Spatially-Coupled Coded CPM: Asymptotic Analysis and Optimization

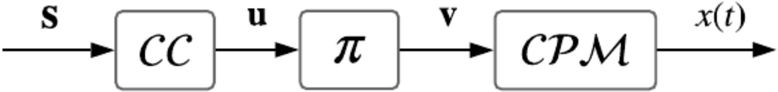
Tarik Benaddi

IMT Atlantiques, tarik.benaddi@imt-atlantique.fr

GdR ISIS Rennes, 11 June 2017

Plan

System Model

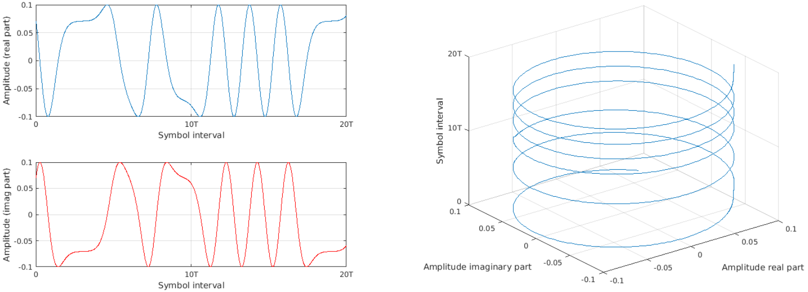


$$x(t) = \sqrt{\frac{2E_s}{T}} \cos\left(2\pi f_0 t + \theta(t, v) + \theta_0\right)$$

with

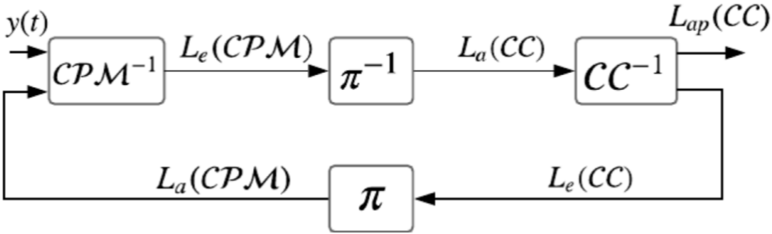
$$\theta(t, v) = \pi h \sum_{i=0}^{N-1} v_i q(t - iT), \quad q(t) = \begin{cases} \int_0^t g(\tau) d\tau \\ 1/2, t > L \end{cases}$$

- Signal $x(t)$ has a constant envelope and a continuous phase.

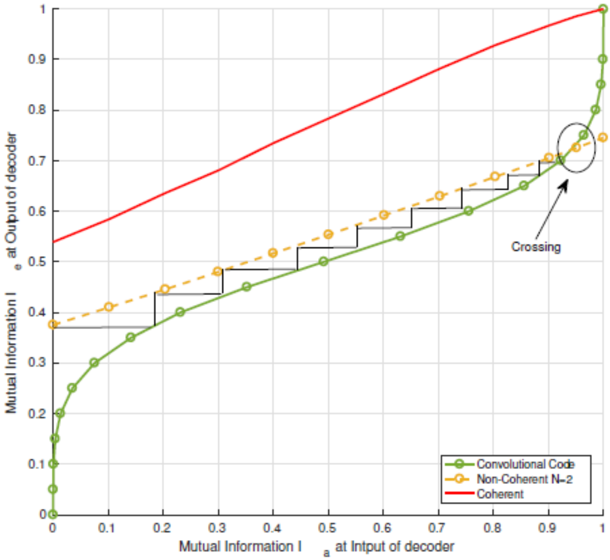


(a.) (b.)

Figure: Binary 2GMSK $h = 1/2$, $BT = 0.25$ (a.) Amplitude (b.) Envelop.

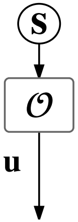


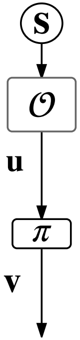
- At the receiver, we assume SISO CPM detection based on any Rimoldi's or Laurent's representation based receiver.

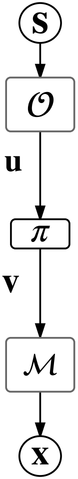

Mutual Information I_e at Output of decoder
Mutual Information I_a at Intput of decoder
Crossing
Convolutional Code
Non-Coherent N=2
Coherent
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

- Serially concatenated systems based on convolutional codes have inherently limited performance in terms of convergence (BP) threshold due to the outer code (except if time-varying convolutinal codes are considered).
- Capacity approaching codes can be designed to improve convergence thresholds by considering sparse graph codes enabling better BP thresholds.
- **Question:** can we improve the threshold of serially concatenated schemes with simple outer convolutional codes? $\Rightarrow$ spatial coupling could help to improve convergence threshold towards the MAP threshold of the underlying coding scheme.

