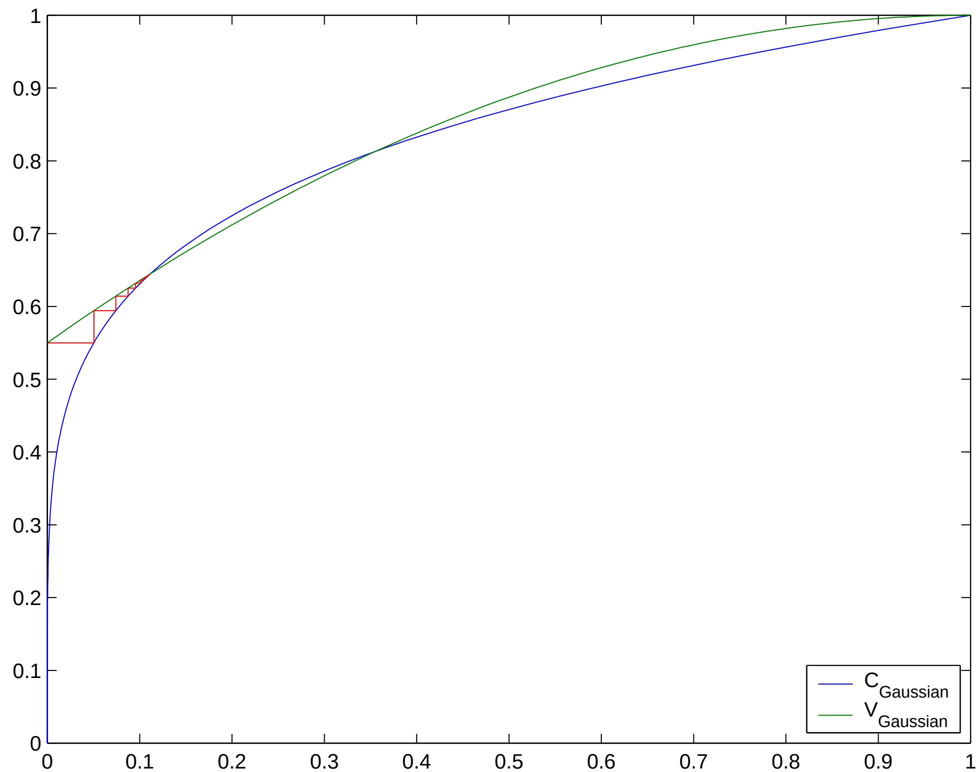
Application 2: EXIT Chart

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

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

Iterative Equation:

