IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 13 June 2023, accepted 13 July 2023, date of publication 17 July 2023, date of current version 25 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3296250

METHODS

Efficient Design of Read Voltages and LDPC Codes in NAND Flash Memory Using Density Evolution

CHATUPORN DUANGTHONG, WATID PHAKPHISUT, AND PARAMOTE WARDKEIN
School of Engineering, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand

Corresponding author: Watid Phakphisut (watid.ph@kmitl.ac.th)

This work was supported by the National Science, Research and Innovation Fund (NSRF) through the Research Project " Efficient Design of Read Voltages and LDPC codes in NAND Flash Memory Using Density Evolution'' by the King Mongkut's Institute of Technology Ladkrabang (KMITL) under Grant FRB660065/0258-RE-KRIS/FF66/66.

ABSTRACT Low-density parity-check (LDPC) codes play an important role in the reliability enhancement of commercial NAND flash memory. Unfortunately, due to the requirement of the reading speed of NAND flash memory, the LDPC decoder will not obtain precise soft information to achieve high error-correcting capability. In this work, we use a density evolution (DE) algorithm to reveal the decoding threshold of the LDPC decoder affected by the read voltages. We propose the efficient design of read voltages so that the LDPC decoder has the lowest decoding threshold. Therefore, this method can guarantee that the designed read voltages are suitable for a given LDPC code. Moreover, since we found that the designed read voltages are related to the structure of the LDPC code, the joint design of the read voltages and LDPC code is then proposed to achieve the capacity of NAND flash memory. The simulation results demonstrate that our proposed design significantly improves the frame error rate (FER) performance of NAND flash memory.

INDEX TERMS NAND flash memory, read voltages, density evolution, LDPC codes.

I. INTRODUCTION

Nowadays, NAND flash memory become the dominant storage of high-performance computing systems due to its appealing reading/writing speed and storage density. Besides the memory cell scaling, the increasing number of bits per cell, i.e., multi-level cells (MLC) [1], triple-level cells (TLC) [2], and quadruple-level cells (QLC) [3], is the alternative technique to improve the NAND Flash memory density. Unfortunately, the increasing number of bits per cell degrades the reliability of NAND flash memory intensely [4], [5], [6]. Therefore, advanced error correction codes (ECCs) such as the low-density parity-check (LDPC) codes have been widely used to improve the reliability of NAND flash memory [7]. It is well known that the high error-correcting capability of LDPC codes can be achieved when the soft channel information has a very high resolution. Due to the requirement of high reading speed, the resolution of soft channel information will be limited. Since NAND flash memory employs the multiple reads [8] to obtain the soft channel information, the increasing number of reads inevitably increases the reading latency. Therefore, the read voltages must be designed so that the soft channel information adequately describes channel characteristics.

The multiple reads with different read voltages can be viewed as a signal quantizer. The uniform quantizer, in which the read voltages are applied with equally spaced levels, cannot yield a good error performance [7]. The research problem will unavoidably fall into the design of an excellent quantizer. In [9], the maximization of mutual information (MMI) was introduced to design the read voltages of the quantizer. An alternate technique, called entropy-based quantization, was presented in [10]. This method aims to find the balance between the error probability and erasure probability so that the LDPC decoder provides a good bit error rate (BER). The quantizer designed by the entropy-based method outperforms the MMI method. However, the entropy-based method may suffer from the optimization of entropy value through Monte Carlo simulation. Moreover, the number of the read voltages must be an even number. In [11], the channel coding rate (CCR) was used to design the read voltages for MLC NAND flash memory. However, this method does not consider the

The associate editor coordinating the review of this manuscript and approving it for publication was Dongxiao Yu.

behavior of the LDPC decoder. In our early study [12], we proposed the read voltages for MLC NAND flash memory using density evolution (DE). By our proposed design, the read voltages are optimized in such a way that the LDPC decoder has the lowest decoding threshold. Our proposed design is more flexible than the entropy technique [10], in which the number of the read voltages can be an odd or an even number. The theoretical and simulation results demonstrate that our read voltages provide better FER performance than the read voltages optimized by the MMI [9] and Entropy [10]. Moreover, the performance evaluation of the LDPC decoder through Monte Carlo simulation is avoided. Recently, the read voltages optimization through DE was studied extensively in [13]. The authors suggest a two-step optimization technique to design the read voltages, whereby the MMI method was used to design the coarse read voltages and then the DE was applied to generate the precise read voltages. In [14], the read voltages were designed by combining recurrent neural networks (RNN) and DE. The design process needs RNN to find hard decisions. Then, the DE was used to adjust the erasure widths. Similar to the entropy-based method [10], the number of read voltages in the dominant overlapped regions will be an even number.

In this work, we present the comprehensive design of the read voltages through DE. Our design supports any number of read voltages and any channel models of NAND flash memory. Firstly, the multiple sets of read voltages are designed to support every PE cycle or signal-to-noise ratio (SNR). In this design, each set of read voltages is obtained by minimizing the error probability of the LDPC decoder. Secondly, a single set of read voltages is designed for any PE cycle. We propose to use the decoding threshold as the objective function of optimization instead of error probability.

In our read voltages optimization, we found that the optimized read voltages are related to the structure of the LDPC code. Therefore, the read voltages and LDPC code must be jointly designed to achieve the capacity of NAND flash memory. Using the DE algorithm, both the read voltages and LDPC codes can be designed easily. The theoretical and simulation results demonstrate that our joint design can improve the FER performance of NAND flash memory, significantly. Moreover, in practical read operations [8], multiple decoding is used to confine the reading speed. We then introduce the design rule of read voltages, whereby the performance of first and second decoding can be defined through DE.

Since there are several design rules proposed in this work, we will summarize our contributions as follows:

- We propose the comprehensive design of read voltages by using the DE technique (incorporating our conference version with preliminary results [12]). Given the structure of LDPC codes, there are two design rules. For NAND flash memory with multiple sets of read voltages, we propose to design the read voltages by minimizing the error probability. For NAND flash memory with a single set of read voltages, we propose to design the read voltages by maximization of the decoding threshold. The theoretical and simulation results demonstrate that our read voltages provide better FER performance than the read voltages optimized by the MMI [9] and Entropy [10].
- Our design of read voltages can support the various LDPC decoders such as belief propagation, min-sum [15], normalized min-sum [16], and offset min-sum algorithms [17]. The results confirm that our read voltages can improve the NAND flash memory. Particularly, the proposed read voltages for min-sum, normalized min-sum, and offset min-sum decoder offers better FER performance than the read voltages optimized by MMI [9].
- We propose the joint design of read voltages and LDPC codes to achieve the capacity of NAND flash memory. Besides improving the FER performance, our technique can support any number of the read voltages and any channel models of NAND flash memory. The theoretical and simulation results indicate that our joint design provides better FER performance than the separate design. Especially, the design of irregular LDPC code and protograph LDPC code are presented. Our joint design of read voltages and irregular LDPC codes provides better FER performance than the irregular codes designed for existing read voltages (MMI [9] and Entropy [10]). Moreover, our joint design of read voltages and protograph LDPC code shows FER improvements when compared to AR4JA [18] and OARA codes [19]. Note that we will consider the bit-interleaved coded modulation (BICM) system [20]. For the bit-interleaved coded modulation with iterative detection and decoding (BICM-ID) system [21], the proposed design can be applied straightforwardly.
- We introduce the design rule of read voltages for multiple decoding [8], whereby the performance of first and second decoding can be considered intensively. Given the structure of LDPC code, the DE algorithm still helps us to design the read voltages for first and second decoding. We also present the joint design of read voltages and LDPC codes in NAND flash memory with multiple decoding. For example, the read voltages and LDPC code are jointly optimized whereby the decoding threshold of the first decoding or second decoding is at the highest level.

The rest of the paper is organized as follows. Section II covers the channel models for MLC and TLC NAND flash memory. The multiple reads of NAND flash memory are explained in Section III. The error performance estimation functions using density evolution are given in Section IV. Previous designs of read voltages are discussed in Section V. Our designs of read voltages and LDPC codes are presented in Sections VI, VII, and III. The design of read voltages and LDPC codes for multiple decoding is presented in IX. The simulation results and conclusion are presented in Sections X and XI, respectively.