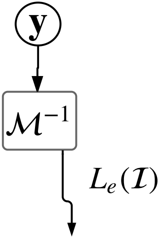

S

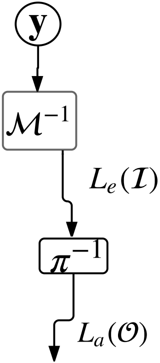

S
$\mathcal{O}$
u

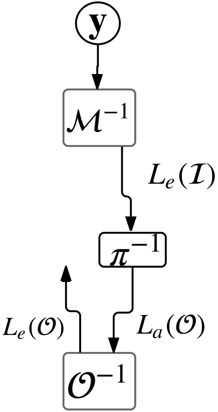

S
$\mathcal{O}$
u
π
v

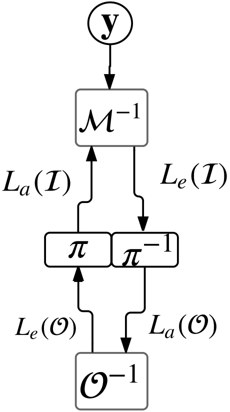

S
$\mathcal{O}$
u
π
v
$\mathcal{M}$
x

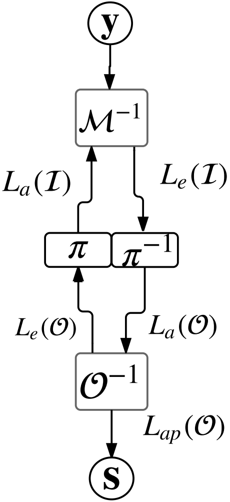

y


y
$\mathcal{M}^{-1}$
$L_e(\mathcal{I})$


y
$\mathcal{M}^{-1}$
$L_e(\mathcal{I})$
π^{-1}
$L_a(\mathcal{O})$


$\mathbf{y}$
$\mathcal{M}^{-1}$
$L_e(\mathcal{I})$
π^{-1}
$L_e(\mathcal{O})$
$L_a(\mathcal{O})$
$\mathcal{O}^{-1}$


y
$\mathcal{M}^{-1}$
$L_a(\mathcal{I})$
$L_e(\mathcal{I})$
π
π^{-1}
$L_e(\mathcal{O})$
$L_a(\mathcal{O})$
$\mathcal{O}^{-1}$


y
$\mathcal{M}^{-1}$
$L_a(\mathcal{I})$
$L_e(\mathcal{I})$
π
π^{-1}
$L_e(\mathcal{O})$
$L_a(\mathcal{O})$
$\mathcal{O}^{-1}$
$L_{ap}(\mathcal{O})$
s

1999 Felstrom *et al*: Convolutional LDPC codes

1999 Felstrom *et al*: Convolutional LDPC codes

2011 Kudekar *et al*: Saturation proof for the BEC

1999 Felstrom *et al*: Convolutional LDPC codes

2011 Kudekar *et al*: Saturation proof for the BEC

2014 Mitchell *et al*: Spatially coupled protographs

1999 Felstrom *et al*: Convolutional LDPC codes

2011 Kudekar *et al*: Saturation proof for the BEC

2014 Mitchell *et al*: Spatially coupled protographs

2014 Nguyen *et al*: Saturation proof for ISI channels

1999 Felstrom *et al*: Convolutional LDPC codes

2011 Kudekar *et al*: Saturation proof for the BEC

2014 Mitchell *et al*: Spatially coupled protographs

2014 Nguyen *et al*: Saturation proof for ISI channels

2015 Mouloudi *et al*: Spatially coupled turbo-codes

1999 Felstrom *et al*: Convolutional LDPC codes

2011 Kudekar *et al*: Saturation proof for the BEC

2014 Mitchell *et al*: Spatially coupled protographs

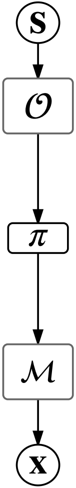
2014 Nguyen *et al*: Saturation proof for ISI channels

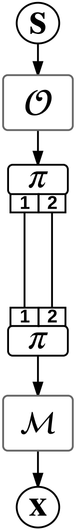
2015 Mouloudi *et al*: Spatially coupled turbo-codes

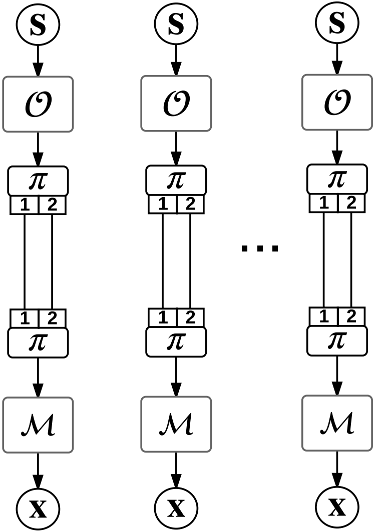
2016 Costello *et al*: Braided turbo-codes spatially-coupled turbo-codes are obtained by performing an *edge-spreading-like rule*:

