

Optimization of Read Thresholds in MLC NAND Memory for LDPC Codes

YISHEN YEH

TABLE OF CONTENT

- I. Read Thresholds in MLC NAND**
- II. Optimization via Maximum Mutual Information (MMI)**
- III. Optimization via Density Evolution**



I. Preliminary

Read Thresholds in MLC NAND



NAND Flash Memory and Read Thresholds

Floating Gate Transistor

- **Write:**
 - We “write” a symbol to a transistor by storing a specific voltage μ_i in the floating gate,
 $\mu_i \in [-3, -1, 1, 3] \equiv ['11', '10', '00', '01']$
- **Read:**
 - We “read” the voltage by applying a threshold γ to the transistor, we receive a binary response whether the stored voltage is higher or lower than γ
- **Noise:**
 - Throughout many P/E cycles, the stored voltages will shift and become distorted. We model the noise as AWGN,
 - Note that in real life, the noise is not Gaussian, nor is it identically distributed.

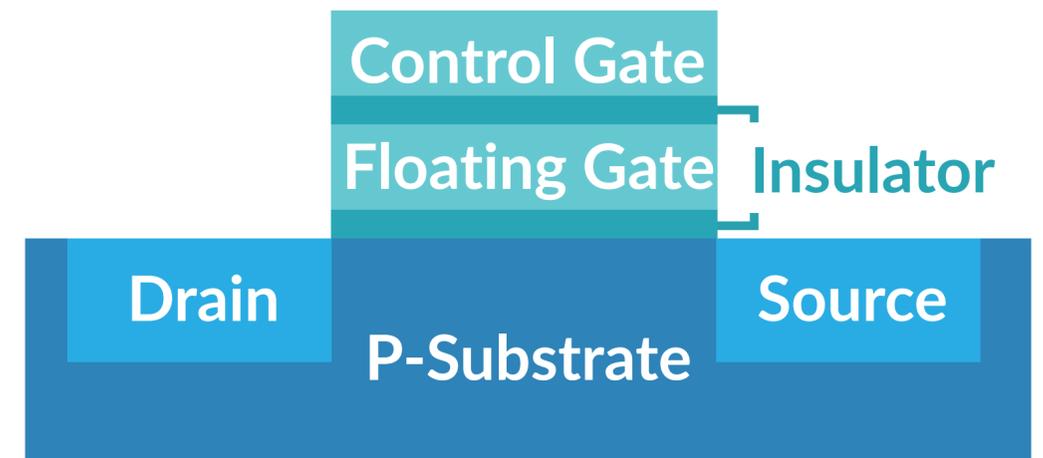


Fig. 1: Floating Gate Transistor



NAND Flash Memory and Read Thresholds

Optimization Goal

- Challenge:
 - Optimize the number of reads (m) subject to a minimum bit error rate (BER) gain per additional threshold
 - Optimize the position of the reads that gives the best performance in terms of BER
- Optimization Criteria:
 - Maximizing Mutual Information (MMI)
 - Density Evolution (DE)

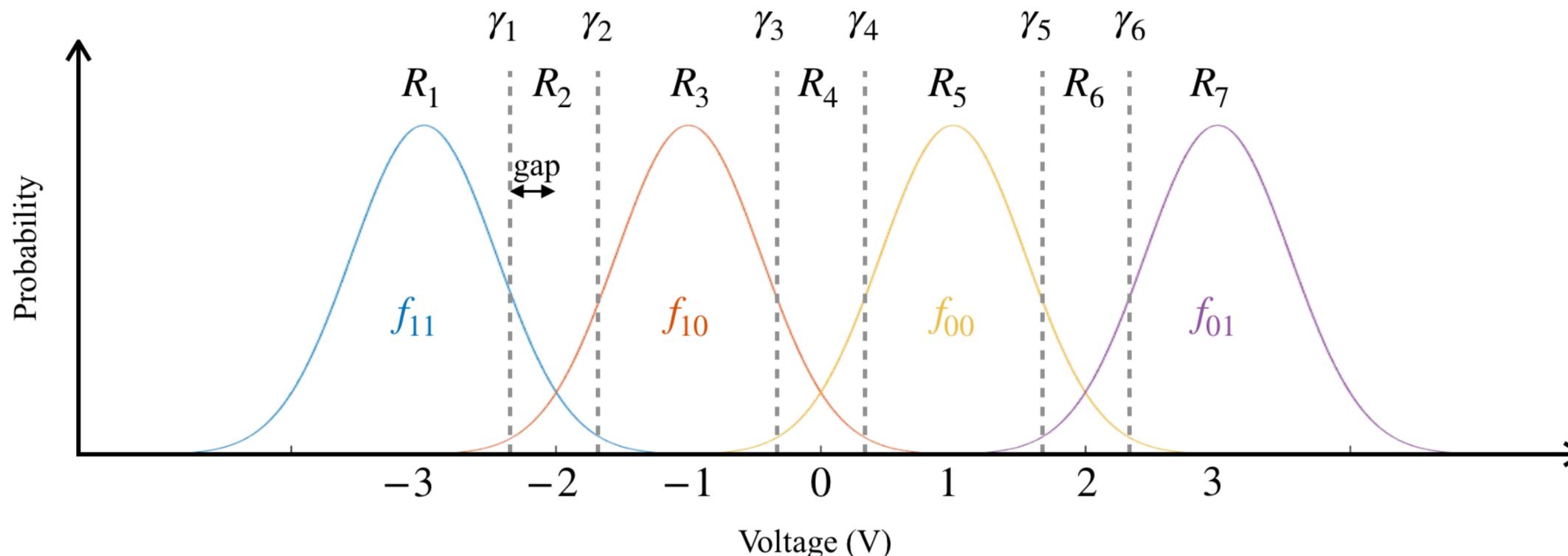


Fig. 2: Illustration of threshold voltages and voltage distributions for a MLC NAND flash memory with $m = 6$ read thresholds



NAND Flash Memory and Read Thresholds

Gap

- We consider $m = 6$ case and set the thresholds initially at positions $[\gamma_1, \gamma_2, \dots, \gamma_6] = [-2, -2, 0, 0, 2, 2]$
- Gaps are defined as the shifts in voltages away from the initial positions
- Positive gap is decrease in voltages for odd thresholds and increase for even thresholds
- For example if all gaps = 0.4V, $[\gamma_1, \gamma_2, \dots, \gamma_6] = [-2.4, -1.6, -0.4, 0.4, 1.6, 2.4]$

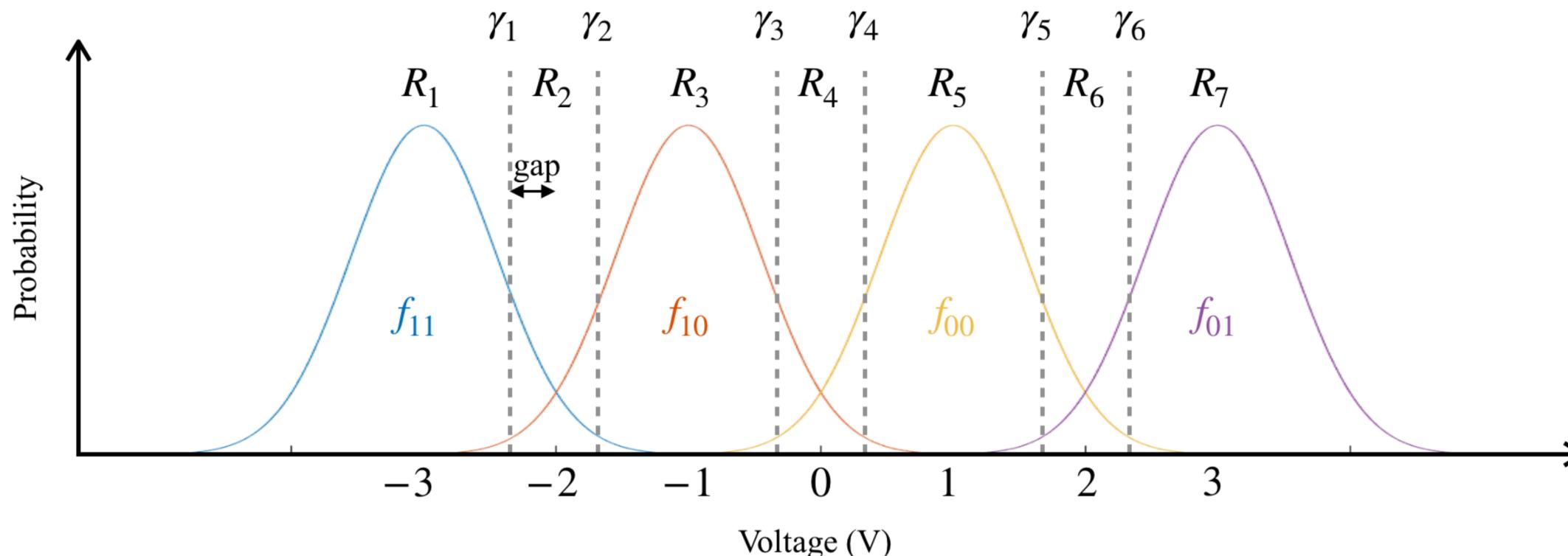


Fig. 2: Illustration of threshold voltages and voltage distributions for a MLC NAND flash memory with $m = 6$ read thresholds