II. CHANNEL MODELS OF NAND FLASH MEMORY

The memory cell of NAND flash memory employs the floating gate transistor. The threshold voltage v_{th} of the memory cell is determined by the amount of charge in the transistor. Each threshold voltage level, which determines whether the transistor is turned on or off, represents a symbol of the data. Before the data are stored in the memory cells, the data in the cells, i.e. the charge of the cell must be eliminated making the lowest threshold voltage. The write voltage is then applied to the memory cell to increase the threshold voltage to a certain voltage level. In this work, we consider two channel models of NAND flash memory such as the Gaussian channel model [9], [22] and the empirical channel model [5], [6], [7], [10].

A. GAUSSIAN CHANNEL MODEL

In [9] and [22], the Gaussian channel model channel has been presented. For MLC NAND flash memory, two bits are stored in a single cell. Thus, they require four distinct threshold voltages: $\mu_{11}, \mu_{10}, \mu_{00}$, and μ_{01}, which correspond to the symbols "11," "10," "00," and "01," respectively. For each threshold, we assume that the distribution is a Gaussian, i.e.,

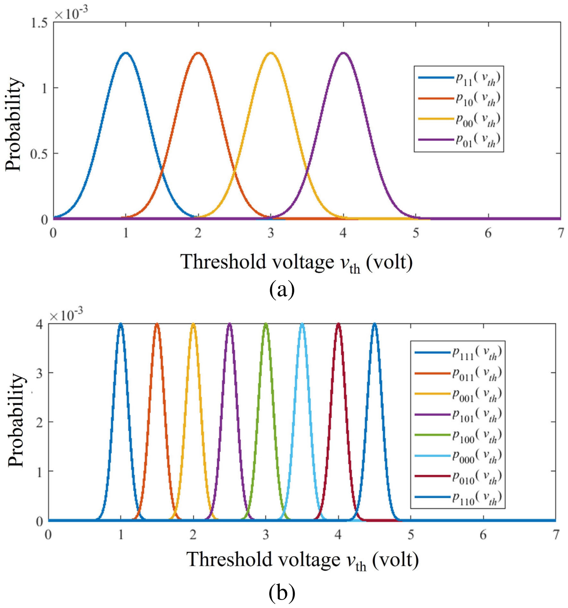
$$\mathrm{p}_{\mathbf{s}}(v_{\text{th}}) = \frac{1}{\sigma_{\mathbf{s}}\sqrt{2\pi}} \exp\left(-\frac{(v_{\text{th}} - \mu_{\mathbf{s}})^2}{2\sigma_{\mathbf{s}}^2}\right) \tag{1}$$

where $\mathrm{p}_{\mathbf{s}}(v_{\text{th}})$ is the distribution function of the threshold voltage, $\sigma_{\mathbf{s}}^2$ and $\mu_{\mathbf{s}}$ are the variance and mean, and $\mathbf{s} \in \{11, 10, 00, 01\}$. This MLC NAND flash memory with the Gaussian model can be considered as 4-pulse amplitude modulation (4-PAM) with additive white Gaussian noise (AWGN). For TLC NAND flash memory with 3 bits per cell, the threshold voltage may be set to one of eight levels: $\mu_{111}, \mu_{011}, \mu_{001}, \mu_{101}, \mu_{100}, \mu_{000}, \mu_{010}$, and μ_{110}, which correspond to the symbols "111," "011," "001," "101," "100," "000," "010," and "110," respectively. The signal-to-noise ratio, SNR, for Gaussian channels can be determined by the ratio of symbol energy, E_s, and noise energy, $N_0 = 2\sigma^2$, i.e., SNR = $10\log_{10}\left(E_s/2R\sigma^2\right)$, where R is a code rate.

Fig. 1 (a) shows the threshold voltage distribution function of the MLC Gaussian channel. The write levels $\mu_{11}, \mu_{10}, \mu_{00}$, and μ_{01}, are set as 1, 2, 3, and 4 volts, respectively. The variances $\sigma_{11}^2, \sigma_{10}^2, \sigma_{00}^2$, and σ_{10}^2 are set to be 0.1 volt. Fig. 1 (b) shows the threshold voltage distribution function of the TLC Gaussian channel. The write levels $\mu_{111}, \mu_{011}, \mu_{001}, \mu_{101}, \mu_{100}, \mu_{000}, \mu_{010}$, and μ_{110}, are set as 1, 1.5, 2, 2.5, 3, 3.5, 4, and 4.5 volts, respectively. The variances $\sigma_{111}^2, \sigma_{011}^2, \sigma_{001}^2, \sigma_{101}^2, \sigma_{100}^2, \sigma_{000}^2, \sigma_{010}^2$, and σ_{110}^2 are set to be 0.01 volt.

B. EMPIRICAL CHANNEL MODEL

In practice, the memory cell composes of several noise sources [5], [6], [7], [10], including Program and Erase Noise (PE), Cell-to-Cell Interference (CCI), Random Telegraph Noise (RTN), and Retention Time (RT) Noise. In [5], [6], [7], and [10], the empirical channel model of the MLC NAND flash model has been presented. Here, we derive the threshold voltages by exploiting the characteristic function. Thus, the threshold voltages can be written in a closed-form expression and are easily visualized with any channel parameter. Given threshold voltages, $V_s \in \{V_{11}, V_{10}, V_{00}, V_{01}\}$, for symbols '11', '10', '00' and '01', respectively. The closed-form expression of MLC NAND flash memory is presented as follows



(a)

(b)

FIGURE 1. (a) The threshold voltages of MLC Gaussian channel model, (b)The threshold voltages of TLC Gaussian channel model.

1) PROGRAM AND ERASE NOISE

Before memory cells are programmed, the charges in the floating-gate layer of all memory cells in the same block are removed [5], [6], [7], [10]. Thus, the memory cell possesses the lowest threshold voltage V_{11}. This erased state represents the symbol "11." In [5], [6], [7], and [10], the distribution function $\mathrm{p}_{\mathrm{E}}(v_{\text{th}})$ is modeled as a Gaussian distribution with mean $V_{11} = 1.2$ volts and standard deviation $\sigma_{\mathrm{E}} = 0.35$ volt. Therefore, the distribution of threshold voltage can be written by

$$\mathrm{p}_{\mathrm{E}}(v_{\text{th}}) = \frac{1}{\sigma_{\mathrm{E}}\sqrt{2\pi}} \exp\left(-\frac{(v_{\text{th}} - V_{11})^2}{2\sigma_{\mathrm{E}}^2}\right). \tag{2}$$

The characteristic function of this normal distribution is given by

$$\varphi_{\mathrm{p}_{\mathrm{E}}}(t) = \exp(itV_{11} - \frac{1}{2}t^2\sigma_{\mathrm{E}}^2). \tag{3}$$

To write a memory cell, the target threshold voltage $V_l \in \{V_{10}, V_{00}, V_{01}\}$ must be defined. The incremental step pulse programming (ISPP) [23] with $\Delta V_{\text{pp}} = 0.3$ volts is applied at the gate terminal of a floating gate transistor. Then, the threshold voltage v_{th} will be verified immediately. If v_{th} is less than the target voltage V_l, the ISPP with ΔV_{pp} will be performed. From ISPP, the threshold voltage becomes the uniform distribution as [23]

$$\mathrm{p}_{\mathrm{ISPP}_l}(v_{\mathrm{th}}) = \begin{cases} \frac{1}{\Delta V_{\mathrm{pp}}}, & \text{if } V_l \leq v_{\mathrm{th}} \leq V_l + \Delta V_{\mathrm{pp}} \\ 0, & \text{else} \end{cases}, \tag{4}$$

and its characteristic function is described by

$$\varphi_{p_{\mathrm{ISPP}_l}}(t) = \frac{\exp(it(V_l + \Delta V_{\mathrm{pp}})) - \exp(itV_l)}{it\Delta V_{\mathrm{pp}}}. \tag{5}$$