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



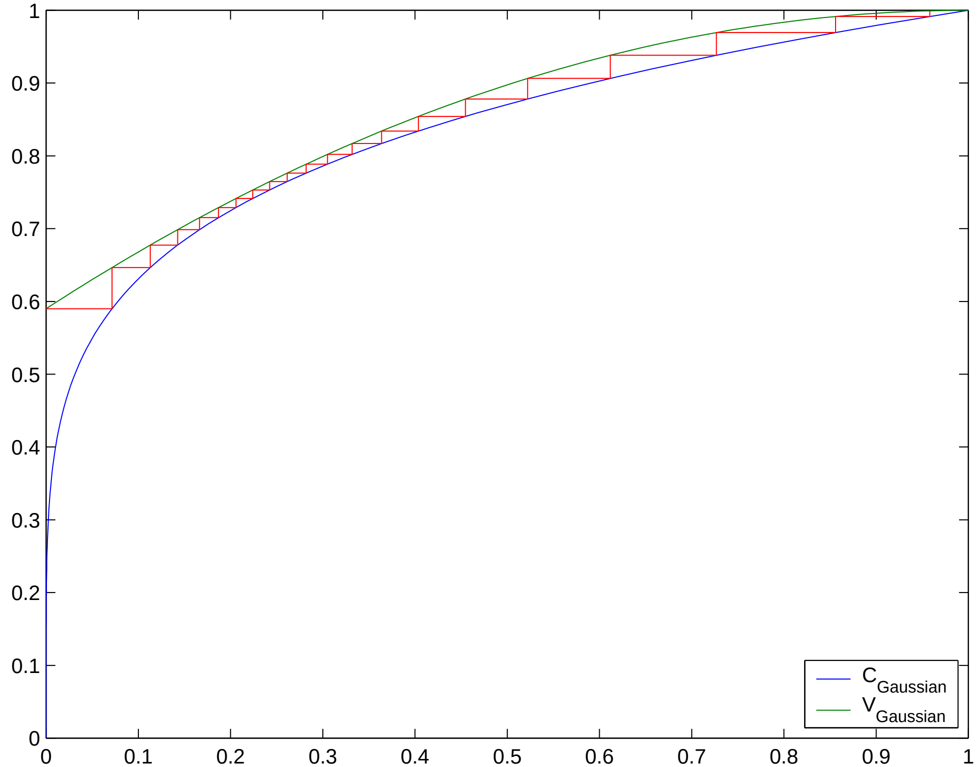
Application 2: EXIT Chart

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

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

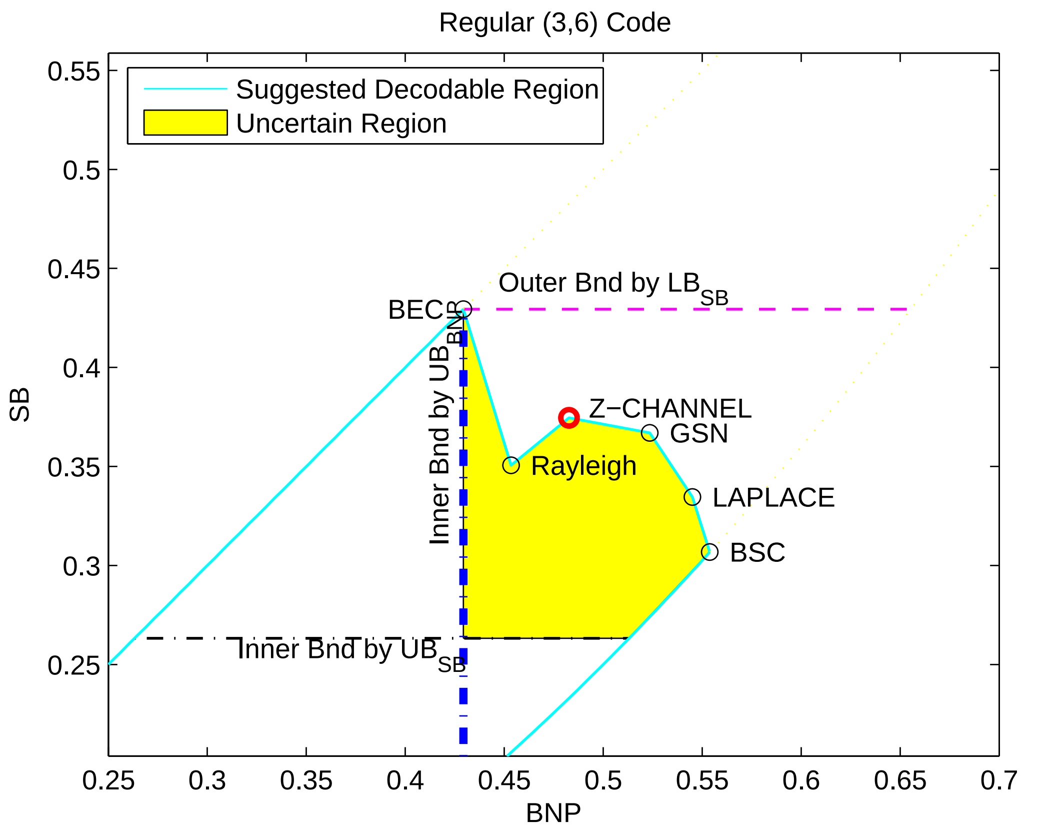
Iterative Equation:

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



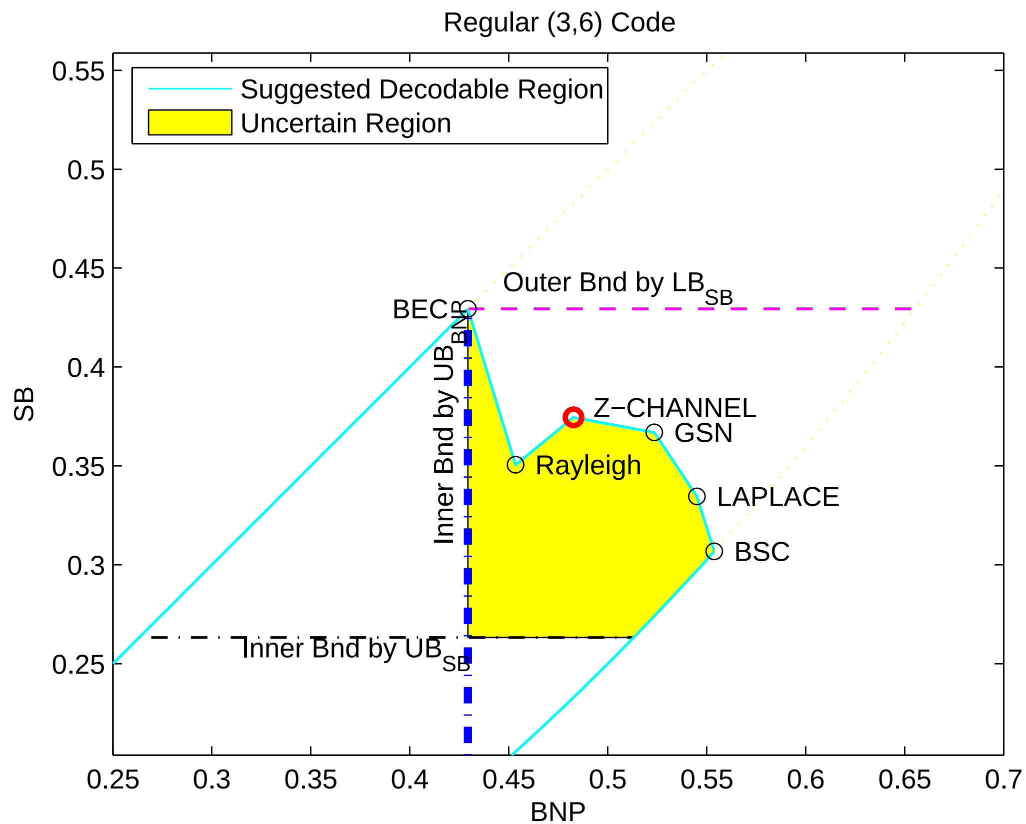
The First Bound for Non-Symmetric Channels

$$\mathrm{BNP}^{(l+1)} \leq \mathrm{BNP}^{(0)} \left(1 - \left(1 - \mathrm{BNP}^{(l)}\right)^{d_c - 1}\right)^{d_v - 1}$$