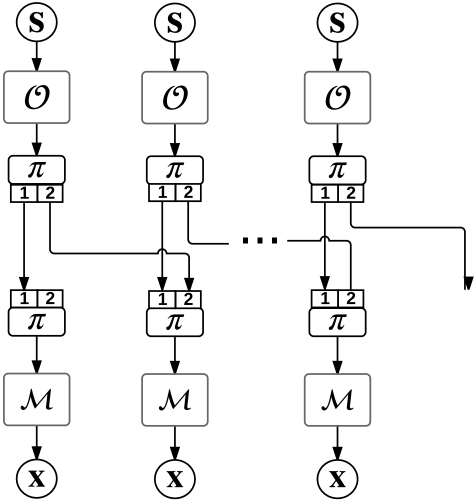
1. the encoded bits u are $m_s + 1$ bundles,
2. the obtained graph is then replicated L times,
3. we interconnect the L graphs by substituting the bundles of the same type. This substitution is given by the coupling matrix

$$B = [b_0, b_1, \dots b_{m_s}]$$

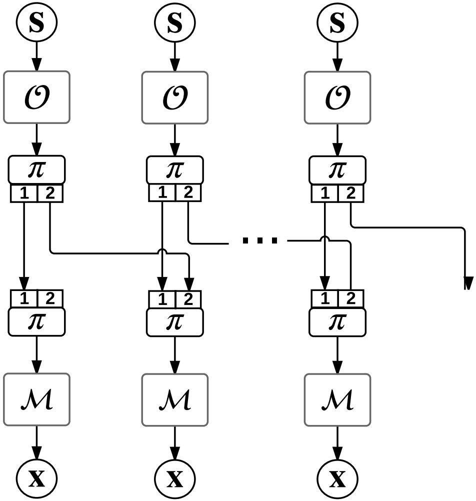

S
$\mathcal{O}$
π
$\mathcal{M}$
X

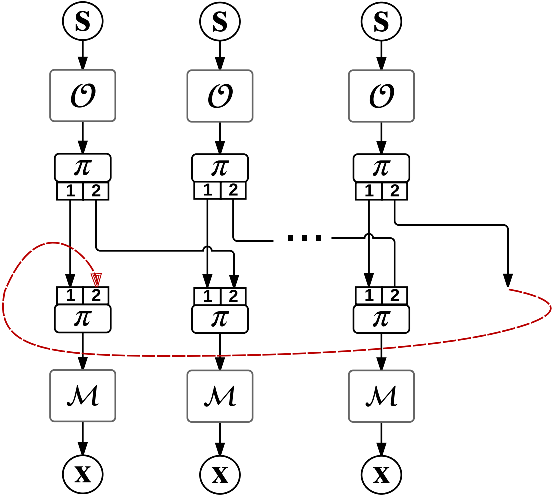

S
$\mathcal{O}$
π
1
2
1
2
π
$\mathcal{M}$
x

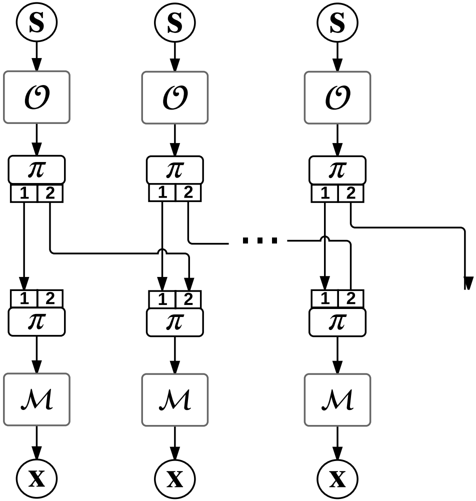

S
S
S
$\mathcal{O}$
$\mathcal{O}$
$\mathcal{O}$
π
1 2
...
$\mathcal{M}$
X

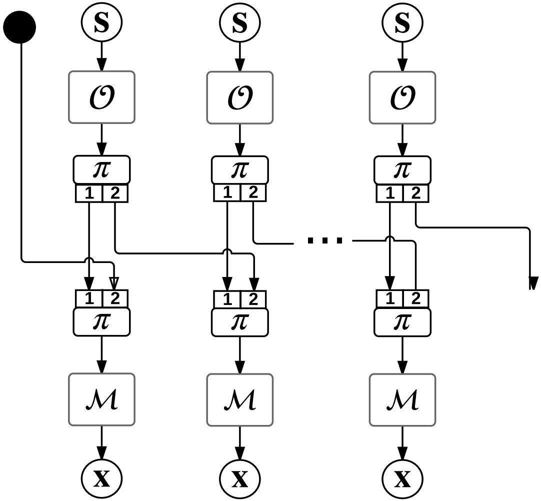

S
S
S
$\mathcal{O}$
$\mathcal{O}$
$\mathcal{O}$
π
1 2
...
$\mathcal{M}$
X
X
X

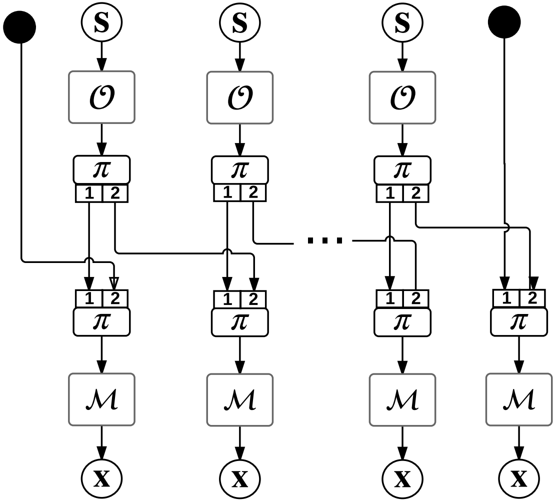
- m_s: coupling memory
- L: coupling length
- $B = [b_0, b_1 \ldots b_{m_s}]$: coupling matrix. $b_i \in \{0 \ldots N\}$
- $\sum_{i=1}^{m_s} b_i = N$

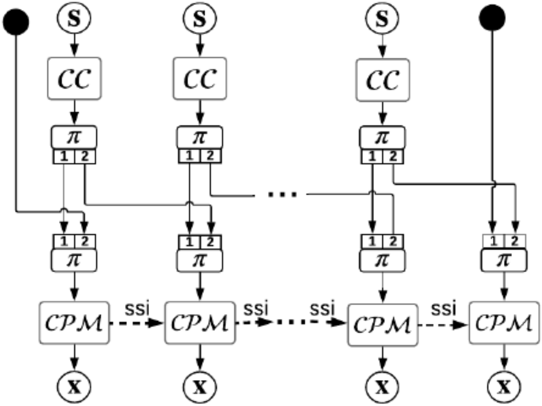

S
S
S
$\mathcal{O}$
$\mathcal{O}$
$\mathcal{O}$