2) CELL-TO-CELL INTERFERENCE

There are 2 types of structures such as even/odd bit-line structure [24] and all-bit-line structure [25]. In this paper, the all-bit-line structure is considered only. The memory cells in the word line are influenced by the 2 cells in the even bit-line and 1 cell in the odd bit-line, whose coupling ratios are γ_{xy} and γ_y, respectively. The cell-to-cell interference causes the shifted threshold voltage V_{shifted} [5], [6], [7], [10] to be

$$V_{\mathrm{shifted}} = \sum_{k \in \{\mathrm{xy,y,xy}\}} \Delta V_k \gamma_k, \tag{6}$$

where ΔV_k is the change of threshold voltage of neighbor cell k. From [7], the distribution of cell-to-cell interference is modeled as the Gaussian distribution written by

$$\mathrm{p}_{\mathrm{C2C}}(v_{\mathrm{th}}) = \begin{cases} \frac{0.75}{\sigma_{\mathrm{C2C}}\sqrt{2\pi}} \exp\left(-\frac{(v_{\mathrm{th}}-\mu_{\mathrm{C2C}})^2}{2\sigma^2_{\mathrm{C2C}}}\right), & \text{if } v_{\mathrm{th}} \neq 0 \\ 0.25, & \text{else} \end{cases}, \tag{7}$$

where σ_{C2C} and μ_{C2C} represent a standard deviation and a mean of shifted voltage, respectively. We then define the characteristic function as

$$\varphi_{p_{\mathrm{C2C}}}(t) = 0.75\exp(it\mu_{\mathrm{C2C}} - \frac{1}{2}t^2\sigma^2_{\mathrm{C2C}}) + 0.25. \tag{8}$$

3) RANDOM TELEGRAPH NOISE

The tunnel oxide layer in a memory cell is damaged by the number of programmed and erased (PE) cycles. At the charge trap site, the electrons capture and emission situation cause the fluctuated threshold voltage [26]. The distribution of random telegraph noise is modeled as a symmetric exponential function as [5], [6], and [7]

$$\mathrm{p}_{\mathrm{RTN}}(v_{\mathrm{th}}) = \frac{1}{2\lambda_{\mathrm{RTN}}} \exp\left(-\frac{|v_{\mathrm{th}}|}{\lambda_{\mathrm{RTN}}}\right), \tag{9}$$

where $\lambda_{\mathrm{RTN}} = 0.00025(\mathrm{PE})^{0.5}$. The characteristic function can be defined as

$$\varphi_{p_{\mathrm{RTN}}}(t) = \frac{1}{1 + \lambda^2_{\mathrm{RTN}}t^2}. \tag{10}$$

4) RETENTION NOISE

The increasing number of PE cycles damages the tunnel oxide layer. When the data are stored in a memory cell for a long time, the charge in the floating-gate layer is de-trapped [27] and leads to decreased threshold voltages, i.e., threshold voltages are shifted. The distribution of retention noise p_{RT} is approximately modeled as a Gaussian distribution [23] given by

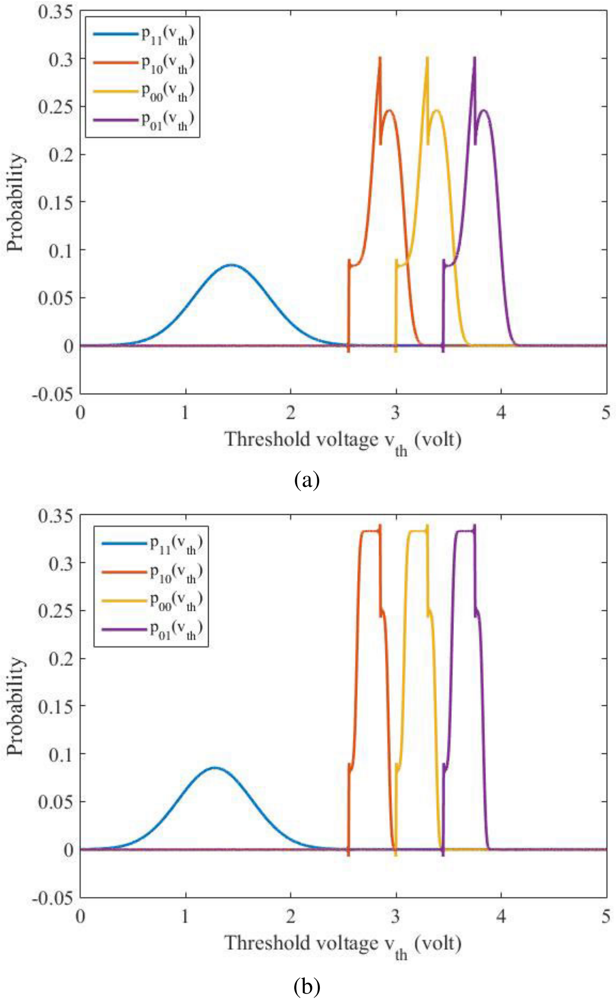


FIGURE 2. (a) The empirical channel model of MLC NAND flash memory with sf = 1.5, (b) The empirical channel model of MLC NAND flash memory with sf = 0.5.

$$\mathrm{p}_{\mathrm{RT}_s}(v_{\mathrm{th}}) = \frac{1}{\sigma_{\mathrm{RT}_s}\sqrt{2\pi}}\exp\left(-\frac{(v_{\mathrm{th}} - \mu_{\mathrm{RT}_s})^2}{2\sigma^2_{\mathrm{RT}_s}}\right), \tag{11}$$

where $\sigma_{\mathrm{RT}_s} \in \{\sigma_{\mathrm{RT}_{11}}, \sigma_{\mathrm{RT}_{10}}, \sigma_{\mathrm{RT}_{00}}, \sigma_{\mathrm{RT}_{01}}\}$ and $\mu_{\mathrm{RT}_s} \in \{\mu_{\mathrm{RT}_{11}}, \mu_{\mathrm{RT}_{10}}, \mu_{\mathrm{RT}_{00}}, \mu_{\mathrm{RT}_{01}}\}$ denote a standard deviation and a mean of retention noise, which can be computed from

$$\mu_{\mathrm{RT}_s} = (\tilde{V}_s - x_0)(A_t\mathrm{PE}^{\alpha_i} + B_t\mathrm{PE}^{\alpha_o})\log(1 + T), \tag{12}$$

$$\sigma_{\mathrm{RT}_s} = 0.4|\mu_{\mathrm{RT}_s}|, \tag{13}$$

where $\tilde{V}_s \in \{\tilde{V}_{11}, \tilde{V}_{10}, \tilde{V}_{00}, \tilde{V}_{01}\}$ and $\tilde{V}_{11} = V_{11} + V_{\mathrm{shifted}}, \tilde{V}_{10} = V_{10} + V_{\mathrm{shifted}}, \tilde{V}_{00} = V_{00} + V_{\mathrm{shifted}}, \tilde{V}_{01} = V_{01} + V_{\mathrm{shifted}}$. In this work, we set $A_t = 0.000055$, $B_t = 0.000235$, $\alpha_i = 0.62$, $\alpha_o = 0.32$, and $x_0 = 1.2$. The T denotes the retention time. Thus, the characteristic function is given by

$$\varphi_{p_{\mathrm{RT}_s}}(t) = \exp(it\mu_{\mathrm{RT}_s} - \frac{1}{2}t^2\sigma^2_{\mathrm{RT}_s}). \tag{14}$$

5) THRESHOLD VOLTAGES OF MLC NAND FLASH MEMORY

The convolution operation of $\mathrm{p}_{\mathrm{PE}}, \mathrm{p}_{\mathrm{ISPP}_l}, \mathrm{p}_{\mathrm{C2C}}, \mathrm{p}_{p_{\mathrm{RTN}}}$ and $\mathrm{p}_{\mathrm{RT}_s}$ can generate the threshold voltages of MLC NAND flash memory. The threshold voltage of the erased cell, which is corrupted by cell-to-cell interference, random telegraph noise, and data retention noise can be written as

$$\mathrm{p}_{11}(v_{\mathrm{th}}) = \mathrm{p}_{\mathrm{E}}(v_{\mathrm{th}}) * \mathrm{p}_{\mathrm{C2C}}(v_{\mathrm{th}}) * \mathrm{p}_{\mathrm{RTN}}(v_{\mathrm{th}}) * \mathrm{p}_{\mathrm{RT}_{11}}(v_{\mathrm{th}}), \tag{15}$$

where $*$ denotes the convolution operation. For the programmed cell, the threshold voltage distribution $\mathrm{p}_l \in \{\mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}\}$ can be generated as follow

$$\mathrm{p}_l(v_{\mathrm{th}}) = \mathrm{p}_{\mathrm{ISPP}_l}(v_{\mathrm{th}}) * \mathrm{p}_{\mathrm{C2C}}(v_{\mathrm{th}}) * \mathrm{p}_{\mathrm{RTN}}(v_{\mathrm{th}}) * \mathrm{p}_{\mathrm{RT}_l}(v_{\mathrm{th}}). \tag{16}$$