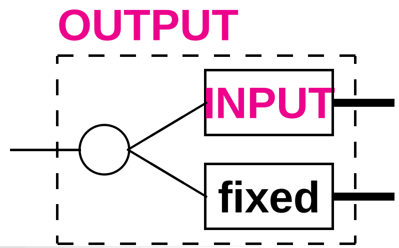
The First Bound for Non-Symmetric Channels

$$\mathrm{BNP}^{(l+1)} \leq \mathrm{BNP}^{(0)} \left(1 - \left(1 - \mathrm{BNP}^{(l)}\right)^{d_c - 1}\right)^{d_v - 1}$$



- Uniformly good performance for general fading channels.

$UB_{\text{BNP},SB}$ **given** $(\text{BNP}_{in}, SB_{in})$



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

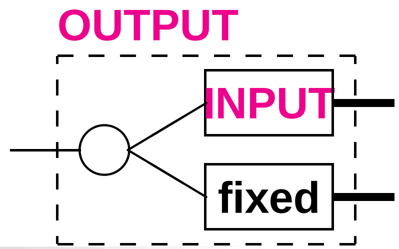
$$\forall p_2 \in [0, 1/2], \max \quad \int \text{BNP}_{out}(p_1, p_2) dP_1(p_1)$$

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

$$\text{subject to} \quad \int \text{BNP}(p_1) dP_1(p_1) \leq \text{BNP}_{in,1}$$

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

$UB_{\text{BNP},SB}$ given $(\text{BNP}_{in}, SB_{in})$

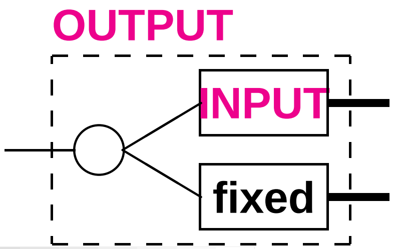


Equivalent to finding the universal optimizer for all $p_2 \in [0, 1/2]$:

$$\begin{aligned}
\forall p_2 \in [0, 1/2], \max \quad & \int \text{BNP}_{out}(p_1, p_2) dP_1(p_1) \\
& \text{and} \int SB_{out}(p_1, p_2) dP_1(p_1) \\
\text{subject to} \quad & \int \text{BNP}(p_1) dP_1(p_1) \leq \text{BNP}_{in,1} \\
& \int SB(p_1) dP_1(p_1) \leq SB_{in,1}
\end{aligned}$$

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

$UB_{\mathrm{BNP},SB}$ given $(\mathrm{BNP}_{in}, SB_{in})$

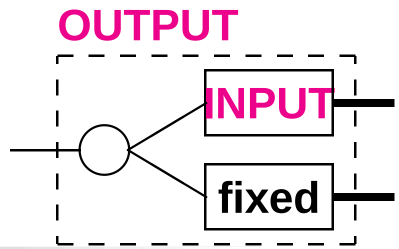


Equivalent to finding the universal optimizer for all $p_2 \in [0, 1/2]$:

$$
\begin{aligned}
\forall p_2 \in [0,1/2], \quad & \int \mathrm{BNP}_{out}(p_1, p_2)\, dP_1^*(p_1) \\
& \text{and} \int SB_{out}(p_1, p_2)\, dP_1^*(p_1) \\
\text{satisfying} \quad & \int \mathrm{BNP}(p_1)\, dP_1^*(p_1) \leq \mathrm{BNP}_{in,1} \\
& \int SB(p_1)\, dP_1^*(p_1) \leq SB_{in,1}
\end{aligned}
$$

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

$UB_{\mathrm{BNP},SB}$ **given** $(\mathrm{BNP}_{in}, SB_{in})$



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

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

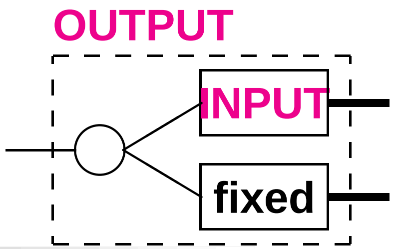
$$\text{and} \int SB_{out}(p_1, p_2) dP_1(p_1)$$

$$\text{subject to} \int \mathrm{BNP}(p_1) dP_1(p_1) \leq \mathrm{BNP}_{in,1}$$

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

- 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., $dP_1^\dagger(p_1)$.

$UB_{\mathrm{BNP},SB}$ given $(\mathrm{BNP}_{in}, SB_{in})$



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

$$\int \mathrm{BNP}_{out}(p_1, p_2) dP_1^{\dagger}(p_1) \geq \int \mathrm{BNP}_{out}(p_1, p_2) dP_1(p_1), \text{ and}$$

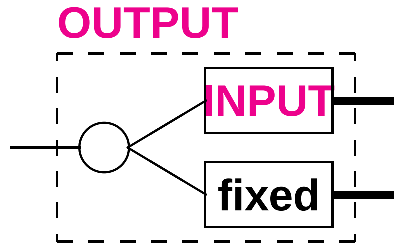
$$\int SB_{out}(p_1, p_2) dP_1^{\dagger}(p_1) \geq \int SB_{out}(p_1, p_2) dP_1(p_1), \forall p_2 \in [0, 1/2]$$

$$\text{subject to } \int \mathrm{BNP}(p_1) dP_1(p_1) \leq \mathrm{BNP}_{in,1}$$

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

- 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., $dP_1^{\dagger}(p_1)$.