π
π
π
1 2
1 2
1 2
1 2
1 2
1 2
π
π
π
$\mathcal{M}$
$\mathcal{M}$
$\mathcal{M}$
X
X
X


S
S
S
$\mathcal{O}$
$\mathcal{O}$
$\mathcal{O}$
π
1 2
π
1 2
π
1 2
1 2
π
1 2
π
1 2
π
$\mathcal{M}$
$\mathcal{M}$
$\mathcal{M}$
X
X
X


S
S
S
$\mathcal{O}$
$\mathcal{O}$
$\mathcal{O}$
π
1 2
...
$\mathcal{M}$
$\mathcal{M}$
$\mathcal{M}$
X
X
X


S
S
S
$\mathcal{O}$
$\mathcal{O}$
$\mathcal{O}$
π
1 2
...
$\mathcal{M}$
$\mathcal{M}$
$\mathcal{M}$
X
X
X


S
S
S
$\mathcal{O}$
$\mathcal{O}$
$\mathcal{O}$
π
1 2
...
$\mathcal{M}$
X


S
S
S
CC
CC
CC
π
1 2
...
CPM
ssi
X

- **Tail-biting**: same rate as concatenated schemes

$$R_L = R$$

- **Tail-biting**: same rate as concatenated schemes

$$R_L = R$$

- **Termination**: rate loss:

$$R_L = \frac{LR}{L + m_s} = R - \frac{m_s}{L + m_s}R$$

- Density evolution is complex for general channels/components

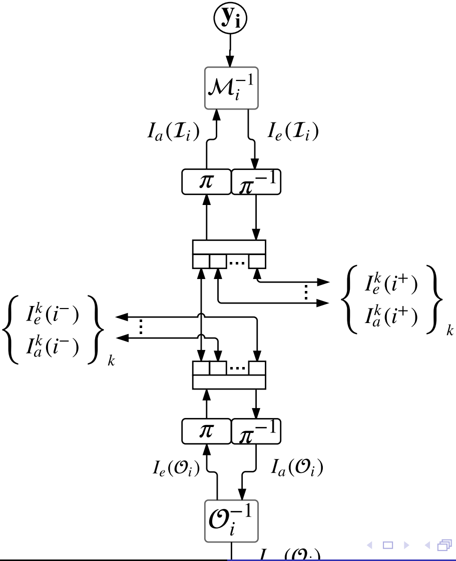
- Density evolution is complex for general channels/components
- EXIT chart analysis is usually used