To facilitate the optimization of the read voltages, we exploit the characteristic function to obtain the compact channel model. By using a characteristic function, the convolution operation in (15) and (16) will become the multiplication operation. The closed-form equations of threshold voltages corrupted by several noises are written by

$$\begin{aligned}
&\mathrm{p}_{11}(v_{\mathrm{th}})\\
&= \frac{0.25}{2\pi}\left(\int_{-\infty}^{\infty} \frac{3\exp\left(it(\mu_{\mathrm{ES1}} - v_{\mathrm{th}}) - (\sigma_{\mathrm{ES1}}^2 t^2)/2\right)}{1+\lambda_{\mathrm{RTN}}^2 t^2}dt\right.\\
&\quad \left. + \int_{-\infty}^{\infty} \frac{\exp\left(it(\mu_{\mathrm{ES2}} - v_{\mathrm{th}}) - (\sigma_{\mathrm{ES2}}^2 t^2)/2\right)}{1+\lambda_{\mathrm{RTN}}^2 t^2}dt\right), \qquad (17)\\
&\mathrm{p}_l(v_{\mathrm{th}})\\
&= \frac{0.25}{2\pi}\left(\int_{-\infty}^{\infty} \frac{3\exp\left(it(\mu_{\mathrm{PS1}_l} - v_{\mathrm{th}}) - (\sigma_{\mathrm{PS1}_l}^2 t^2)/2\right)}{\Delta V_{\mathrm{pp}} it(1+\lambda_{\mathrm{RTN}}^2 t^2)}dt\right.\\
&\quad - \int_{-\infty}^{\infty} \frac{3\exp\left(it(\mu_{\mathrm{PS2}_l} - v_{\mathrm{th}}) - (\sigma_{\mathrm{PS1}_l}^2 t^2)/2\right)}{\Delta V_{\mathrm{pp}} it(1+\lambda_{\mathrm{RTN}}^2 t^2)}dt\\
&\quad + \int_{-\infty}^{\infty} \frac{\exp\left(it(\mu_{\mathrm{PS3}_l} - v_{\mathrm{th}}) - (\sigma_{\mathrm{PS2}_l}^2 t^2)/2\right)}{\Delta V_{\mathrm{pp}} it(1+\lambda_{\mathrm{RTN}}^2 t^2)}dt\\
&\quad \left. - \int_{-\infty}^{\infty} \frac{\exp\left(it(\mu_{\mathrm{PS4}_l} - v_{\mathrm{th}}) - (\sigma_{\mathrm{PS2}_l}^2 t^2)/2\right)}{\Delta V_{\mathrm{pp}} it(1+\lambda_{\mathrm{RTN}}^2 t^2)}dt\right), \qquad (18)
\end{aligned}$$

where $\mu_{\mathrm{ES1}} = V_{11} + \mu_{\mathrm{C2C}} + \mu_{\mathrm{RT}_{11}}$, $\mu_{\mathrm{ES2}} = V_{11} + \mu_{\mathrm{RT}_{11}}$, $\sigma_{\mathrm{ES1}}^2 = \sigma_{\mathrm{E}}^2 + \sigma_{\mathrm{C2C}}^2 + \sigma_{\mathrm{RT}_{11}}^2$, $\sigma_{\mathrm{ES2}}^2 = \sigma_{\mathrm{E}}^2 + \sigma_{\mathrm{RT}_{11}}^2$, $\mu_{\mathrm{PS1}_l} = V_l + \Delta V_{\mathrm{pp}} + \mu_{\mathrm{C2C}} + \mu_{\mathrm{RT}_l}$, $\mu_{\mathrm{PS2}_l} = V_l + \mu_{\mathrm{C2C}} + \mu_{\mathrm{RT}_l}$, $\mu_{\mathrm{PS3}_l} = V_l + \Delta V_{\mathrm{pp}} + \mu_{\mathrm{RT}_l}$, $\mu_{\mathrm{PS4}_l} = V_l + \mu_{\mathrm{RT}_l}$, $\sigma_{\mathrm{PS1}_l}^2 = \sigma_{\mathrm{C2C}}^2 + \sigma_{\mathrm{RT}_l}^2$, $\sigma_{\mathrm{PS2}_l}^2 = \sigma_{\mathrm{RT}_l}^2$, and $l \in \{10, 00, 11\}$.

Fig. 2 (a) and (b) show the threshold voltages generated from (17) and (18), where PE $= 0$ cycles, $V_{01} = 2.55$ volts, $V_{00} = 3$ volts, $V_{01} = 3.45$ volts, $T = 1$ year and $\mu_y = 0.08$sf. In Fig. 2 (a), we set strength factor (sf) to 1.5, $\sigma_{\mathrm{C2C}}^2 = 0.0038$, $\mu_{\mathrm{C2C}} = 0.2340$ volts and $\tilde{V}_{11} = 1.3747$ volts. In Fig. 2 (b), we use sf $= 0.5$, the $\sigma_{\mathrm{C2C}}^2 = 4.2276 \times 10^{-4}$, $\mu_{\mathrm{C2C}} = 0.0780$ volts and $\tilde{V}_{11} = 1.2583$ volts. Note that we will use sf $= 0.5$ and $T = 1$ in all simulations of this paper.

III. MULTIPLE READS OF NAND FLASH MEMORY

Multiple reads are required to obtain the soft channel information from a memory cell [7], for example, the MLC NAND flash memory requires read voltages of at least 6 different voltages and the TLC NAND flash memory requires read voltages of at least 14 distinct voltages. Let $\mathbf{r} = [r_1, r_2, r_3, r_4, r_5, r_6]$ be the read voltage set of MLC NAND flash memory. The read voltage r_1 is an initial read voltage applied to a memory cell. Then, the current output of the memory cell is probed by the sense amplifier. If the sense amplifier cannot probe any current, the read voltage r_2 will be applied. This process continues until the read voltage r_6 is applied or the current output is probed. Then, the threshold voltage of the memory cell will be estimated. The read voltages applied to the memory cell can be viewed as the quantizer where the threshold voltage is quantized at certain levels.

Since the MLC NAND flash memory stores 2 bits/cell consisting of the most significant bit (MSB) and the least significant bit (LSB), we will consider the discrete memoryless channel (DMC) of MSB and LSB separately. Considering the soft information required from the LDPC decoder, the read voltages $\mathbf{r} = [r_1, r_2, r_3, r_4, r_5, r_6]$ will divide the threshold voltage into 7 regions $\Omega = \{\Omega_1, \Omega_2, \Omega_3, \Omega_4, \Omega_5, \Omega_6, \Omega_7\}$, where $\Omega_1 \in (-\infty, r_1]$, $\Omega_2 \in (r_1, r_2]$, $\Omega_3 \in (r_2, r_3]$, $\Omega_4 \in (r_3, r_4]$, $\Omega_5 \in (r_4, r_5]$, $\Omega_6 \in (r_5, r_6]$ and $\Omega_7 \in (r_6, \infty)$. Let x_s be an input symbol. Hence, the conditional probability $\mathrm{p}_j(\Omega_j|x_s)$ is calculated by

$$\mathrm{p}_j(\Omega_j|x_s) = \int_{\Omega_j} \mathrm{p}_s(v_{\mathrm{th}})dv_{\mathrm{th}}, \tag{19}$$

where $s \in \{11, 10, 00, 01\}$ and $j \in \{1, 2, 3, 4, 5, 6, 7\}$. Let L_{M} and L_{L} be the log-likelihood ratio (LLR) of MSB and LSB with distributions $\mathrm{p}_{L_{\mathrm{M}}}$ and $\mathrm{p}_{L_{\mathrm{L}}}$, respectively. Thus,

$$L_{\mathrm{M}_j} = \frac{\sum_{s\in\{11,10\}} \mathrm{p}_j(\Omega_j|x_s)}{\sum_{s\in\{00,01\}} \mathrm{p}_j(\Omega_j|x_s)} \text{ and} \tag{20}$$

$$L_{\mathrm{L}_j} = \frac{\sum_{s\in\{11,01\}} \mathrm{p}_j(\Omega_j|x_s)}{\sum_{s\in\{10,00\}} \mathrm{p}_j(\Omega_j|x_s)}. \tag{21}$$

IV. THEORETICAL ANALYSIS OF LDPC DECODER IN NAND FLASH MEMORY

In this section, we resurrect the theoretical analysis of LDPC codes, called density evolution (DE) [28]. Note that the LDPC decoder generally uses the sum-product algorithm. In this work, we will show the theoretical analysis of various LDPC decoders such as belief propagation, min-sum [15], normalized min-sum [16], and offset min-sum [17]. The theoretical analysis through DE will be the essential component of our proposed optimization, for example, the DE is used to analyze the read voltages of NAND flash memory. Here, we will derive the DE for MLC NAND flash memory only since the DE can be extended to the TCL and QLC NAND flash memories, straightforwardly. The behaviors of optimal read voltages for LDPC codes under sum-product and min-sum algorithms are investigated in this section and their results confirm that the DE can be used to find the suitable read voltages for the LDPC decoder.