$dP^*(p)$ and $dP^\dagger(p)$



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

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

$$t = \frac{SB_{in,1}}{\text{BNP}_{in,1}}$$

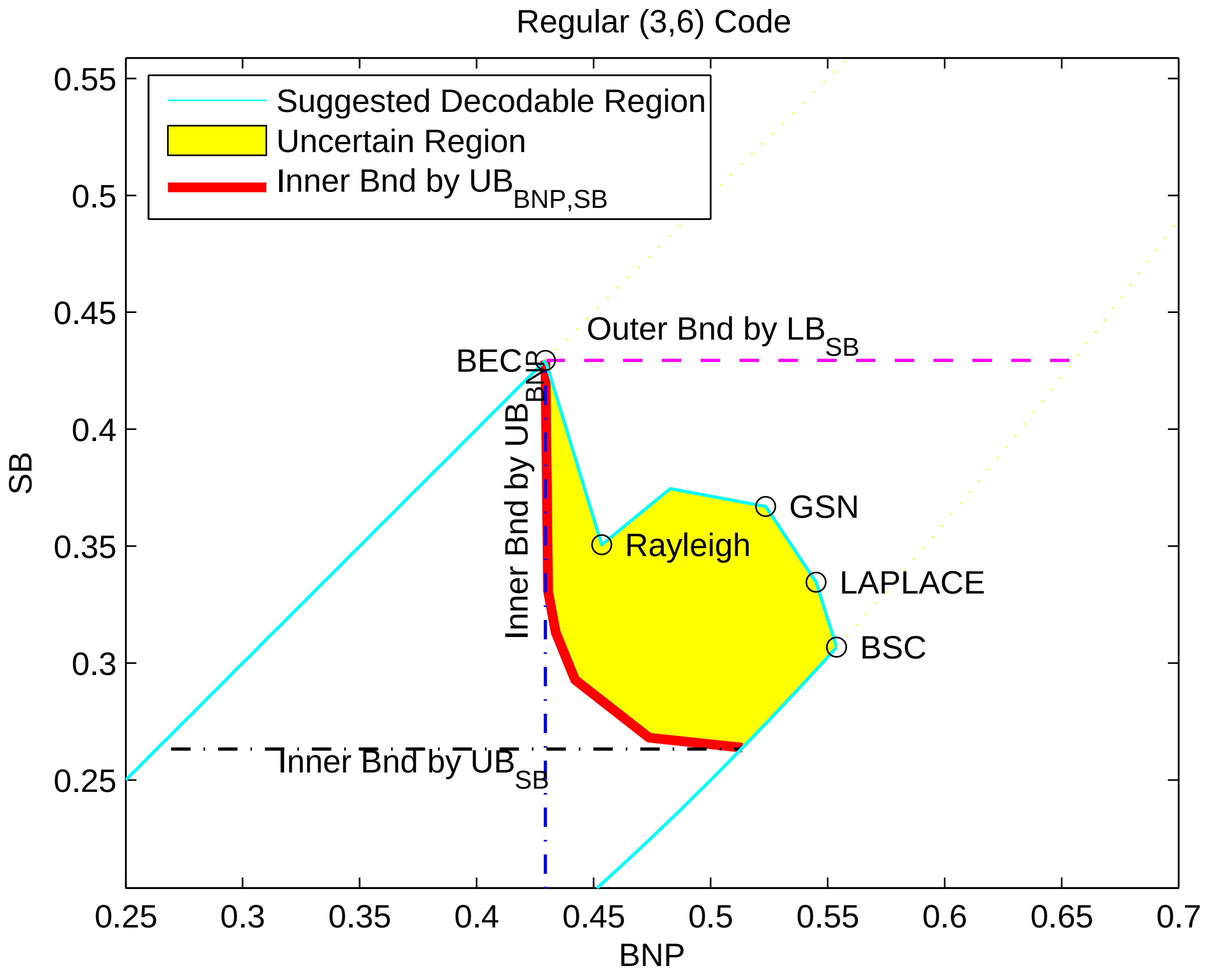
$$f_{SB} = \begin{cases} 0 & \text{if } 2\sqrt{t\text{BNP}_{in,1}} - t + \sqrt{\text{BNP}_{in,1}(2t - \text{BNP}_{in,1})} \geq 0 \\ \frac{\eta(w^*)}{2t(t-\text{BNP}_{in,1})^2} & \text{otherwise} \end{cases},$$