II. MMI

Optimization by Maximizing Mutual
Information



Mutual Information

Definitions

- Mutual information is given by [1]:

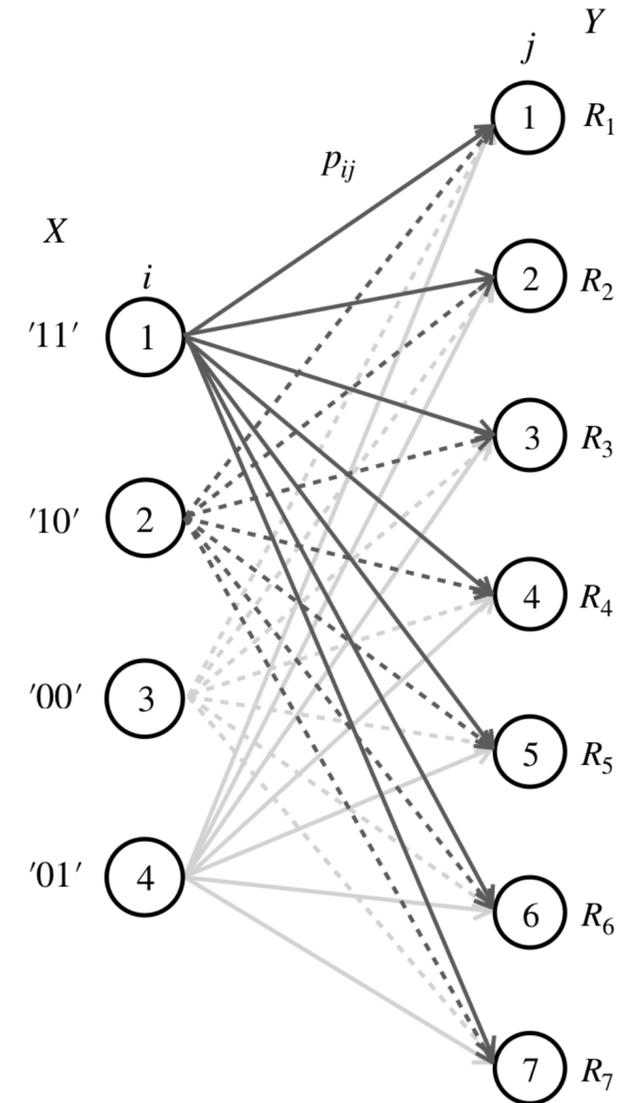
$$I(X; Y) = H(Y) - H(X|Y)$$

$$= H\left(\frac{p_{11} + p_{21} + p_{31} + p_{41}}{4}, \frac{p_{12} + p_{22} + p_{32} + p_{42}}{4}, \dots, \frac{p_{17} + p_{27} + p_{37} + p_{47}}{4}\right) - \frac{1}{4} \left[H(p_{11}, p_{12}, \dots, p_{17}) + H(p_{21}, p_{22}, \dots, p_{27}) + \dots + H(p_{71}, p_{72}, \dots, p_{77}) \right]$$

where $H(\cdot)$ denotes the entropy function, $p_{ij} = \Pr\{\mathbf{X} = \mu_i | \mathbf{Y} \in R_j\}$,

$\mu_i \in [-3, -1, 1, 3]$ and $j \in [1, m+1]$

[1]. J. Wang *et al*, “Enhanced precision through multiple reads for LDPC de- coding in flash memories,” *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 5, pp. 880–891, April 2014





Find MMI using Gradient Descent (GD)

Set up

- We let our cost function be:

$$g(\bar{\gamma}) = I(X; Y)$$

- Gradient descent (GD) algorithm gives us:

$$\bar{\gamma}^{(l+1)} = \bar{\gamma}^{(l)} + \mu \cdot \nabla g(\bar{\gamma}^{(l)})$$

where $\bar{\gamma} = [\gamma_1, \gamma_2, \dots, \gamma_m]^T$, μ is the step size, $\bar{\gamma}^{(l)}$ denotes the vector of read thresholds at iteration l .

- Consequently, the goal becomes to maximize mutual information (MMI):

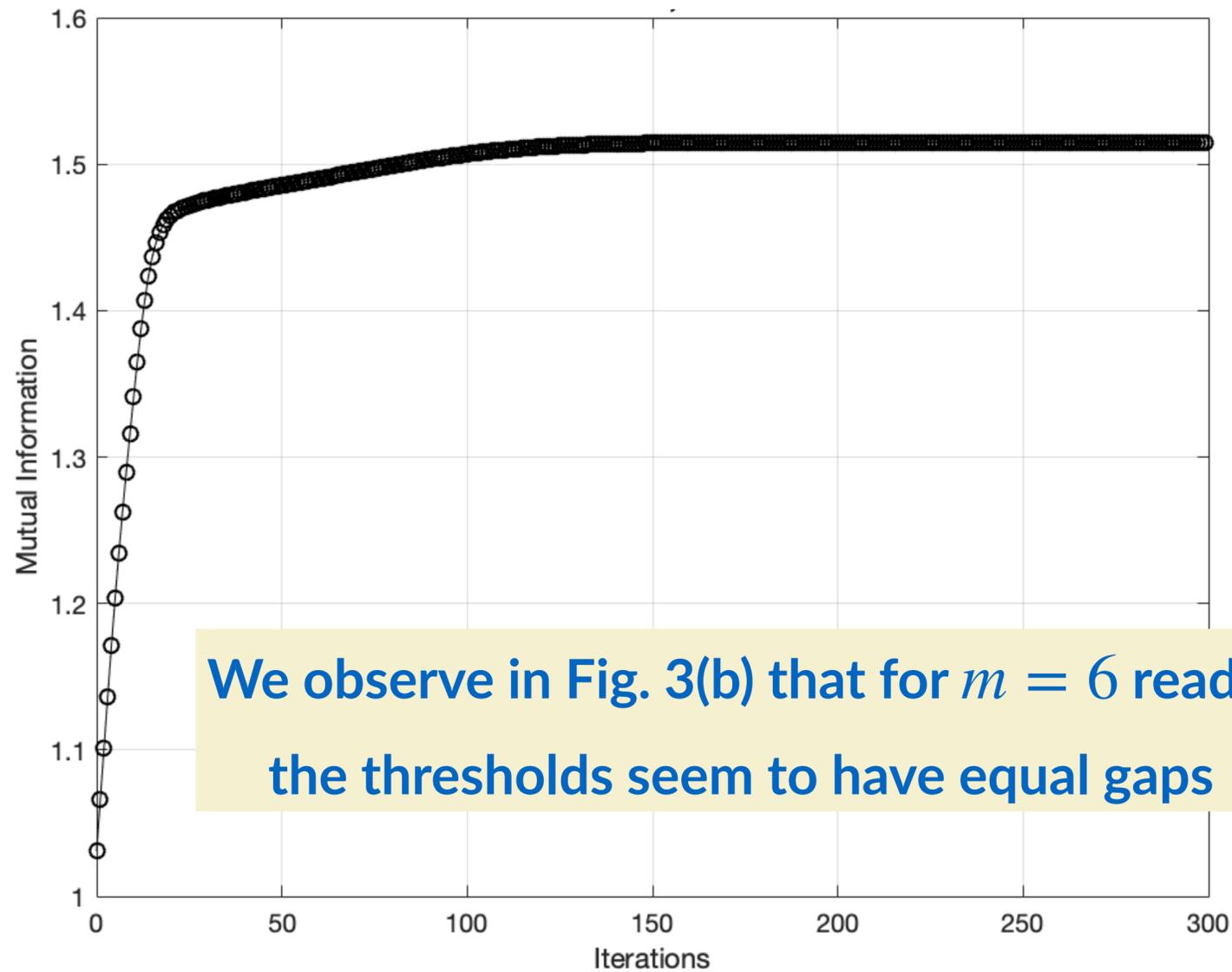
$$\max_{\bar{\gamma}} g(\bar{\gamma}) \quad \text{with } -\infty < \gamma_i < \infty \text{ for } 1 \leq i \leq m.$$