A. DENSITY EVOLUTION (DE)

The DE [28] is an effective tool to analyze the error-correcting capability of LDPC codes. The concept of the DE is to track the LLR distribution of the check node and variable node during the iterative decoding. Finally, the error probability is estimated from the LLR distribution in the final iterations at the variable node. In this work, the error-correcting capabilities of regular LDPC codes, irregular LDPC codes, and protograph LDPC codes are presented. Since their parity-check matrixes are different, we will define the DE computations separately. The details of DE are as follows.

1) DENSITY EVOLUTION (DE) FOR REGULAR LDPC CODES

Regular LDPC codes with constant row weight and column weight represent the parity check matrix with pairs of row weight and column weight $(d_{\mathrm{v}}, d_{\mathrm{c}})$. Let L_{CN} be an outgoing LLR of check node with the distribution $\mathrm{p}^{(l)}_{L_{\mathrm{CN}}}$ at the l-th iteration and L_{VN} be an outgoing LLR of variable node with the distribution $\mathrm{p}^{(l)}_{L_{\mathrm{VN}}}$ at the l-th iteration. The L_{CH} is defined as a channel output with the distribution $\mathrm{p}_{L_{\mathrm{CH}}}$.

At the variable node process, the distribution of the outgoing LLR is given by

$$\mathrm{p}^{(l)}_{L_{\mathrm{VN}}}(L) = \mathrm{p}_{\mathrm{CH}}(L) * \left(\mathrm{p}^{(l-1)}_{L_{\mathrm{CN}}}(L)\right)^{*(d_{\mathrm{v}}-1)}, \tag{22}$$

where $\mathrm{p}_{\mathrm{Z}}(z) = \mathrm{p}_{\mathrm{X}}(x) * \mathrm{p}_{\mathrm{Y}}(y) = \int_{-\infty}^{\infty} \mathrm{p}_{\mathrm{X}}(z-y)\mathrm{p}_{\mathrm{Y}}(y)dy$ and $\mathrm{p}_{\mathrm{X}}(x) * \mathrm{p}_{\mathrm{X}}(x) = \mathrm{p}_{\mathrm{X}}(x)^{*2}$.

At the check node process, the distribution of outgoing LLR is obtained by

$$\mathrm{p}^{(l)}_{L_{\mathrm{CN}}}(L) = \mathrm{p}^{(l)}_{L_{\mathrm{VN}}}(L)^{\otimes(d_{\mathrm{c}}-1)}, \tag{23}$$

where

$$\begin{aligned} \mathrm{p}_{\mathrm{Z}}(z) &= \mathrm{p}_{\mathrm{X}}(x) \otimes \mathrm{p}_{\mathrm{Y}}(y) \\ &= \int_{z=\mathrm{g}(x,y)} \mathrm{p}_{\mathrm{X}}(x)\mathrm{p}_{\mathrm{Y}}(y)dxdy, \end{aligned} \tag{24}$$

and $\mathrm{p}_{\mathrm{X}}(x) \otimes \mathrm{p}_{\mathrm{X}}(x) = \mathrm{p}_{\mathrm{X}}(x)^{\otimes 2}$. This distribution function can be calculated differently due to the different decoding algorithms. The function $\mathrm{g}(x, y)$ for belief propagation (BP), min-sum (MS), normalized min-sum (NMS), and offset min-sum (OMS) decoding can be calculated from

$$\mathrm{g}_{\mathrm{BP}}(x, y) = -2\tanh^{-1}(\tanh(x/2)\tanh(y/2)). \tag{25}$$

$$\mathrm{g}_{\mathrm{MS}}(x, y) = \mathrm{sign}(xy)\min(|x|, |y|), \tag{26}$$

$$\mathrm{g}_{\mathrm{NMS}}(x, y) = \mathrm{sign}(xy)\min(|x|, |y|)\alpha, \tag{27}$$

and

$$\mathrm{g}_{\mathrm{OMS}}(x, y) = \mathrm{sign}(xy)\max(\min(|x|, |y|) - \beta, 0), \tag{28}$$

where $0 < \alpha < 1$ and $0 < \beta < 1$. Note that the complexity of the min-sum decoder is exist in [29].

To analyze the theoretical performance of the LDPC decoder, all zero codewords with infinity length are assumed. The density evolution will produce the LLR distribution of the check node and variable node until the maximum iteration number $l_{\max}$ is reached. At the final iteration, the distribution of the output of the LDPC decoder will be estimated from

$$\mathrm{p}^{(l_{\max})}_{L_{\mathrm{VN}}}(L) = \mathrm{p}_{\mathrm{CH}}(L) * \left(\mathrm{p}^{(l_{\max}-1)}_{L_{\mathrm{CN}}}(L)\right)^{*(d_{\mathrm{v}})}, \tag{29}$$

and the error probability is approximated by

$$\mathrm{P}_{\mathrm{err}} = \int_{-\infty}^{0} \mathrm{p}^{(l_{\max})}_{L_{\mathrm{VN}}}(L)dL. \tag{30}$$

2) DENSITY EVOLUTION (DE) FOR IRREGULAR LDPC CODES

For irregular LDPC codes, each row has different weights and each column has different weights, the parity check matrix is then represented by degree distribution pairs (λ, ρ). The degree distribution pairs can be represented by polynomial, that is

$$\lambda(X) = \sum_{i=2}^{d_{\mathrm{vmax}}} \lambda_i X^{i-1}, \tag{31}$$

and

$$\rho(X) = \sum_{i=2}^{d_{\mathrm{cmax}}} \rho_i X^{i-1}, \tag{32}$$

where d_{vmax} and d_{cmax} is a maximum column weight and row weight, $\sum_{i=2}^{d_{\mathrm{vmax}}} \lambda_i = 1$, $\sum_{i=2}^{d_{\mathrm{cmax}}} \rho_i = 1$, $\lambda_2, \lambda_3, \ldots, \lambda_{d_{\mathrm{vmax}}} \geq 0$ and $\rho_2, \rho_3, \ldots, \rho_{d_{\mathrm{cmax}}} \geq 0$.

Since the row weight and column weight of an irregular LDPC code are not constant, calculations at variable nodes and check nodes need to be slightly modified from regular LDPC code. The variable node and check node calculation can be rewritten by

$$\mathrm{p}^{(l)}_{L_{\mathrm{VN}}}(L) = \mathrm{p}_{\mathrm{CH}}(L) * \sum_{i=2}^{d_{\mathrm{vmax}}} \left(\lambda_i \mathrm{p}^{(l-1)}_{L_{\mathrm{CN}}}(L)\right)^{*(i-1)}, \tag{33}$$

and

$$\mathrm{p}^{(l)}_{L_{\mathrm{CN}}}(L) = \sum_{i=2}^{d_{\mathrm{cmax}}} \rho_i \mathrm{p}^{(l)}_{L_{\mathrm{VN}}}(L)^{\otimes(i-1)}, \tag{34}$$

3) DENSITY EVOLUTION (DE) FOR PROTOGRAPH LDPC CODES

A protograph LDPC code can be represented by a base graph or protomatrix $\mathbf{B}$. The base graph consists of a set of variable nodes $\mathcal{V}$, a set of check nodes $\mathcal{C}$, and a set of edges $\mathcal{E}$. The base graph can allow degree-1 variable nodes, punctured variable nodes, and parallel edges. Let (i, j, k) be the index of each edge where $i \in \{1, 2, \ldots, M_p\}$ is an index of the check node in the base graph, $j \in \{1, 2, \ldots, N_p\}$ is an index of the variable node in the base graph, $k \in \{1, 2, \ldots, \mathrm{b}(i, j)\}$ is an index of the parallel edge in the base graph, M_p and N_p is a number of check node and variable node in the base graph and $\mathrm{b}(i, j) \in \mathbf{B}$

is a number of the parallel edge. We define $L_{\mathrm{VN}}(i, j, k)$ and $L_{\mathrm{CN}}(i, j, k)$ as an outgoing LLR from j-th variable node to i-th check node via k-th parallel edge and an outgoing LLR from i-th check node to j-th variable node via k-th parallel edge. Let $\mathrm{p}_{\mathrm{VN}}(L(i, j, k))$ and $\mathrm{p}_{\mathrm{CN}}(L(i, j, k))$ be the distribution function of $L_{\mathrm{VN}}(i, j, k)$ and $L_{\mathrm{CN}}(i, j, k)$.