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

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

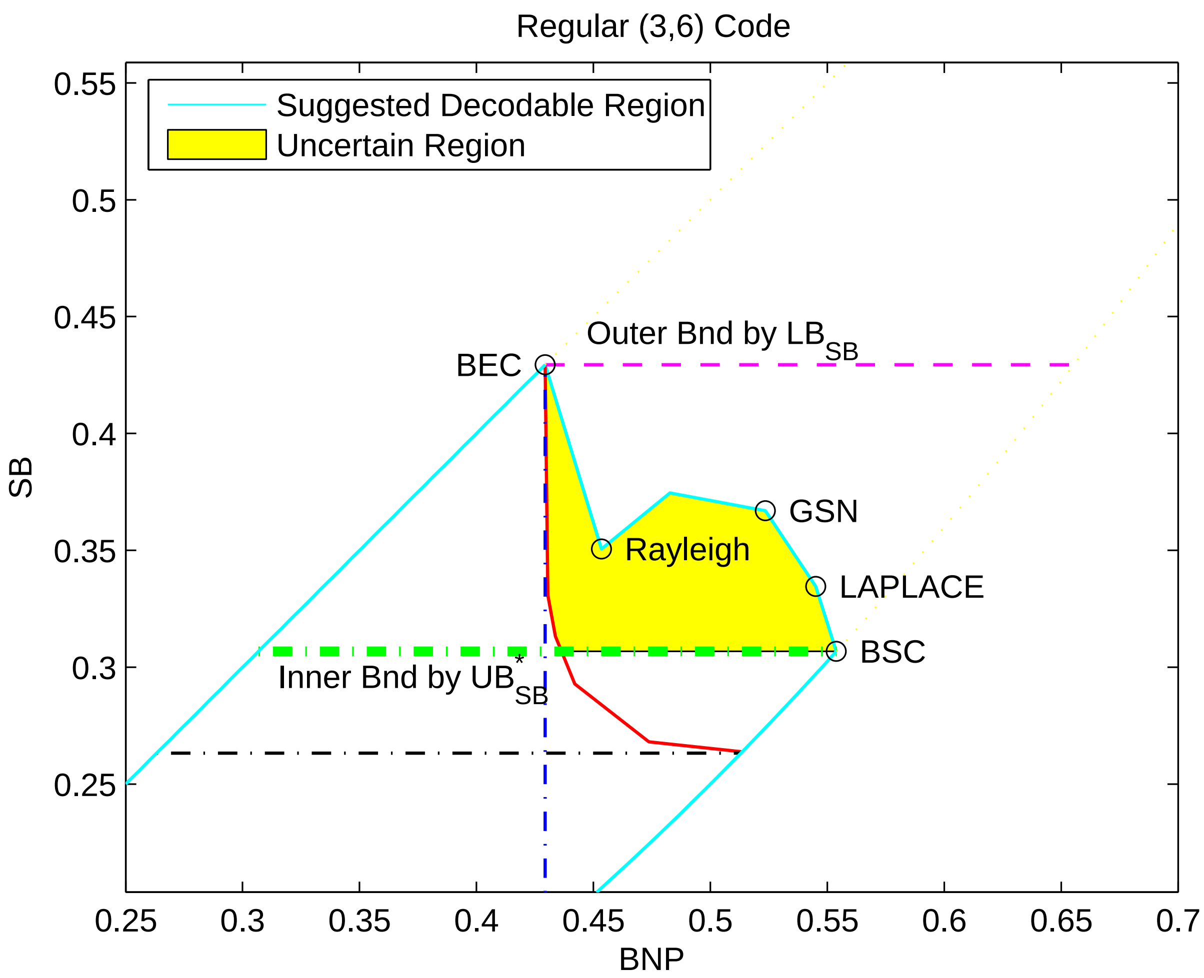
$UB_{\mathrm{BNP},SB}$

- Strict improvements over existing one-dimensional bounds.



A Non-iterative Tight SB Bound

- Iteration-based approach vs. Non-iteration-based approach



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

- The same $\mathbf{H}$ as in the $\mathsf{GF}(2)$ codes. But $\mathbf{Hx} = \mathbf{0}$ in $\mathbb{Z}_m$.

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

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

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

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

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

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

Theorem (A Pairwise-BNP-Based Bound)

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

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

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

Theorem (A Pairwise-BNP-Based Bound)

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

- Matched nec. & suff. stability conditions, depending on $\max_{x \in \mathbb{Z}_m \backslash \{0\}} \mathrm{BNP}^{(0)}(0 \to x)$

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

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

Theorem (A Pairwise-BNP-Based Bound)

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

- Matched nec. & suff. stability conditions, depending on $\max_{x \in \mathbb{Z}_m \setminus \{0\}} \mathrm{BNP}^{(0)}(0 \to x)$

C.-C. Wang, S.R. Kulkarni, and H.V. Poor, "On Finite-Dimensional Bounds for LDPC-like Codes with Iterative Decoding," in *Proc. Int'l Symp. Inform. Theory & its Applications.*, Parma, Italy, Oct. 2004.

$\mathbb{Z}_m$ LDPC Coded 8PAM

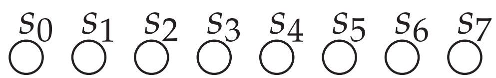
- [MacKay *et al.* 98]: With the same amount of info. bits, $\mathbb{Z}_m$ codes have better performance at the cost of complexity, $\mathcal{O}(m)$.

$\mathbb{Z}_m$ 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, $\mathcal{O}(m)$.