Find MMI using Gradient Descent

Results

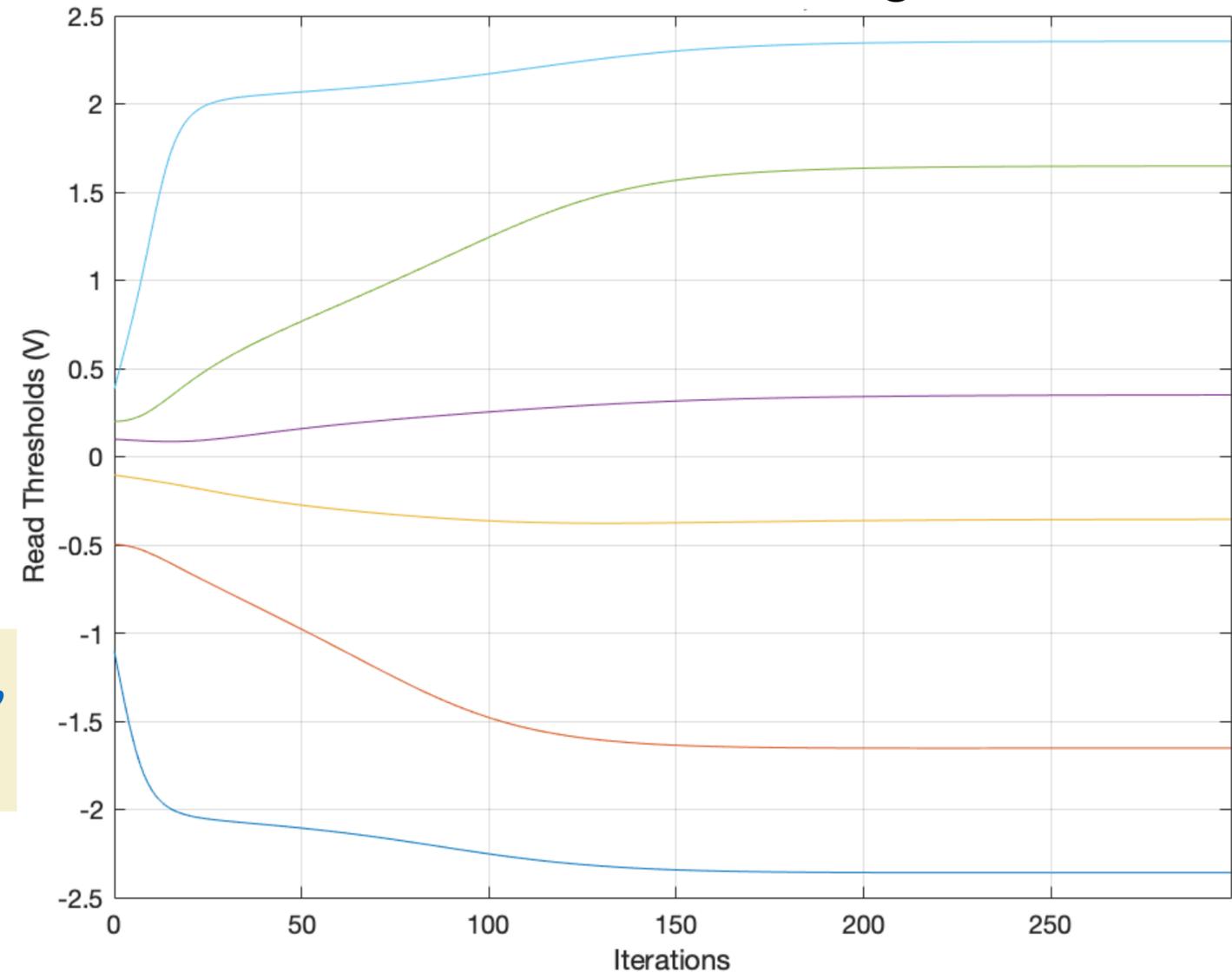
Mutual Information for each iteration



We observe in Fig. 3(b) that for $m = 6$ reads, the thresholds seem to have equal gaps

(a)

Position of thresholds throughout GD



(b)

Fig. 3: Gradient descent iterations adjusting all 6 thresholds simultaneously without the assumption of equal gaps. (a). The mutual information evolution corresponding to each gradient descent iterations when $\mu = 0.5$; (b). The evolution of the positions of read thresholds through the gradient descent algorithm iterations;



Find MMI using Gradient Descent

Results

Optimal gaps for $m = 6$ and SNR 0~18

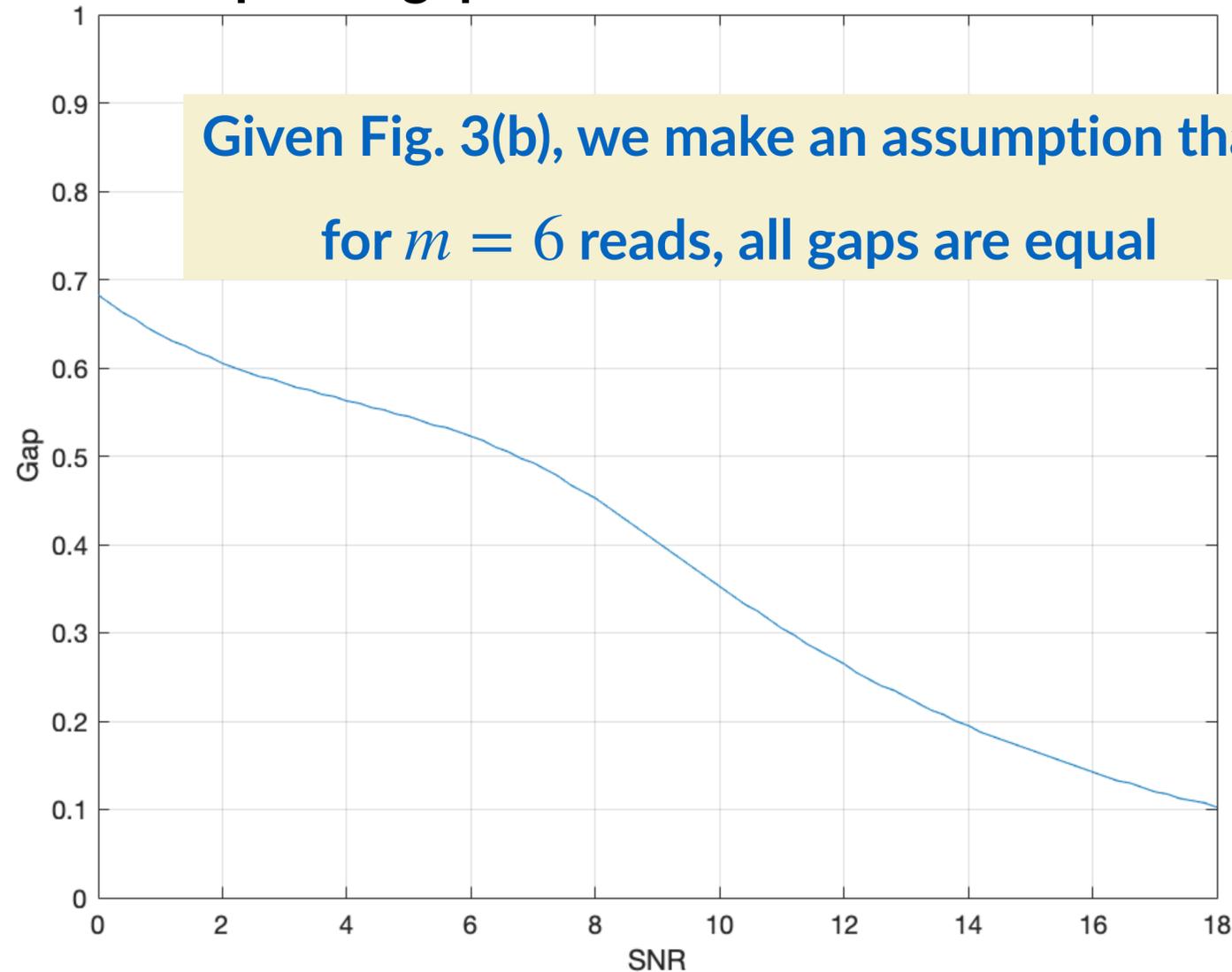


Fig. 4: Optimal position of $m = 6$ read thresholds at SNR 0 to 18 dB obtained through gradient descent algorithm.