- Density evolution is complex for general channels/components
- EXIT chart analysis is usually used
    1. we fix the channel parameter E_b/N_0

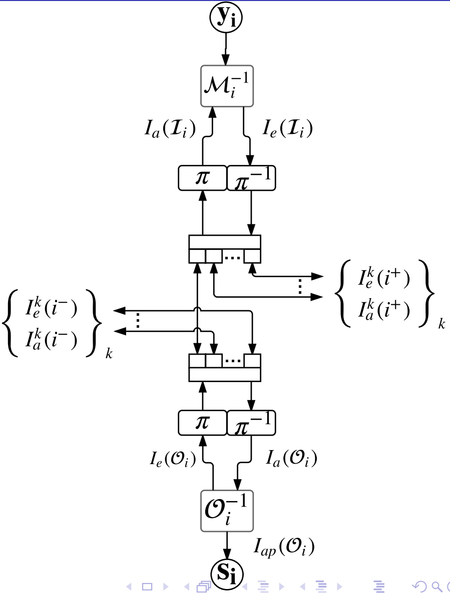
- Density evolution is complex for general channels/components
- EXIT chart analysis is usually used
  1. we fix the channel parameter E_b/N_0
  2. we track the mutual information between the LLRs and the corresponding bits

- Density evolution is complex for general channels/components
- EXIT chart analysis is usually used
    1. we fix the channel parameter E_b/N_0
    2. we track the mutual information between the LLRs and the corresponding bits
    3. the demodulator and the decoder are modeled with their transfer functions $T_{\mathcal{M}}(.)$ and $T_{\mathcal{O}}(.)$

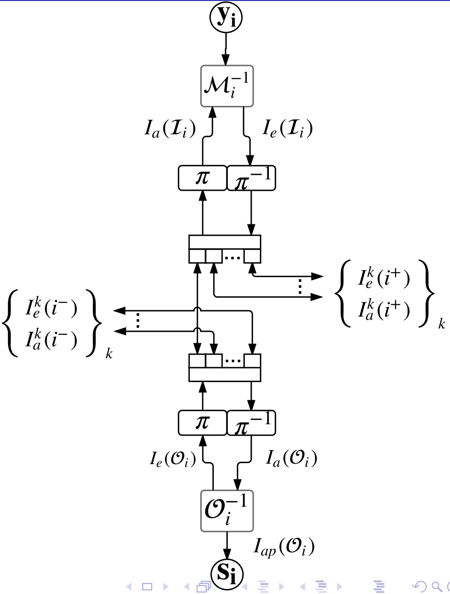
Notations


$\mathbf{y_i}$
$\mathcal{M}_i^{-1}$
$I_a(\mathcal{I}_i)$
$I_e(\mathcal{I}_i)$
π
π^{-1}
$I_e^k(i^+)$
$I_a^k(i^+)$
k
$I_e^k(i^-)$
$I_a^k(i^-)$
k
π
π^{-1}
$I_e(\mathcal{O}_i)$
$I_a(\mathcal{O}_i)$
$\mathcal{O}_i^{-1}$

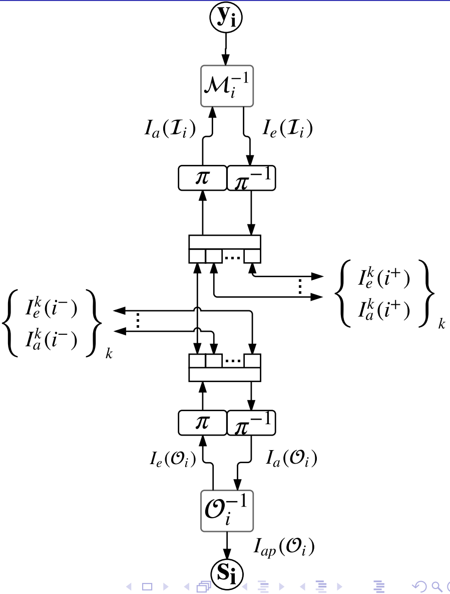
- $I_e(\mathcal{I}_i) = T_{\mathcal{M}}(I_a(\mathcal{I}_i))$



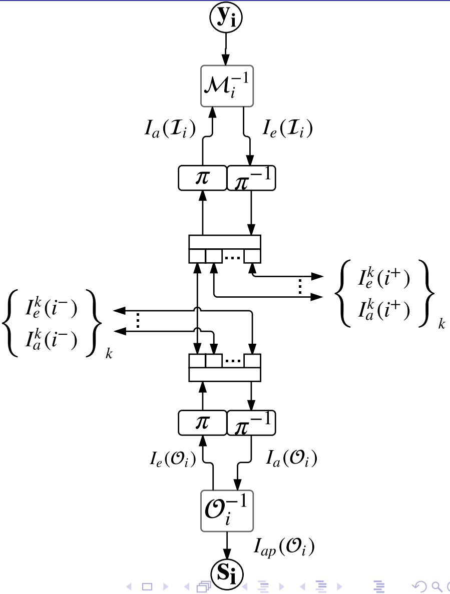
- $I_e(\mathcal{I}_i) = T_{\mathcal{M}}(I_a(\mathcal{I}_i))$
- $I_e^k(i^+) = I_e(\mathcal{I}_i)$