At the check node process, the outgoing LLR distribution is computed by

$$\mathrm{p}^{(l)}_{L_{\mathrm{CN}}}(L(i,j,k)) = \Theta_{\{i,j',k' \in \mathcal{M}(i)\setminus j,k\}} \mathrm{p}^{(l)}_{L_{\mathrm{VN}}}(L(i,j',k')), \tag{35}$$

where $\Theta_{\{i \in \{x1,x2,x3\}\}}\mathrm{p}(i) = \mathrm{p}(x1) \otimes \mathrm{p}(x2) \otimes \mathrm{p}(x3)$ and $\mathcal{M}(i)\setminus j, k$ is the set of all variable nodes and parallel edges connected to i-th check node except j-th variable node and k-th parallel edge.

At the variable node process, the outgoing LLR distribution is obtained by

$$\mathrm{p}^{(l)}_{L_{\mathrm{VN}}}(L(i,j,k)) = \left(\Xi_{\{i',j,k' \in \mathcal{N}(j)\setminus i,k\}} \mathrm{p}^{(l-1)}_{L_{\mathrm{CN}}}(L(i',j,k'))\right) * \mathrm{p}_{\mathrm{CH}}(L(j)), \tag{36}$$

where $\Xi_{\{i \in \{x1,x2,x3\}\}}\mathrm{p}(i) = \mathrm{p}(x1) * \mathrm{p}(x2) * \mathrm{p}(x3)$ and $\mathcal{N}(j)\setminus i, k$ is the set of all variable nodes and parallel edges connected to j-th variable node except i-th check node and k-th parallel edge. The final distribution is given by

$$\mathrm{p}^{(l_{\max})}_{L_{\mathrm{VN}}}(L(j)) = \left(\Xi_{\{i,j,k \in \mathcal{N}(j)\}} \mathrm{p}^{(l_{\max}-1)}_{L_{\mathrm{CN}}}(L(i,j,k))\right) * \mathrm{p}_{\mathrm{CH}}(L(j)), \tag{37}$$

where $\mathcal{N}(j)$ is the set of all variable nodes and parallel edges connected to j-th variable node. Then, the error probability $\mathrm{P}_{\mathrm{err}}$ is calculated by

$$\mathrm{P}_{\mathrm{err}} = \frac{1}{N_p} \sum_{j=1}^{N_p} \int_{-\infty}^{0} \mathrm{p}^{(l_{\max})}_{L_{\mathrm{VN}}}(L(j))dL. \tag{38}$$

B. DEFINITION OF ERROR PROBABILITY ESTIMATION FUNCTION AND DECODING THRESHOLD EXPLORING FUNCTION

Algorithm 1 Error Probability Estimation Function $\mathrm{P}_{\mathrm{avg}}(\mathbf{x})$ for Regular LDPC Codes

```
Input : $\mathbf{r}$, $d_v$, $d_c$, $l_{\max}$, $p_{11}(v_{th})$, $p_{10}(v_{th})$, $p_{00}(v_{th})$ and $p_{01}(v_{th})$
Output : $P_{avg}$
Compute $p_{L_M}$ and $p_{L_L}$ discretized by $\mathbf{r}$.
for $p_{CH} = p_{L_M}$ and $p_{L_L}$.
    for $l = 1, 2, 3 \dots, l_{\max} - 1$
        Calculate $p^{(l)}_{L_{VN}}$, see (22)
        Calculate $p^{(l)}_{L_{CN}}$, see (23)
    end for
    Calculate $p^{(l_{max})}_{L_{VN}}$, see (29)
    Calculate $P_{err}$, see (30)
end for
Calculate $P_{avg}$, see (39)
```

Let $\mathrm{P}_{\mathrm{avg}}(\mathbf{x})$ be an error probability estimation function and PE*($\mathbf{x}$) be a decoding threshold exploring function. The vector $\mathbf{x}$ consists of the distribution of threshold voltages, the read voltage $\mathbf{r}$, the variable node degree d_{v}, the check node degree d_{c}, and the maximum number of iteration $l_{\max}$. The details of $\mathrm{P}_{\mathrm{avg}}(\mathbf{x})$ and PE*($\mathbf{x}$) function are described as follow.

Algorithm 2 Error Probability Estimation Function $\mathrm{P}_{\mathrm{avg}}(\mathbf{x})$ for Irregular LDPC Codes

```
Input : $\mathbf{r}$, $\lambda$, $d_c$, $l_{\max}$, $p_{11}(v_{th})$, $p_{10}(v_{th})$, $p_{00}(v_{th})$ and $p_{01}(v_{th})$
Output : $P_{avg}$
Compute $p_{L_M}$ and $p_{L_L}$ discretized by $\mathbf{r}$.
for $p_{CH} = p_{L_M}$ and $p_{L_L}$.
    for $l = 1, 2, 3 \dots, l_{\max} - 1$
        Calculate $p^{(l)}_{L_{VN}}$, see (33)
        Calculate $p^{(l)}_{L_{CN}}$, see (34)
    end for
    Calculate $p^{(l_{max})}_{L_{VN}}$, see (29)
    Calculate $P_{err}$, see (3)
end for
Calculate $P_{avg}$, see (39)
```

Algorithm 3 Error Probability Estimation Function $\mathrm{P}_{\mathrm{avg}}(\mathbf{x})$ for Protograph LDPC Codes

```
Input : $\mathbf{r}$, $\mathbf{B}$, $l_{\max}$, $p_{11}(v_{th})$, $p_{10}(v_{th})$, $p_{00}(v_{th})$ and $p_{01}(v_{th})$
Output : $P_{avg}$
Compute $p_{L_M}$ and $p_{L_L}$ discretized by $\mathbf{r}$.
for $p_{CH} = p_{L_M}$ and $p_{L_L}$.
    for $l = 1, 2, 3 \dots, l_{\max} - 1$
        Calculate $p^{(l)}_{L_{VN}}$, see (35)
        Calculate $p^{(l)}_{L_{CN}}$, see (36)
    end for
    Calculate $p^{(l_{max})}_{L_{VN}}$, see (37)
    Calculate $P_{err}$, see (38)
end for
Calculate $P_{avg}$, see (39)
```

- The error probability estimation function $\mathrm{P}_{\mathrm{avg}}(\mathbf{x})$: The read voltage $\mathbf{r}$ is applied to a memory cell of NAND flash memory. Then, the DE algorithm will be performed to obtain the error probabilities of MSB and LSB. We define $\mathrm{P}_{\mathrm{avg}}$ as an average of error probabilities,

Algorithm 4 Decoding Threshold Exploring Function PE*($\mathbf{x}$)

```
Input : $\mathbf{r}$, $d_v$ or $\lambda$, $\mathbf{B}$, $d_c$, $l_{\max}$, $p_{11}(v_{th})$, $p_{10}(v_{th})$, $p_{00}(v_{th})$ and $p_{01}(v_{th})$
Output : PE*
Set PE and $P_{avg}$ as 0.
while $P_{avg} < 10^{-6}$
    Calculate $P_{avg}$, see Algorithm 1-3.
    PE* = PE
    PE = PE + 1
end while
```

i.e.,

$$\mathrm{P_{avg}} = \frac{1}{2}(\mathrm{P_{err}^{MSB}} + \mathrm{P_{err}^{LSB}}). \tag{39}$$

The procedures of error probability estimation function $\mathrm{P_{avg}}(\mathbf{x})$ of regular LDPC codes, irregular LDPC codes, and protograph LDPC codes are shown in Algorithm 1 - 3.

- The decoding threshold exploring function $\mathrm{PE}^*(\mathbf{x})$: First, the PE cycle of NAND flash memory is set to be 0. The $\mathrm{PE}^*(\mathbf{x})$ will request the $\mathrm{P_{avg}}(\mathbf{x})$ to compute the $\mathrm{P_{avg}}$. If the $\mathrm{P_{avg}}$ is less than 10^{-6}, the PE cycle is increased. Then the $\mathrm{PE}^*(\mathbf{x})$ will recall the $\mathrm{P_{avg}}(\mathbf{x})$ to compute the $\mathrm{P_{avg}}$. The stopping criterion is that the $\mathrm{P_{avg}}$ is more than or equal to 10^{-6}. The last PE cycle will become the decoding threshold, which is the output of $\mathrm{PE}^*(\mathbf{x})$. The decoding threshold exploring function $\mathrm{PE}^*(\mathbf{x})$ is illustrated in Algorithm 4. Note that the decoding threshold for the Gaussian channel model is the signal-to-noise ratio (SNR).