MMI achievable for different number of reads

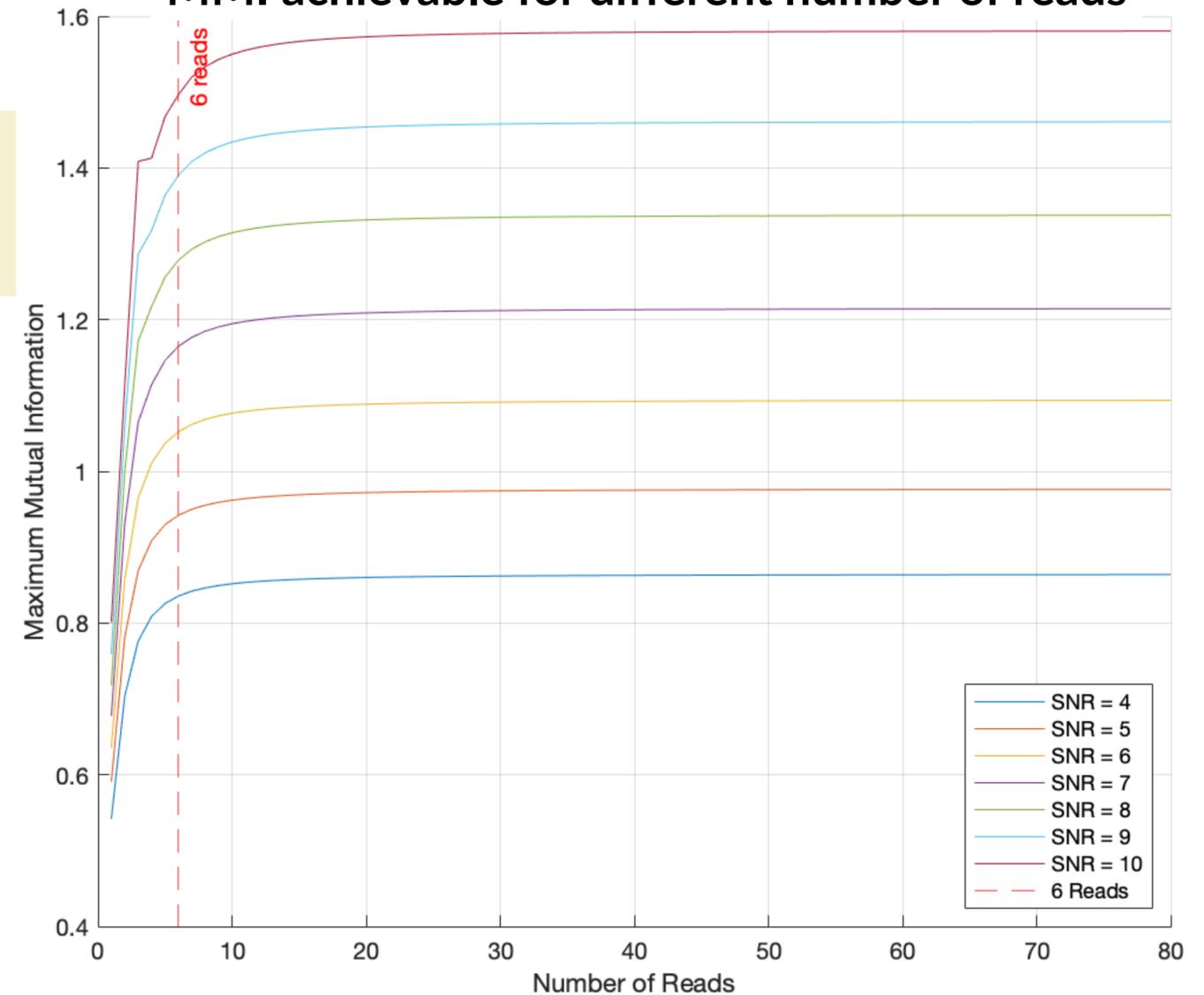


Fig. 5: MMI achievable for $m = 2$ to 80, with the threshold positions optimized by gradient descent algorithm.



III. LDPC & DE

Optimizing performance using Density
Evolution



Density Evolution (DE)

Set up

- We use DE, with a fixed finite number of iterations, as a proxy to find the BER and thereby optimize the read thresholds for MLC
- On the right, we visualize the evolution of LLR densities. They shift towards the right as number of iterations increases
- One major assumption is that we're assuming all 0 codeword is transmitted. Therefore, the error rate is the integration of the densities from $-\infty$ to 0

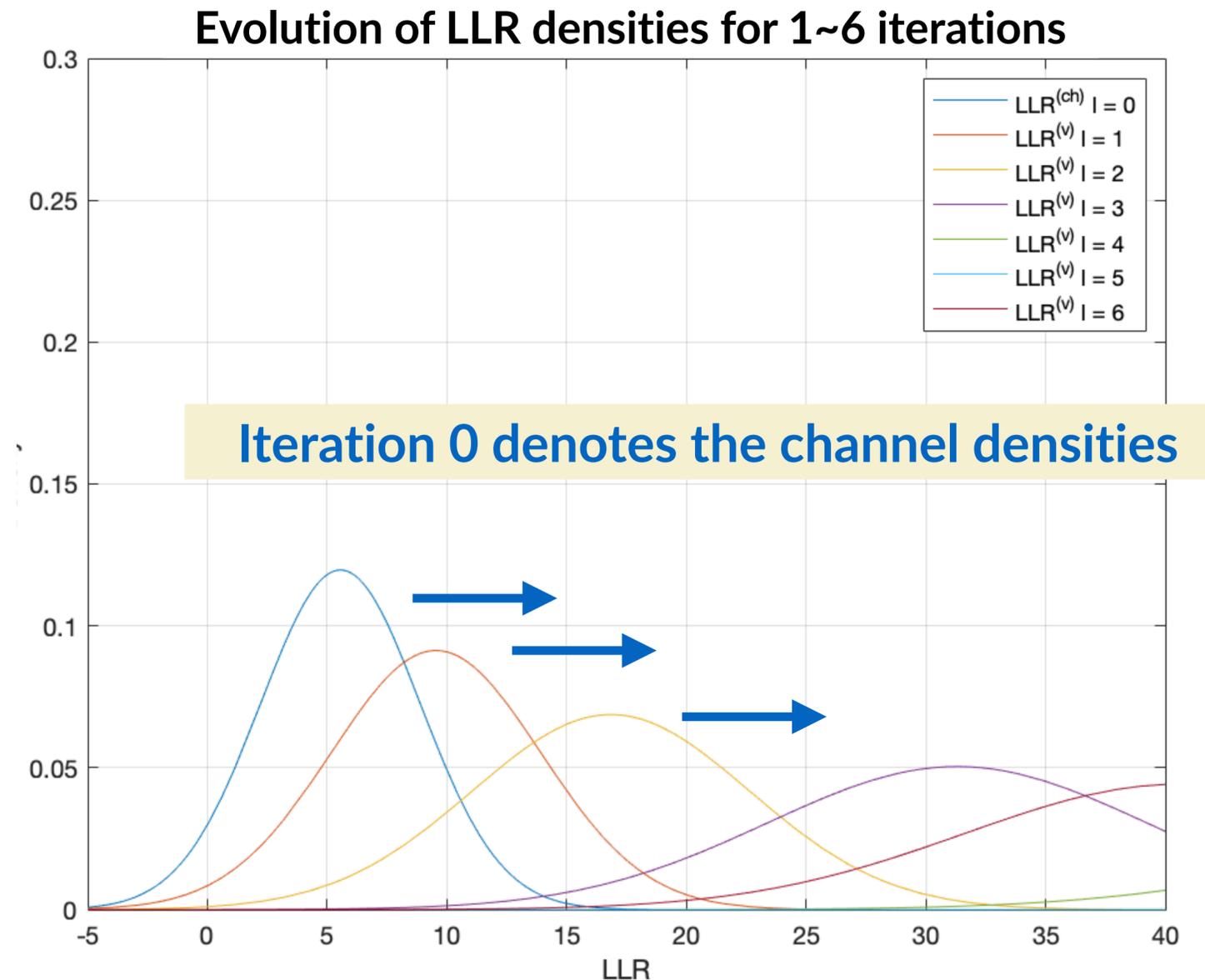


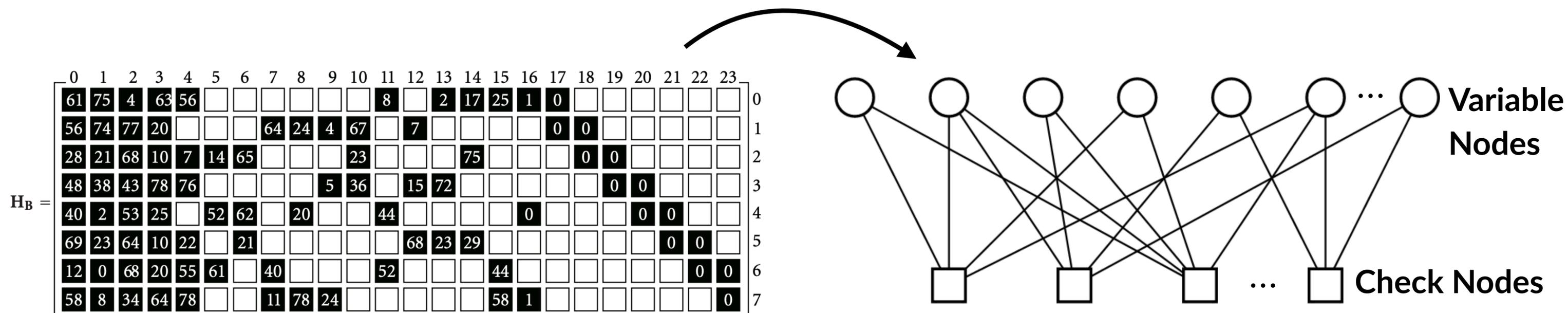
Fig. 6: Illustration of density evolution given a degree distribution of a regular QC-LDPC code.



LDPC Codes & Density Evolution

802.11n uses LDPC codes of rate 2/3 and length 1944

- We use the 802.11n WiFi LDPC code to test the performance of given thresholds.



r 81×81 identity matrix rotated by r \square 81×81 zero matrix

Base Matrix

[6] Rovini, Massimo & Gentile, Giuseppe & Rossi, Francesco & Fanucci, Luca. (2009). Techniques and Architectures for Hazard-Free Semi-Parallel Decoding of LDPC Codes. EURASIP J. Emb. Sys.. 2009. 10.1155/2009/723465.



Discretized Density Evolution

Math

- The discretized density evolution can be described as:

$$p_u^{(t+1)} = \rho(p_{u_0} \otimes \lambda(p_u^{(t)}))$$

where $\rho(\cdot)$ and $\lambda(\cdot)$ are described in [3], \otimes is the discrete convolution, p_{u_0} and p_u are the channel pmf and the initial check node pmf with mass concentrated at 0 respectively.

- The cost function of our optimization problem at iteration t becomes:

$$P_e^{(t)} = \sum_{k=-\infty}^0 p_v^{(t)}[k]$$

[3] S. Chung, G. D. Forney, T. J. Richardson, and R. Urbanke, "On the design of low-density parity-check codes within 0.0045dB of the Shannon limit," *IEEE Communications Letters*, vol. 5, no. 2, pp. 58–60, February 2001.