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

$$\mathbf{S} = \{s_0, s_1, \cdots, s_7\}$$

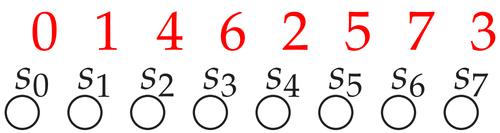


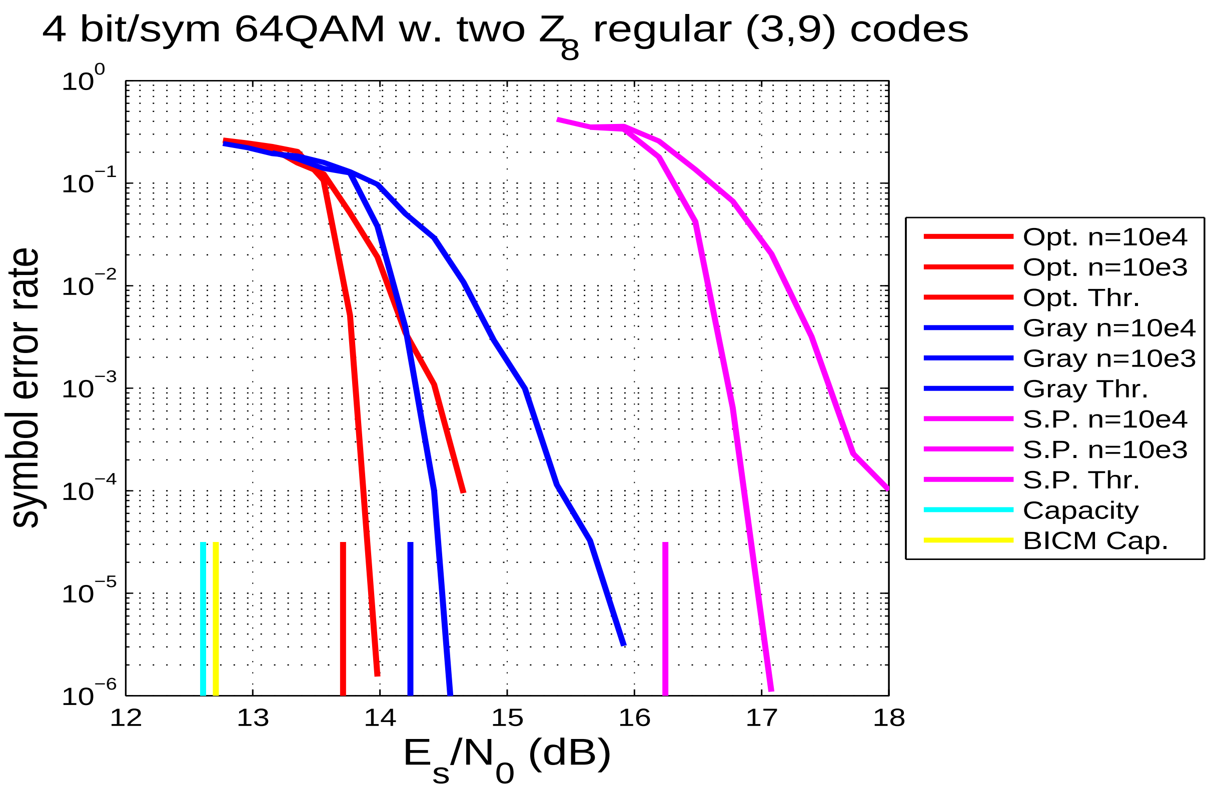
$\mathbb{Z}_m$ LDPC Coded 8PAM

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

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

$$\mathbf{S} = \{s_0, s_1, \cdots, s_7\}$$



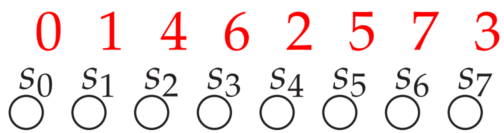


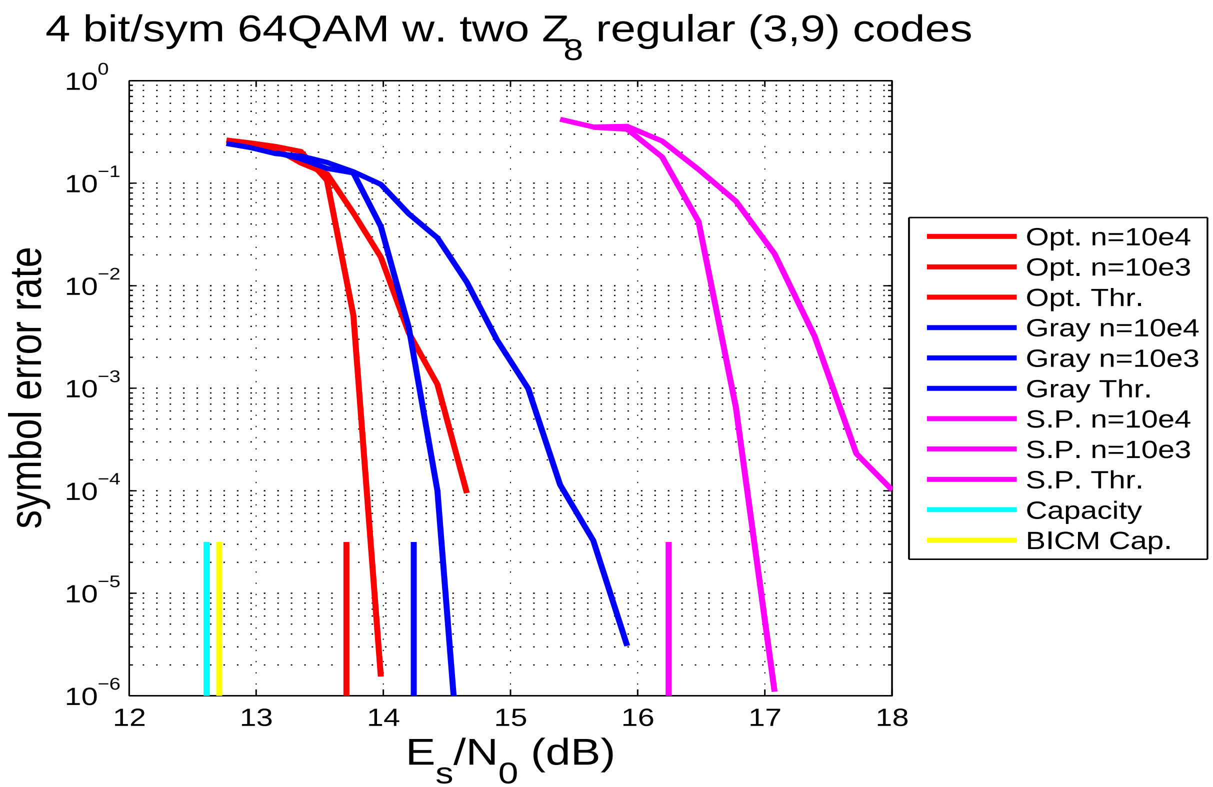