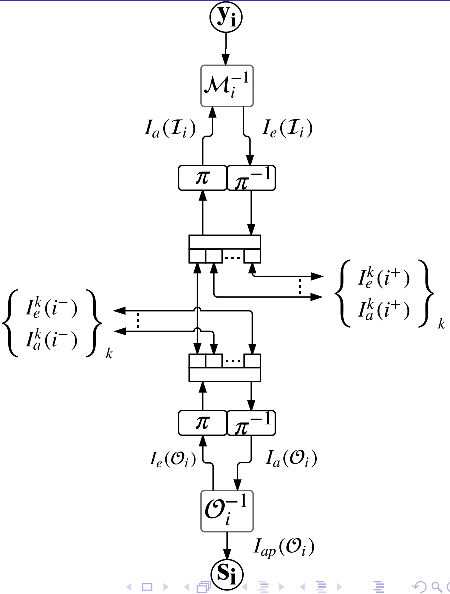
- $I_e(\mathcal{I}_i) = T_{\mathcal{M}}(I_a(\mathcal{I}_i))$
- $I_e^k(i^+) = I_e(\mathcal{I}_i)$
- $I_a(\mathcal{I}_i) = \frac{1}{N}\sum I_a^k(i^+).b_k$



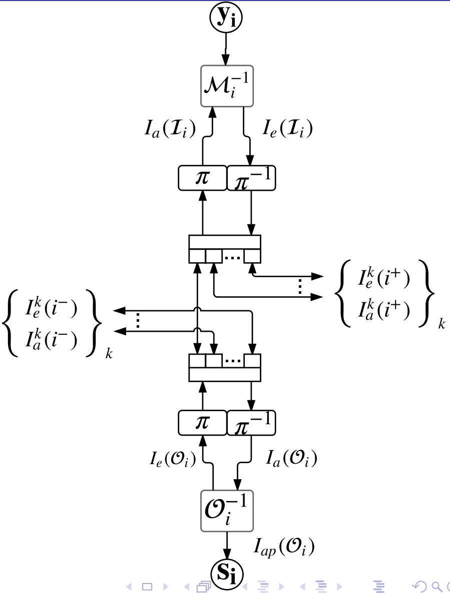
- $I_e(\mathcal{I}_i) = T_{\mathcal{M}}(I_a(\mathcal{I}_i))$
- $I_e^k(i^+) = I_e(\mathcal{I}_i)$
- $I_a(\mathcal{I}_i) = \frac{1}{N}\sum I_a^k(i^+).b_k$
- $I_e^k(i^-) = I_e(\mathcal{O}_i)$



- $I_e(\mathcal{I}_i) = T_{\mathcal{M}}(I_a(\mathcal{I}_i))$
- $I_e^k(i^+) = I_e(\mathcal{I}_i)$
- $I_a(\mathcal{I}_i) = \frac{1}{N}\sum I_a^k(i^+).b_k$
- $I_e^k(i^-) = I_e(\mathcal{O}_i)$
- $I_a(\mathcal{O}_i) = \frac{1}{N}\sum I_a^k(i^-).b_k$



- $I_e(\mathcal{I}_i) = T_{\mathcal{M}}(I_a(\mathcal{I}_i))$
- $I_e^k(i^+) = I_e(\mathcal{I}_i)$
- $I_a(\mathcal{I}_i) = \frac{1}{N}\sum I_a^k(i^+).b_k$
- $I_e^k(i^-) = I_e(\mathcal{O}_i)$
- $I_a(\mathcal{O}_i) = \frac{1}{N}\sum I_a^k(i^-).b_k$
- $I_e(\mathcal{O}_i) = T_{\mathcal{O}}(I_a(\mathcal{O}_i))$



- coupling gain: known bits at the boundaries

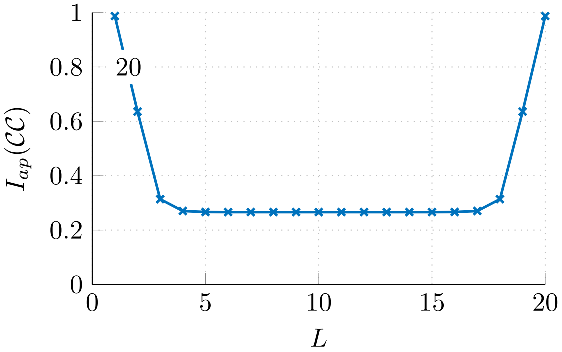


Figure: Convergence of the binary SC-CPM stages as function of the decoding iterations at $E_s/N_0 = -2.58$dB. Here $L = 20$.