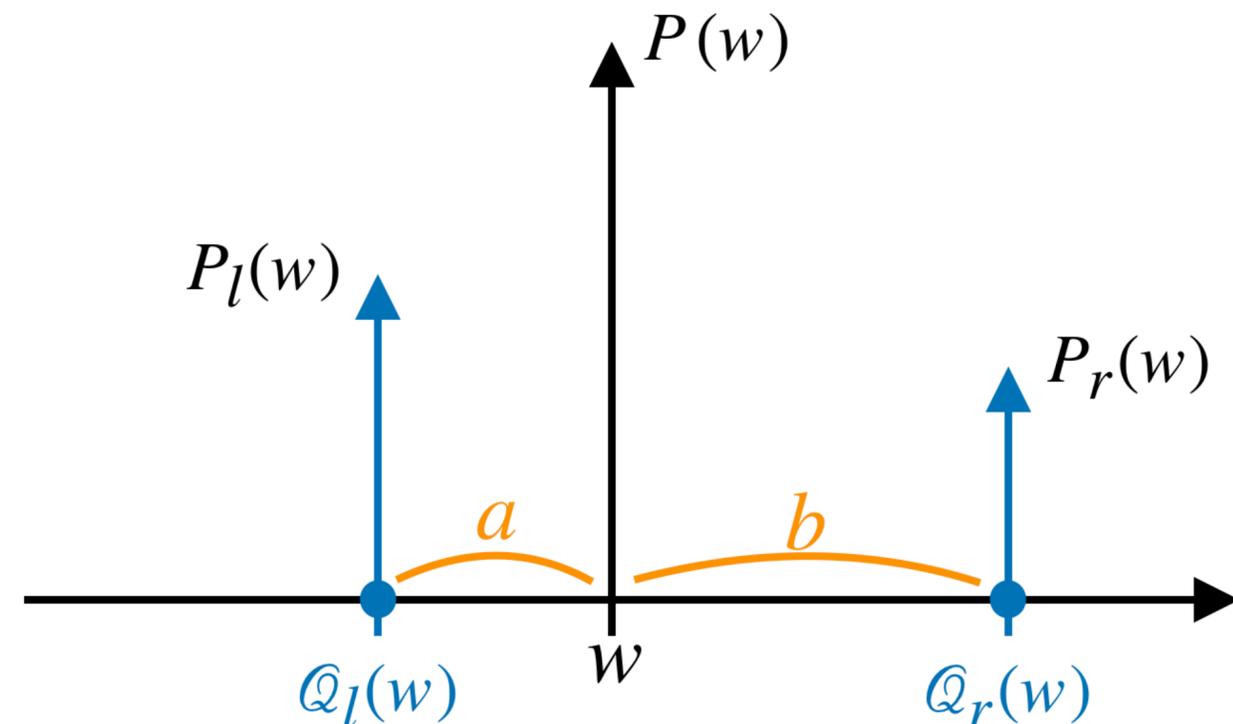
Split Ratio Quantization

To smooth out the discrete jumps

- The quantization method given in [3] is:

$$Q(w) = \begin{cases} \left\lfloor \frac{w}{\Delta} + \frac{1}{2} \right\rfloor \cdot \Delta, & \text{if } w \geq \frac{\Delta}{2} \\ \left\lfloor \frac{w}{\Delta} - \frac{1}{2} \right\rfloor \cdot \Delta, & \text{if } w \leq \frac{\Delta}{2} \\ 0, & \text{otherwise} \end{cases}$$

where Δ is the quantization interval.



- This method will result in unwanted “jumps” in the performance curves which we'll show in later pages. So we introduce the split ratio quantization (SRQ) on top of it to smooth out the transitions at quantization points.
- If the original data point is at position w with value $P(w)$. The new quantized data becomes at position

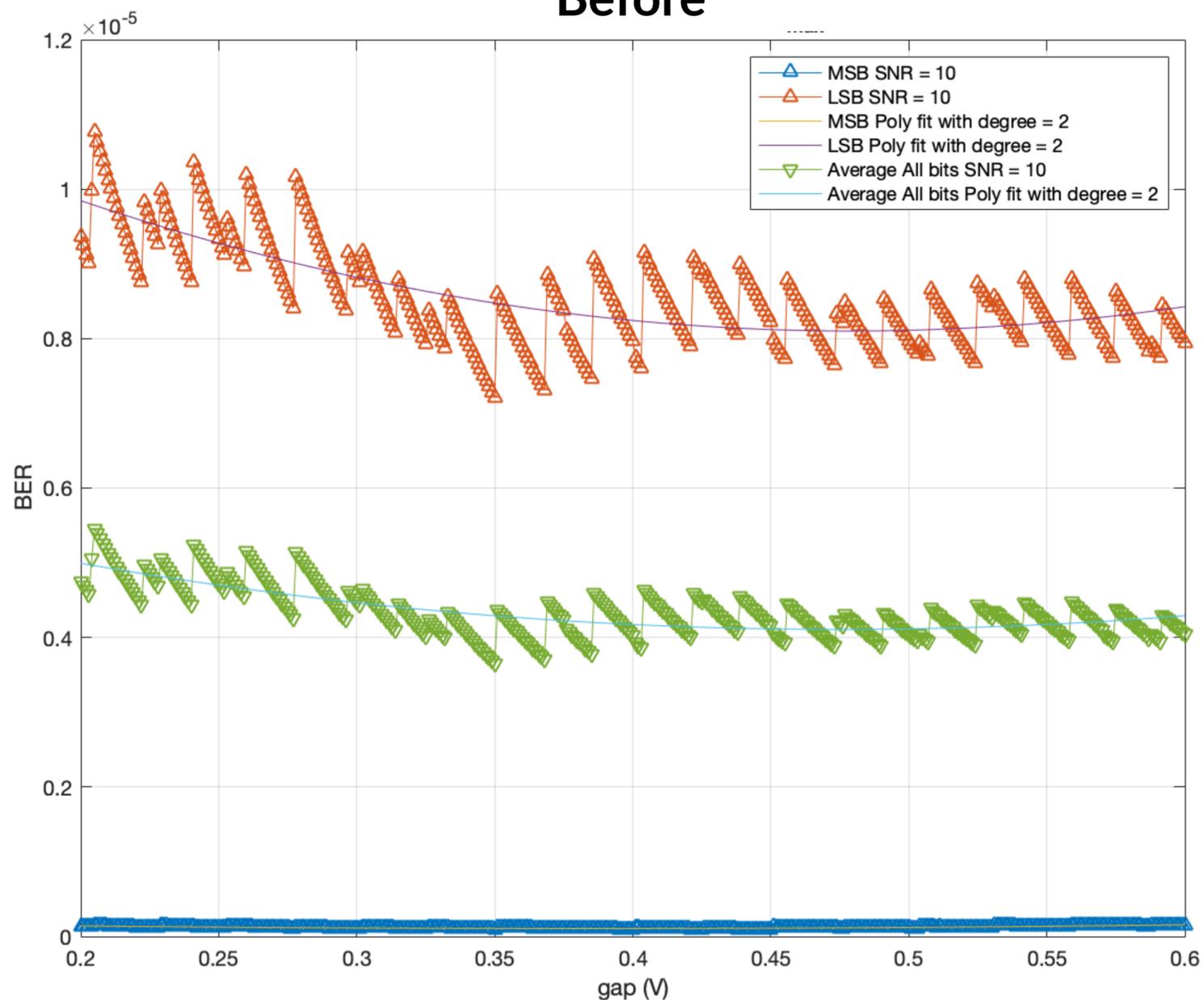
$$Q_l(w) \text{ and } Q_r(w) \text{ with values } P_l(w) = \frac{b}{(a+b)}P(w) \text{ and } P_r(w) = \frac{a}{(a+b)}P(w) \text{ respectively}$$