$\mathbb{Z}_m$ LDPC Coded 8PAM

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

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

$$\mathbf{S} = \{s_0, s_1, \cdots, s_7\}$$





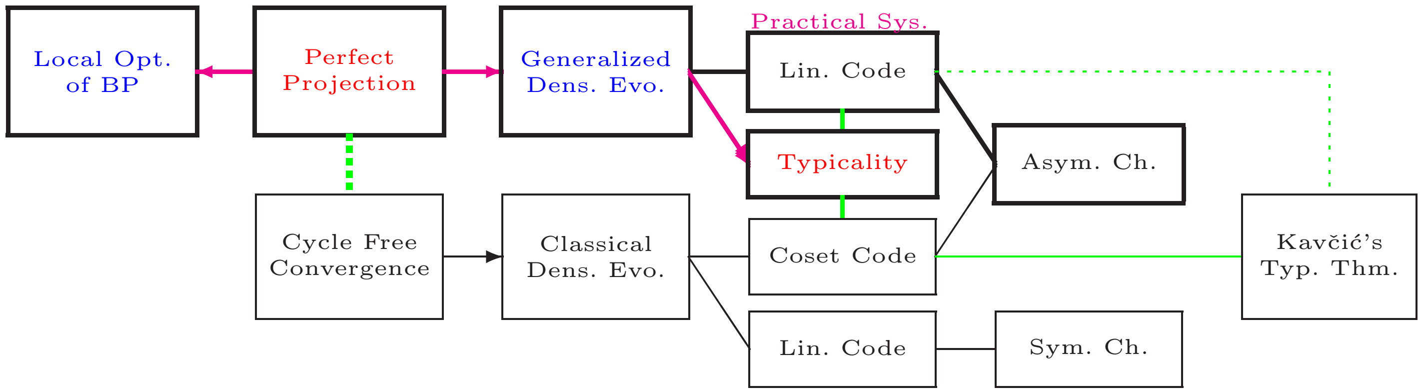
- 0.3dB better than BICM w. bin. regular codes
- 0.3dB worse than BICM w. bin. ir. codes
- Further code optimization is the next step.
- Compatible with almost all equalization schemes