- coupling gain: known bits at the boundaries

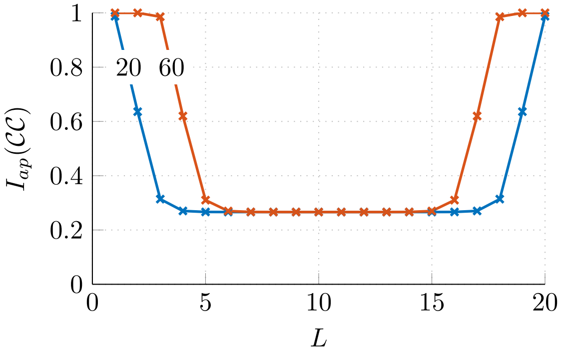


Figure: Convergence of the binary SC-CPM stages as function of the decoding iterations at $E_s/N_0 = -2.58$dB. Here $L = 20$.

- coupling gain: known bits at the boundaries

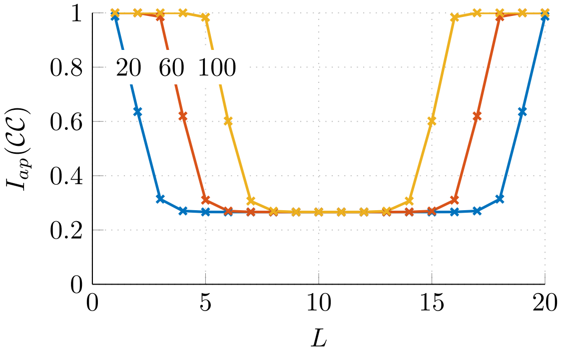


Figure: Convergence of the binary SC-CPM stages as function of the decoding iterations at $E_s/N_0 = -2.58$dB. Here $L = 20$.

- coupling gain: known bits at the boundaries

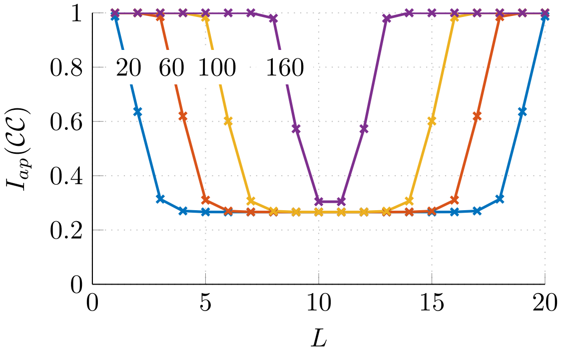


Figure: Convergence of the binary SC-CPM stages as function of the decoding iterations at $E_s/N_0 = -2.58$dB. Here $L = 20$.

- coupling gain: known bits at the boundaries

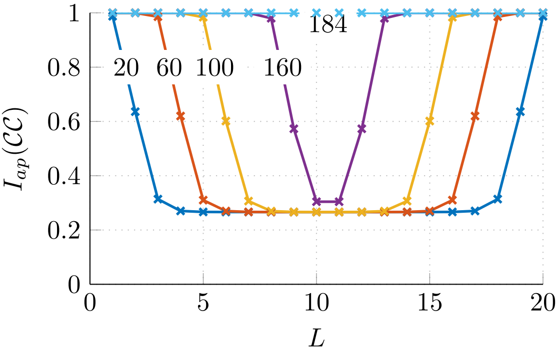


Figure: Convergence of the binary SC-CPM stages as function of the decoding iterations at $E_s/N_0 = -2.58$dB. Here $L = 20$.

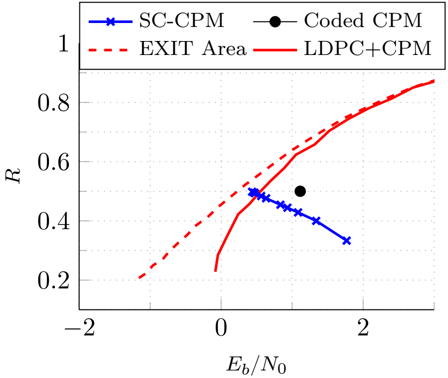


Figure: Threshold of coupled and uncoupled coded CPM. Comparison to the Area under the EXIT is also depicted. $B = [\frac{1}{2}, \frac{1}{2}]$

- $B = [b_0, b_1 \ldots b_{m_s}]$ où $b_i \in \{0 \ldots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$

- $B = [b_0, b_1 \dots b_{m_s}]$ où $b_i \in \{0 \dots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$
- $B = \frac{1}{N}.B \Rightarrow b_k \in [0, 1]$