Split Ratio Quantization

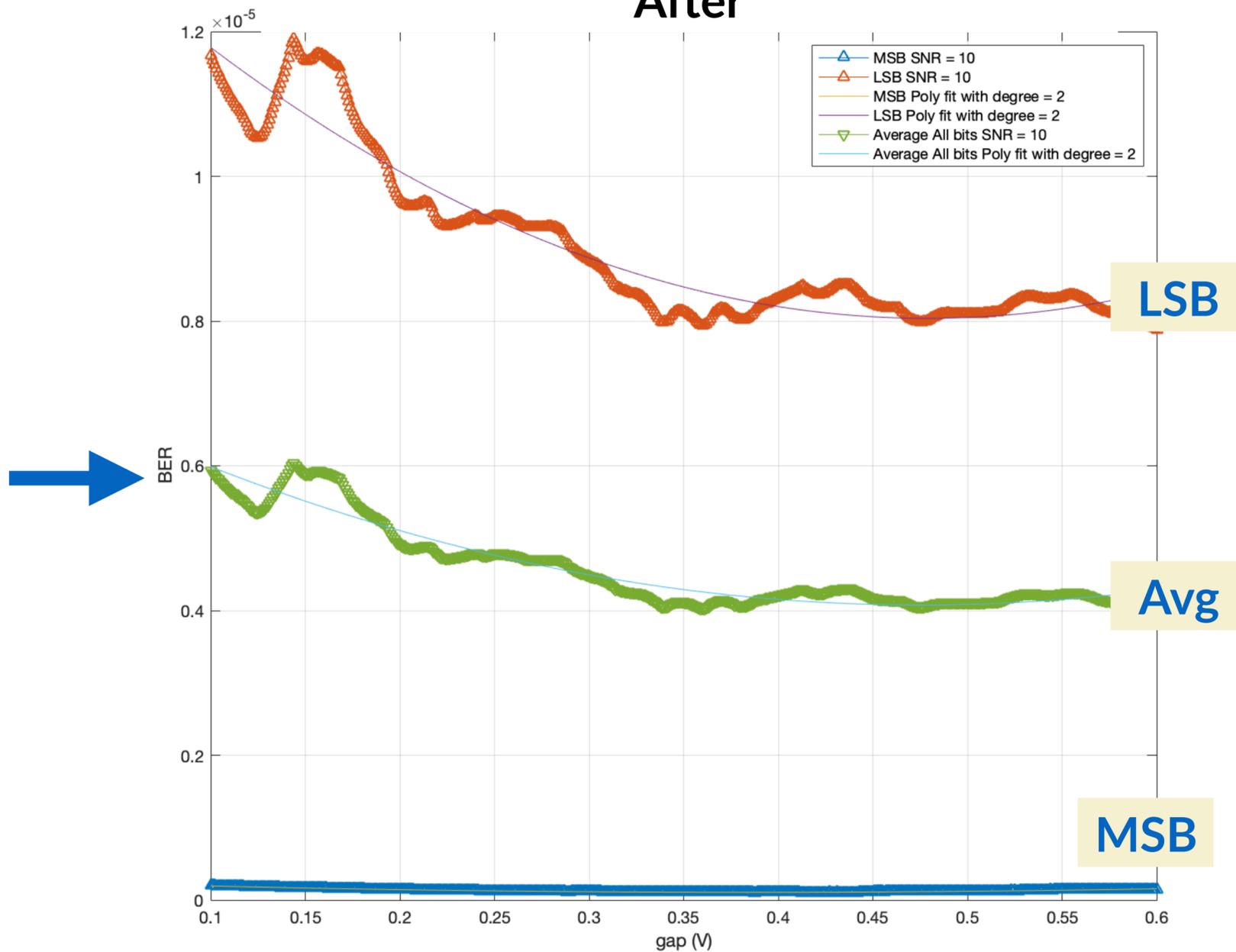
Effects

Before



(a)

After



(b)

Fig. 7: The exhaustive search of density evolution for $m = 6$, SNR = 10(dB) with the assumption of equal gaps; (a) without SRQ and (b) with SRQ.



Fixed Point Exhaustive Search

Zooming in on individual thresholds' optimal gaps

- Due to the remaining fluctuations in Fig. 7(b), we suspect that the optimal threshold positions do not have equal gaps ($m = 6$). Thus we conduct a “Fix point exhaustive search”, where we do an exhaustive search for the optimal gap for one threshold while fixing all other thresholds to 0.5V.

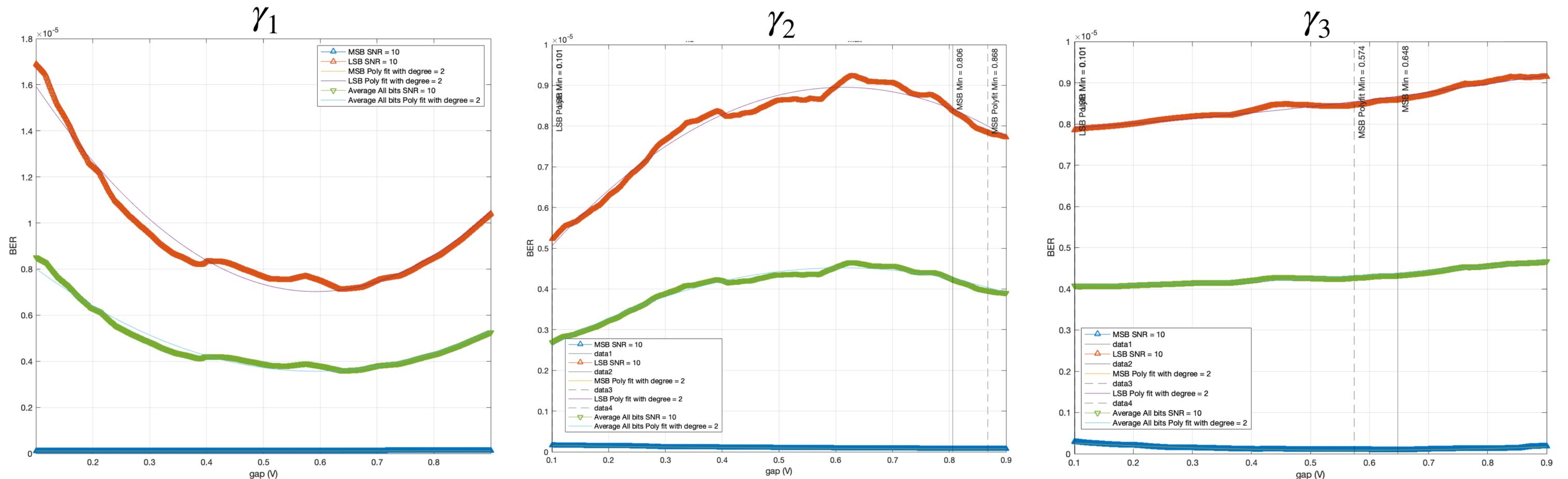


Fig. 8: The fixed point exhaustive search the first 3 thresholds ($m = 6$); Note, the last 3 thresholds are symmetric to the first 3.



Fixed Point Exhaustive Search

Now confirm with actual code implementation

- We double check our result with actual LDPC codes under BP decoding. We see that the characteristics of the performance curves are identical, which proves the validity of using DE.
- One observation is that the LSB performance is the dominant curve effects the overall curve shape
- Also, we argue that the position of third threshold γ_3 does NOT have any significance over LSB performance.

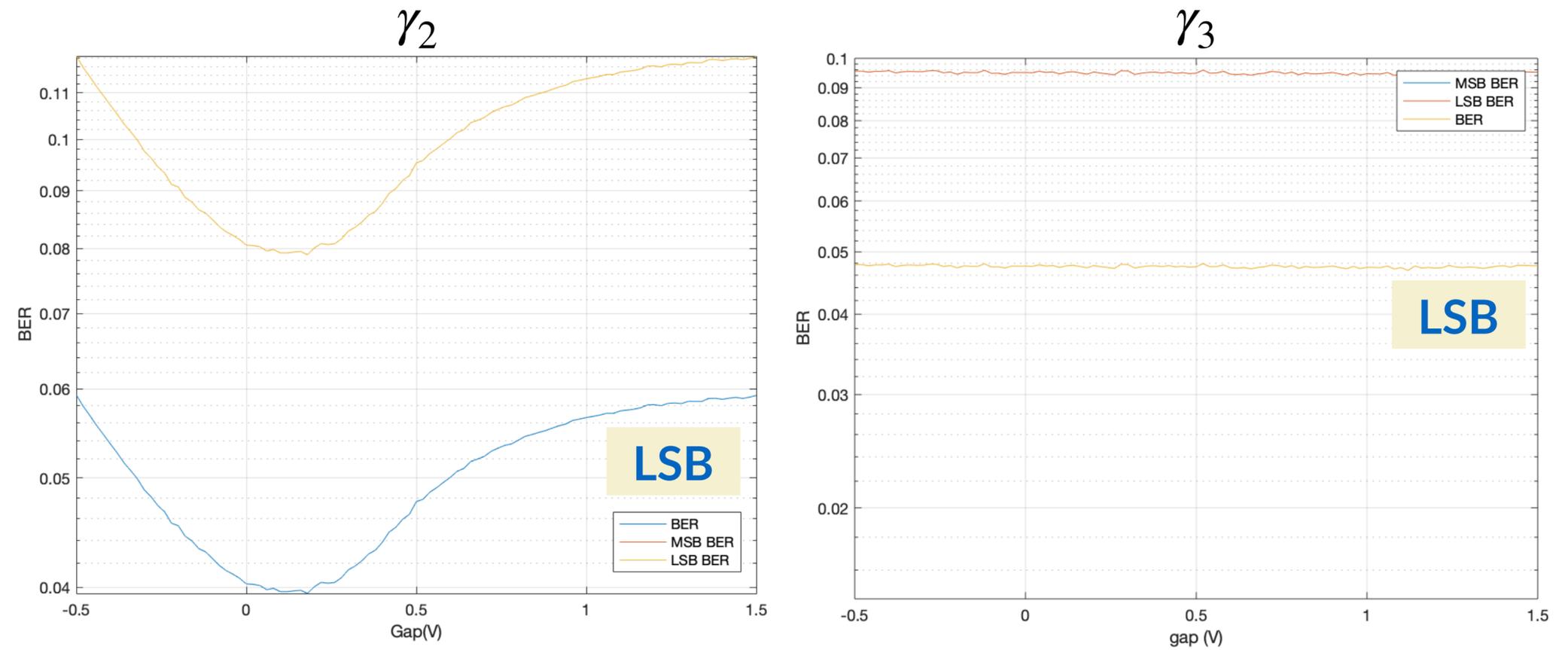


Fig. 9: The fixed point exhaustive search using actual LDPC codes



Apply Gradient Descent on DE

Setup $m = 6$, SNR = 10.8dB

- Because of the previous observations, we decide to optimize the thresholds $[\gamma_1, \gamma_2, \gamma_5, \gamma_6]$ to minimize the LSB BER using DE. Meanwhile, we fix $[\gamma_3, \gamma_4] = [-0.3, 0.3]$ for SNR = 10.8 dB.
- Note that here we do not have the constraint of equal gaps. $[\gamma_1, \gamma_2, \gamma_5, \gamma_6]$ are able to be adjust freely by the gradient descent algorithm individually.
- The goal becomes to minimize BER of LSB:

$$\min_{[\gamma_1, \gamma_2, \gamma_5, \gamma_6]} P_{e,LSB} \quad \text{with } \gamma_i \leq \gamma_j \text{ for all } i < j \text{ and } [\gamma_3, \gamma_4] = [-0.3, 0.3]$$