Conclusion

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

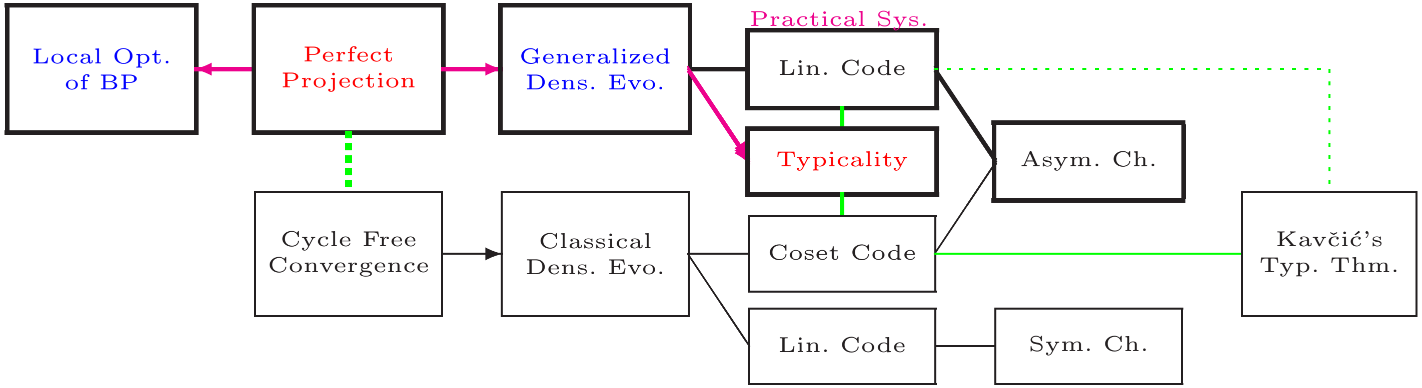
Conclusion

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



Conclusion

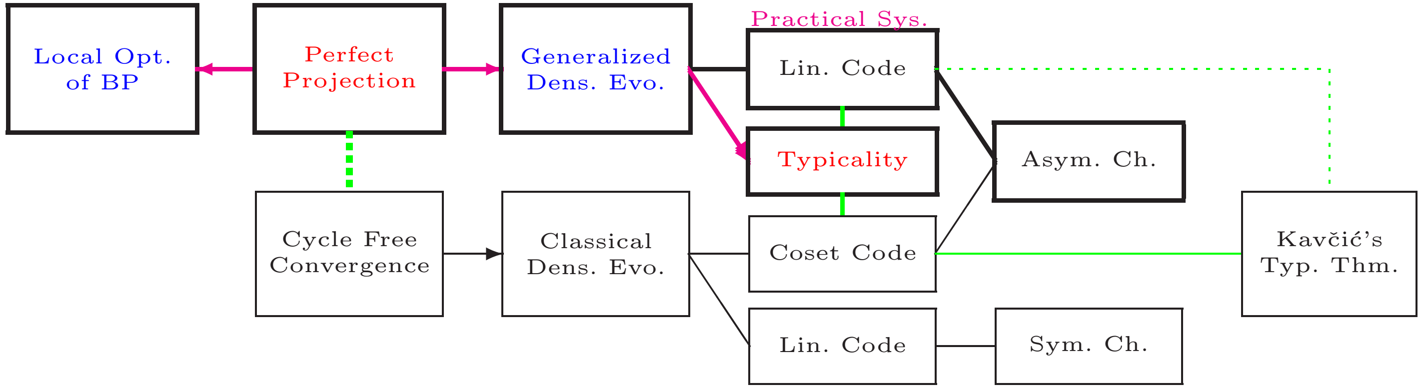
- Part I — Linear LDPC codes on symmetric & non-symmetric channels



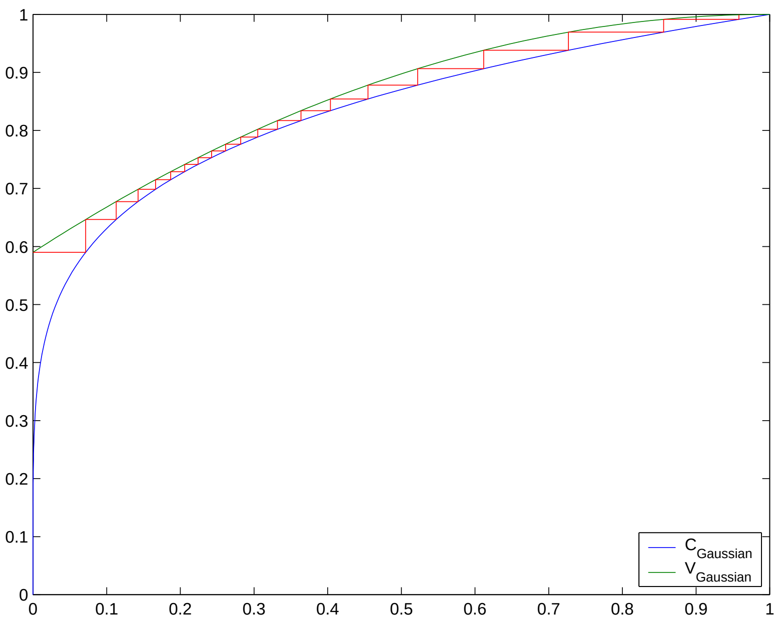
- Part II — EXIT chart & finite-dim. bounds for LDPC codes

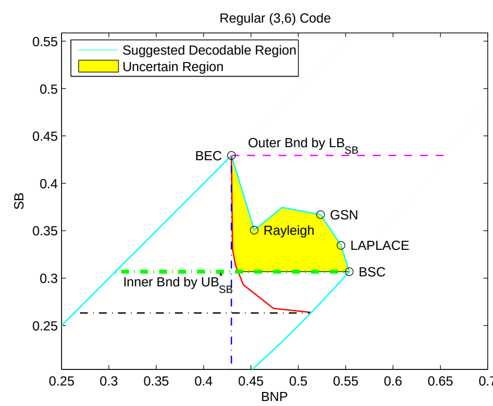
Conclusion

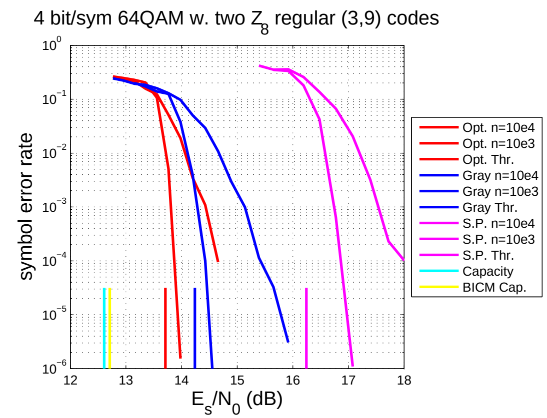
- Part I — Linear LDPC codes on symmetric & non-symmetric channels



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







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.