- $B = [b_0, b_1 \dots b_{m_s}]$ où $b_i \in \{0 \dots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$
- $B = \frac{1}{N}.B \Rightarrow b_k \in [0, 1]$
- Minimize the number of iterations:

- $B = [b_0, b_1 \dots b_{m_s}]$ où $b_i \in \{0 \dots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$
- $B = \frac{1}{N}.B \Rightarrow b_k \in [0, 1]$
- Minimize the number of iterations:
  1. Choose B_0

- $B = [b_0, b_1 \dots b_{m_s}]$ où $b_i \in \{0 \dots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$
- $B = \frac{1}{N}.B \Rightarrow b_k \in [0, 1]$
- Minimize the number of iterations:
  1. Choose B_0
  2. Compute the decoding threshold

- $B = [b_0, b_1 \dots b_{m_s}]$ où $b_i \in \{0 \dots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$
- $B = \frac{1}{N}.B \Rightarrow b_k \in [0, 1]$
- Minimize the number of iterations:
  1. Choose B_0
  2. Compute the decoding threshold
  3. Find a coupling B that minimizes the iterations at the same threshold

- $B = [b_0, b_1 \ldots b_{m_s}]$ où $b_i \in \{0 \ldots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$
- $B = \frac{1}{N}.B \Rightarrow b_k \in [0, 1]$
- Minimize the number of iterations:
  1. Choose B_0
  2. Compute the decoding threshold
  3. Find a coupling B that minimizes the iterations at the same threshold
     - Define $\{\{b_k\}_k | \sum b_k = 1 \text{ et } b_k \in [0, 1]\}$

- $B = [b_0, b_1 \dots b_{m_s}]$ où $b_i \in \{0 \dots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$
- $B = \frac{1}{N}.B \Rightarrow b_k \in [0, 1]$
- Minimize the number of iterations:
  1. Choose B_0
  2. Compute the decoding threshold
  3. Find a coupling B that minimizes the iterations at the same threshold
     - Define $\{\{b_k\}_k | \sum b_k = 1 \text{ et } b_k \in [0, 1]\}$
     - Differential evolution

- $B = [b_0, b_1 \dots b_{m_s}]$ où $b_i \in \{0 \dots n\}$ et avec $\sum_{i=1}^{m_s} b_i = N$
- $B = \frac{1}{N}.B \Rightarrow b_k \in [0,1]$
- Minimize the number of iterations:
  1. Choose B_0
  2. Compute the decoding threshold
  3. Find a coupling B that minimizes the iterations at the same threshold
     - Define $\{\{b_k\}_k | \sum b_k = 1 \text{ et } b_k \in [0,1]\}$
     - Differential evolution
     - Simplification: non symmetric inequality $\sum b_k < 1$, B

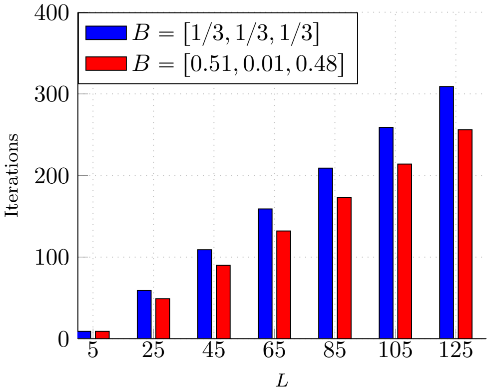


Figure: Iterations before convergence of a binary and octal SC-CPM schemes with syndrome former memory $m_s = 2$

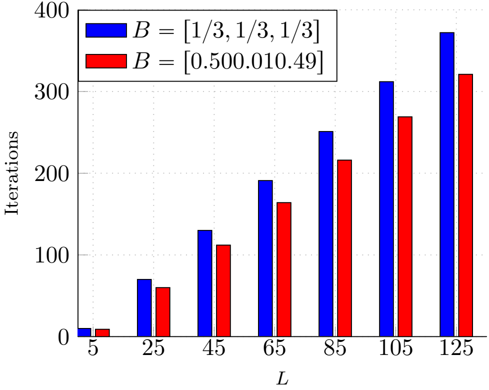


Figure: Iterations before convergence of a binary and octal SC-CPM schemes with syndrome former memory $m_s = 2$

	C-CPM	SC-CPM	LDPC	EXIT Area
Binary CPM	1.11	0.43	0.56	0.24
Quaternary CPM	1.07	0.26	0.20	-0.01
Octal CPM	0.72	-0.16	-0.20	-0.37

Table: Thresholds E_b/N_0 of different schemes at rates close to 1/2. For SC-CPM, $B = [\frac{1}{2}, \frac{1}{2}]$

Thank you for your attention!