Apply Gradient Descent on DE

Results

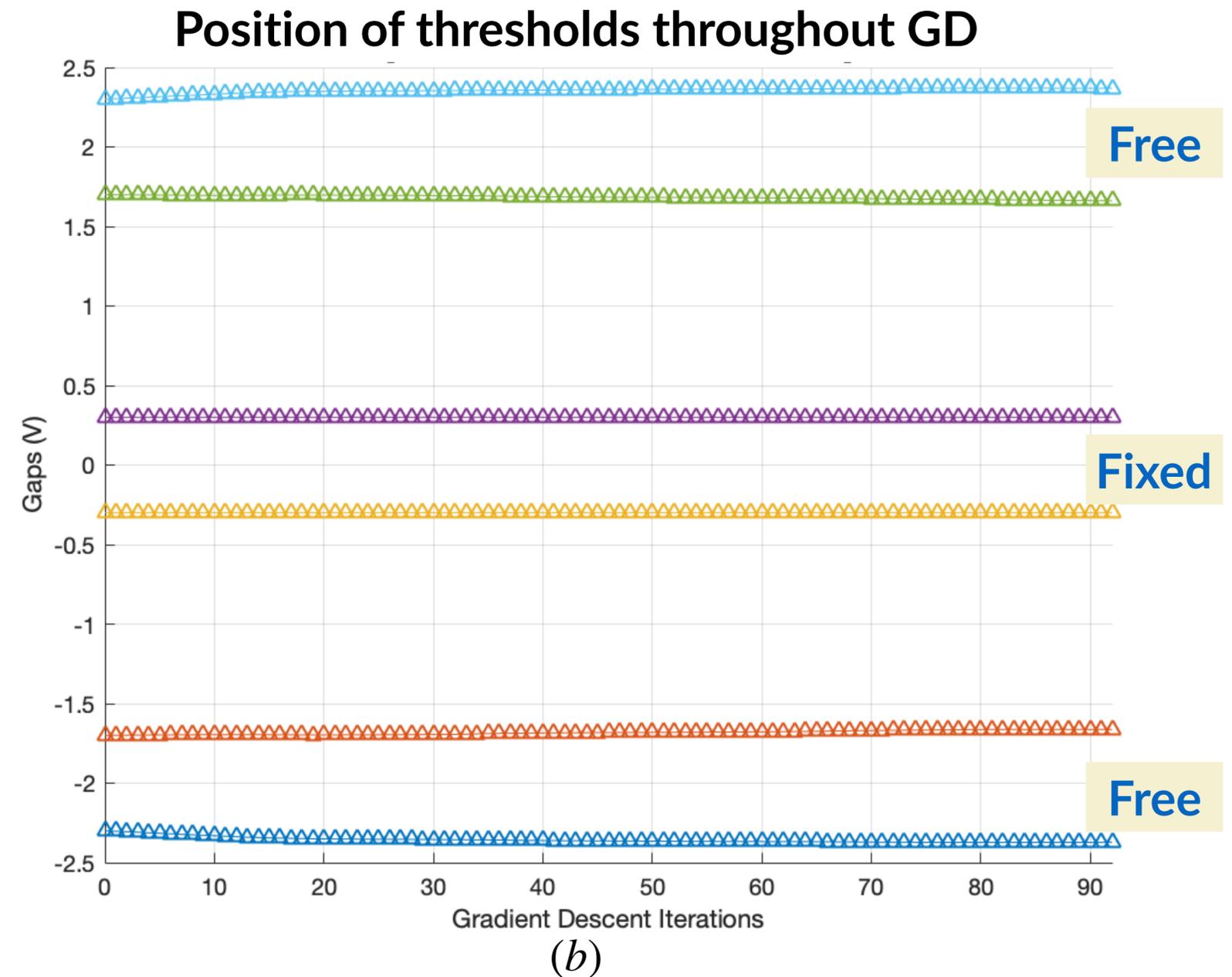
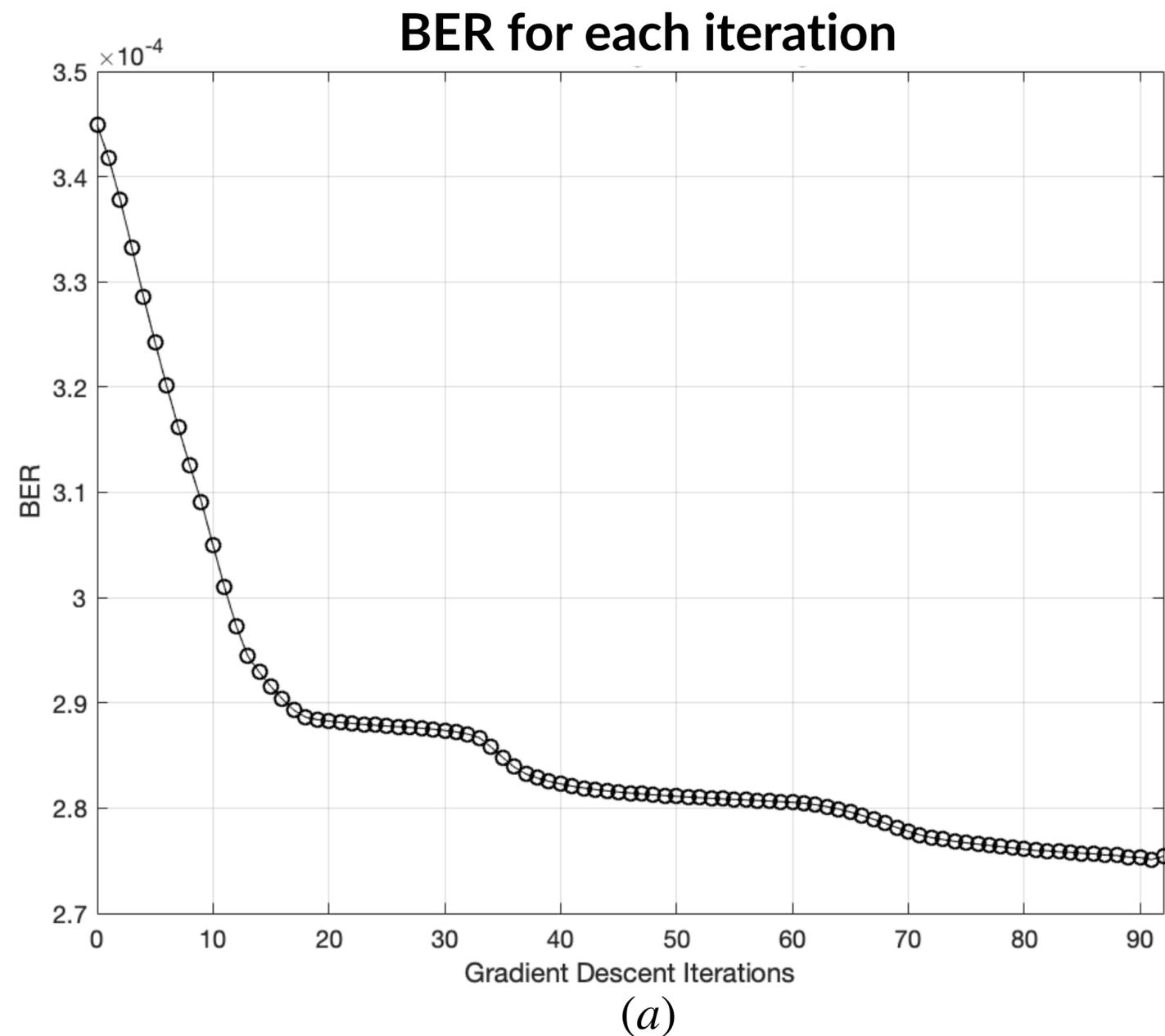