V. THEORETICAL RESULTS OF LDPC DECODER IN NAND FLASH MEMORY

To investigate the behavior of read voltages, we will consider a single-level cell (SLC) NAND flash memory with a Gaussian channel model as shown in Fig. 3, where mean $\mu = 0$ and variance $\sigma^2 = 0.1476$, threshold voltage $v_{\mathrm{th}} \in \{-1, 1\}$. The read voltages r_0 and r_1 must be applied to obtain the soft information. The distance between the r_0 (r_1) and the optimal read voltage for the hard decision is defined as the erasure width in left area r_{hlw} (the erasure width in right area r_{hrw}), respectively. Since the channel model is symmetric, the r_{hlw} and r_{hrw} can be optimized by setting $r_{\mathrm{hlw}} = r_{\mathrm{hrw}}$. We will use the notation r_{hw} instead of r_{hlw} and r_{hrw}. Figs. 4 and 5 show the empirical mutual information of LLR before and after the BP and MS decoding at any r_{hw}. Note that the empirical mutual information can be calculated by using $I(L; X) = 1 - \mathrm{E}\{\log_2(1 + \exp(-L))\}$ as suggested in [30] where L is an LLR and $\mathrm{E}\{\cdot\}$ is an expectation function. We use LDPC code with $d_{\mathrm{v}} = 3$ and $d_{\mathrm{c}} = 30$, whose parity check matrix is constructed by PEG algorithm [31]. The maximum number of LDPC decoding is set to be 10 iterations. We found that the maximum MIs before the BP and MS decoding are given when the $r_{\mathrm{hw}} = 0.158$ and 0.092 volts, respectively. While the maximum MIs after the BP and MS decoding are located at the $r_{\mathrm{hw}} = 0.247$ and 0.145 volts.

The error probability curves for BP decoding of the (3, 6) and (3, 30) LDPC codes are shown in Fig. 6. For the MS decoding, the error probability curves are shown in Fig. 7. Each curve represents the $\mathrm{P_{avg}}$ estimated from Algorithm 1. The square points represent the error probabilities at the read voltages optimized by MMI [9]. These results confirm that the read voltages optimization through MMI does not provide the lowest error probability. Consider the error probability at SNR = 5.3 dB in Fig. 6 (b). This theoretical error probability corresponds to the empirical mutual information after the BP decoder in Fig. 4. For example, in Fig. 4, the simulation results indicate that the maximum mutual information after BP decoding is located at $r_{\mathrm{hw}} = 0.247$ volts. This value can be found by minimizing the $\mathrm{P_{avg}}(\mathbf{x})$ in Fig. 6 (b). Thus, minimizing $\mathrm{P_{avg}}(\mathbf{x})$ can be the objective function of read voltages optimization when the structure of LDPC codes is given. This congruence also appears in the MS decoding i.g. the maximum mutual information in Fig. 5 corresponds to the error probability curve in Fig. 7 (b). Moreover, at the same code rate, we found that the lowest error probabilities of BP and MS decoding are different.

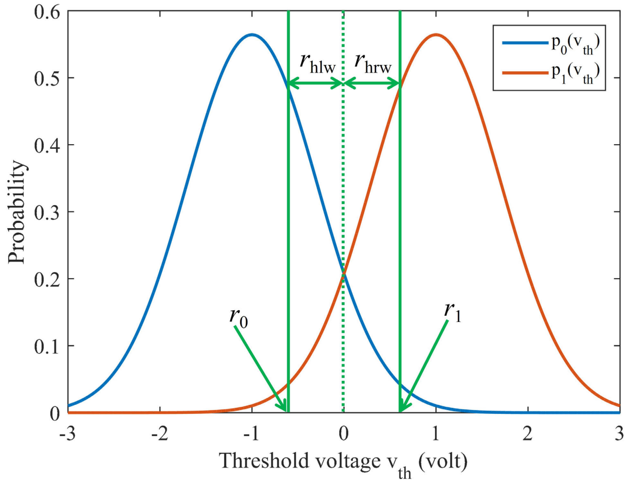


FIGURE 3. The threshold voltages of SLC NAND flash memory with Gaussian channel model.

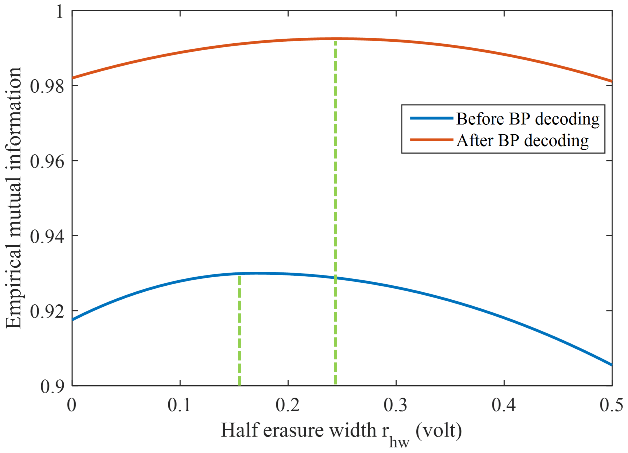


FIGURE 4. The empirical mutual information before and after the (3, 30) LDPC code with BP decoder.

VI. DESIGN OF READ VOLTAGES VIA DENSITY EVOLUTION

In this section, we will introduce the read voltage optimization. As the results in the previous section, the read voltages must be optimized in such a way that the LDPC decoder provides the lowest error probability at any PE cycle. We will use evolutionary programming to find the read voltages that

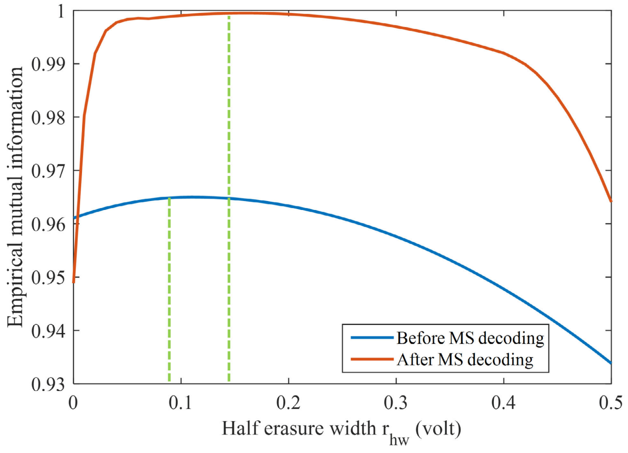
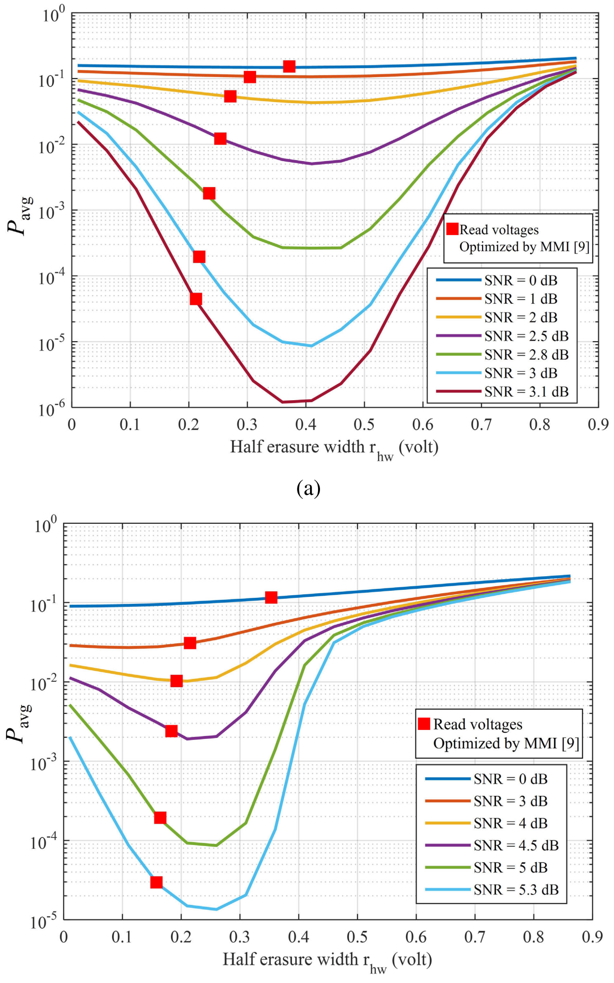