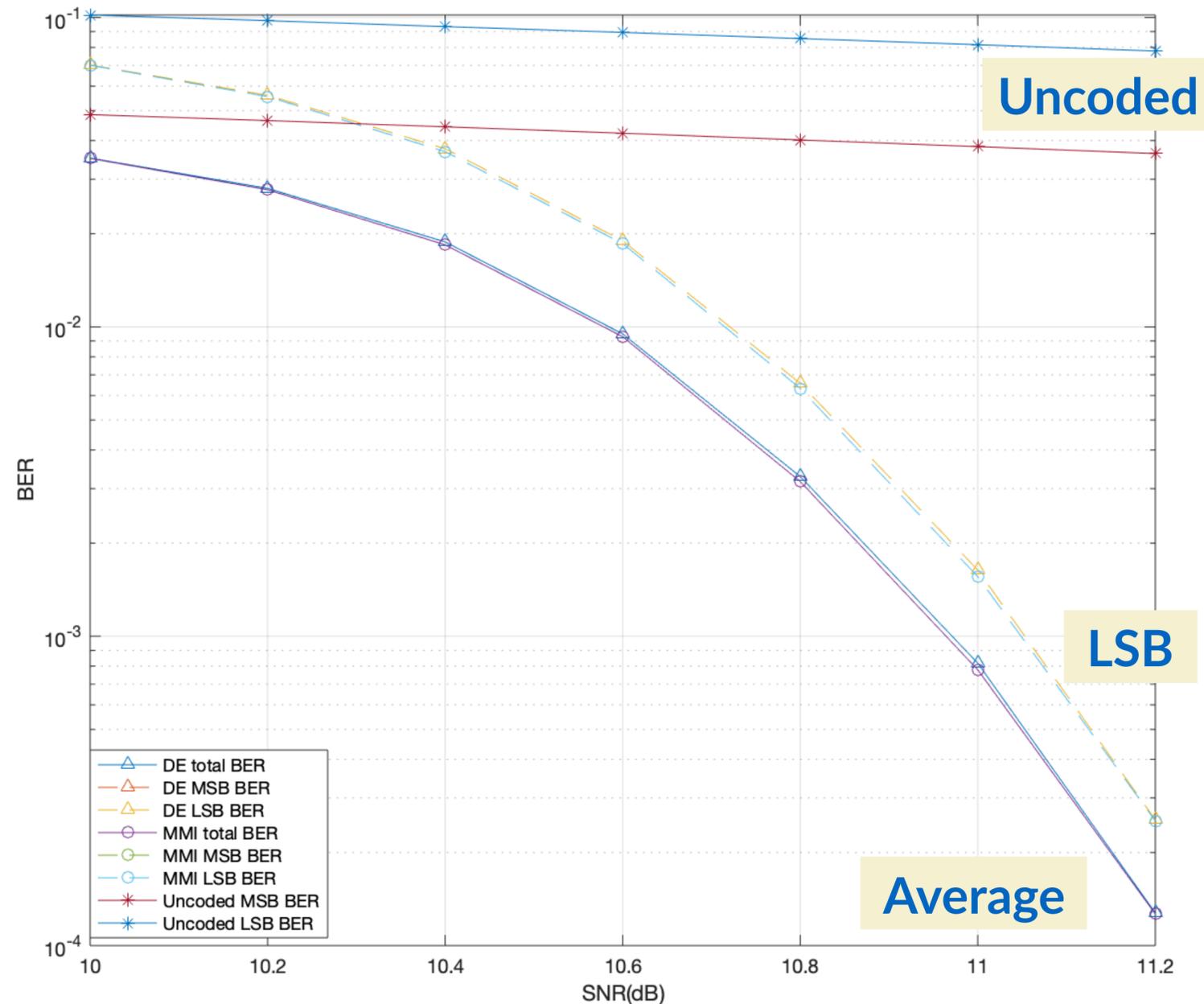
Fig. 10: Gradient descent algorithm with density evolution fixing γ_3 and γ_4 to -0.3V and 0.3V respectively. The results are $[\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6] = [-2.3717, -1.6632, -0.3, 0.3, 1.6632, 2.3717]$.



Comparing MMI and DE

Performance given by actual LDPC encoding/decoding

BER for each SNR

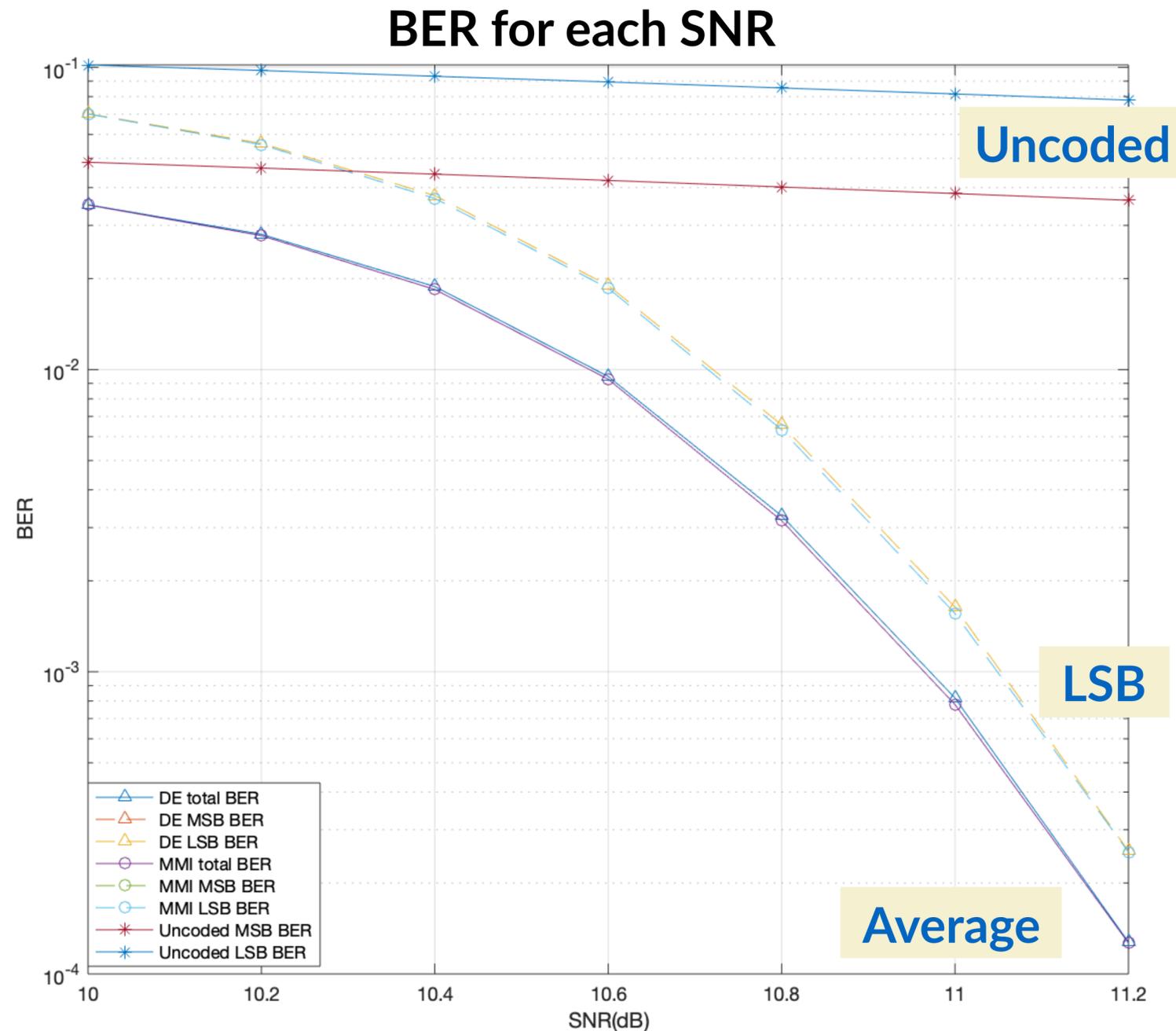


- We see that the DE given threshold positions performs as well as MMI given threshold positions.
- DE given thresholds :
[−2.3717, − 1.6632, − 0.3, 0.3, 1.6632, 2.3717]
- MMI given thresholds :
[−2.315, − 1.685, − 0.3, 0.3, 1.685, 2.315]

Fig. 11: Comparison of DE given thresholds $[\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6] = [-2.3717, -1.6632, -0.3, 0.3, 1.6632, 2.3717]$ and MMI given thresholds $[\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6] = [-2.315, -1.685, -0.3, 0.3, 1.685, 2.315]$ with actual code implementation.



Conclusion



We believe that as we go beyond $m = 6$ case and higher SNRs, we'd expect DE given thresholds to outperform those given by MMI because that DE also accounts for the fact that LSB plays a more significant role in terms of BER. As well as the fact that DE are specifically design to accommodate specific LDPC codes

Fig. 11: Comparison of DE given thresholds $[\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6] = [-2.3717, -1.6632, -0.3, 0.3, 1.6632, 2.3717]$ and MMI given thresholds $[\gamma_1, \gamma_2, \gamma_3, \gamma_4, \gamma_5, \gamma_6] = [-2.315, -1.685, -0.3, 0.3, 1.685, 2.315]$ with actual code implementation.



References

- [1]. J. Wang *et al*, “Enhanced precision through multiple reads for LDPC decoding in flash memories,” *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 5, pp. 880–891, April 2014
- [2] C. Duangthong, W. Phakphisut, and P. Supnithi, “Read Voltage Optimization in MLC NAND Flash Memory via the Density Evolution,” *Proceedings of IEEE International Conference on Telecommunications (ICT)*, pp. 361–365, April 2019.
- [3] S. Chung, G. D. Forney, T. J. Richardson, and R. Urbanke, “On the design of low-density parity-check codes within 0.0045dB of the Shannon limit,” *IEEE Communications Letters*, vol. 5, no. 2, pp. 58–60, February 2001.
- [4] A. Kalsi, A. Bajpai, L. Wuttisittikulij, and P. Kovintaewat, “A base matrix method to construct column weight 3 quasi-cyclic LDPC codes with high girth,” *Proceedings of International Conference on Electronics, Information, and Communications (ICEIC)*, pp. 1–4, January 2016.
- [5] T. Richardson and R. Urbanke, “Modern Coding Theory,” Cambridge University Press, 2008.
- [6] Rovini, Massimo & Gentile, Giuseppe & Rossi, Francesco & Fanucci, Luca. (2009). Techniques and Architectures for Hazard-Free Semi-Parallel Decoding of LDPC Codes. *EURASIP J. Emb. Sys.*. 2009. 10.1155/2009/723465.