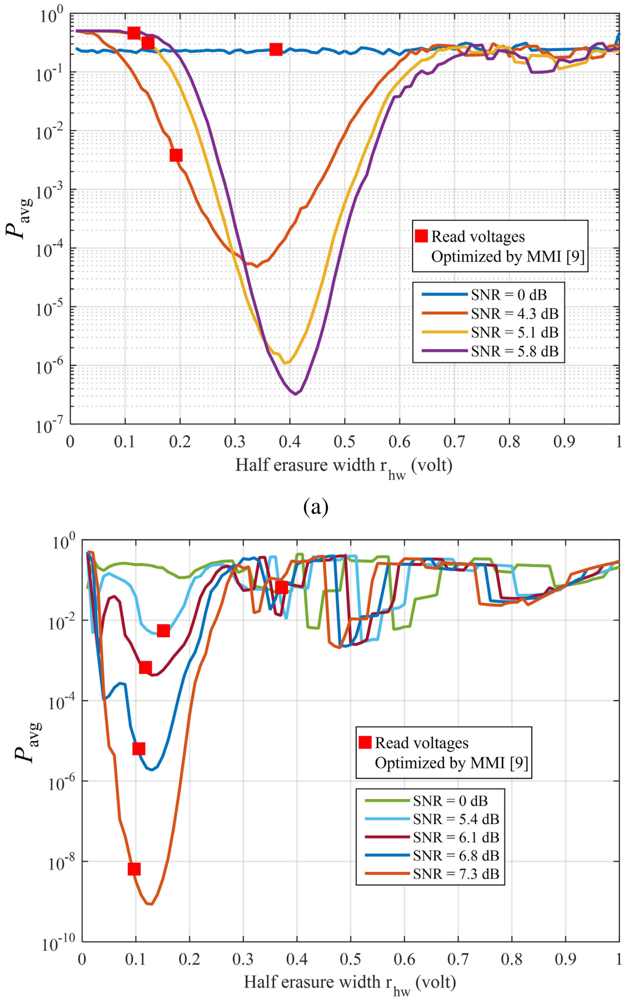
FIGURE 5. The empirical mutual information before and after the (3, 30) LDPC code with MS decoder.



(a)

(b)

FIGURE 6. (a) The error probabilities of (3,6) LDPC codes with BP decoder, (b) The error probabilities of (3,30) LDPC codes with BP decoder.



(a)

(b)

FIGURE 7. (a) The error probabilities of (3,6) LDPC codes with MS decoder, (b) The error probabilities of (3,30) LDPC codes with MS decoder.

the BP, MS, NMS, and OMS decoders have the lowest error probabilities. Note that the error probabilities of BP, MS, NMS, and OMS decoder are estimated from Algorithm 1. We begin to explain the definition and notation of evolutionary programming. Then, the proposed read voltages optimization will be served.

A. EVOLUTIONARY PROGRAMMING

In this work, we will use the differential evolution [32] to confine the feasible solutions. Differential evolution is an evolutionary algorithm for solving optimization problems. Differential evolution possesses 3 important processes such as mutation process, crossover process, and selection process. The details of differential evolution are as follows:

1) INITIALIZATION

Let $\mathbf{x}_g = [\mathbf{x}_{1,g}, \mathbf{x}_{2,g}, \ldots, \mathbf{x}_{i,g}, \ldots, \mathbf{x}_{\mathrm{NP},g}]$ be the solution matrix at g-th generation. Each solution is composed of D parameters, namely, $\mathbf{x}_{i,g} = [x_1, x_2, \ldots, x_j, \ldots, x_D]$. In the initialization, the solutions must be comprehensively generated.

2) MUTATION PROCESS

Let $\mathbf{m}_{i,g} = [m_1, m_2, \ldots, m_j, \ldots, m_D]$ be the mutant vector. The mutation process is the modification of coordinates, i.e., the difference between two solutions in each generation. This difference between the two solutions is gained by scaling factor F and is then added by the other solution. Thus, the modified solution, called mutant vector $\mathbf{m}_{i,g}$, is given by

$$\mathbf{m}_{i,g} = \mathbf{x}_{a,g} + F(\mathbf{x}_{b,g} - \mathbf{x}_{c,g}), \tag{40}$$

where F is a constant and $F \in [0, 2]$ and $\mathbf{x}_{a,g}, \mathbf{x}_{b,g}, \mathbf{x}_{c,g}$ are the solution vectors, where $a, b, c \in \{1, 2, \ldots, \mathrm{NP}\}$.

3) CROSSOVER PROCESS

In this process, the trial vector $\mathbf{v}_{i,g} = [v_1, v_2, \ldots, v_j, \ldots, v_D]$ will be generated from the mutant vector $\mathbf{m}_{i,g}$ and target vector $\mathbf{x}_{i,g}$. The amount of mixture is called a crossover rate, $\mathrm{CR} \in (0, 1)$. The trial vector $\mathbf{v}_{i,g}$ is computed by

$$v_j = \begin{cases} m_j, & \text{if } \mathrm{rand}_j \le \mathrm{CR} \textit{ or } j = i_{\mathrm{rand}} \\ x_j, & \text{if } \mathrm{rand}_j > \mathrm{CR} \textit{ or } j \ne i_{\mathrm{rand}} \end{cases} \tag{41}$$

where rand_j is the value in a range $(0, 1)$ and i_{rand} is a random integer in the range $\{1, 2, \ldots, D\}$.

4) SELECTION PROCESS

The trial vector $\mathbf{v}_{i,g}$ and target vector $\mathbf{x}_{i,g}$ will be the input of the object function $\mathrm{f}(\cdot)$. The vector, which can maximize (minimize) the object function, is selected to be the new target vector $\mathbf{x}_{i,g+1}$ in the next generation i.e.,

$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{x}_{i,g}, & \text{if } \mathrm{f}(\mathbf{x}_{i,g}) > \mathrm{f}(\mathbf{v}_{i,g}) \\ \mathbf{v}_{i,g}, & \text{if } \mathrm{f}(\mathbf{x}_{i,g}) \le \mathrm{f}(\mathbf{v}_{i,g}) \end{cases}. \tag{42}$$

B. DESIGN OF MULTIPLE SETS OF READ VOLTAGES

Let $\mathbf{r}$ be the read voltages. For each specific PE cycle, the read voltages must be optimized by minimizing the error probability, e.g.,

$$\mathbf{r}^* = \arg\min_{\mathbf{r}} \mathrm{P}_{\mathrm{avg}}(\mathbf{r}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_{\mathrm{v}}, d_{\mathrm{c}}, l_{\max}), \tag{43}$$

where the $\mathrm{P}_{\mathrm{avg}}(\cdot)$ is the error probability estimation function given in Algorithm 1-3. Since the NAND flash memory exhibits an asymmetric channel, we will use the i.i.d. channel adapter [33] to convert the asymmetric channel to the symmetric channel. Algorithm 5 shows the differential evolution to find the optimal read voltages $\mathbf{r}^*$. Fig. 6 (a) and (b) show the optimized read voltage for MLC and TLC NAND flash memory with the Gaussian channel model. We use the (3, 30) LDPC codes for the target design of the proposed DE. From the results, the read voltages of the proposed DE are narrow at low SNR, whereas read voltages of MMI [9] are narrow at high SNR. Fig. 7 shows the multiple sets of read voltages obtained by MMI [9], entropy [10], and proposed DE for MLC NAND flash memory. We use the empirical channel model of NAND flash memory as presented in Section II. Note that the number of read voltages optimized by entropy must be an even number and the read voltages at each dominant overlapped region must be an even number due to the limitation of entropy computation. Each read voltages pair $(r_j,\ r_{j+1})$ of the entropy-based method is slightly getting wider when the PE cycle is increased. For the read voltages optimized by MMI, the first dominant overlapped region has only 1 read voltage since the error probability in this dominant overlapped region is small. Due to the high error probability, the second dominant overlapped region has 2 read voltages and the last dominant overlapped region has 3 read voltages. The read voltages of MMI are getting wider when the PE cycle is increased. For the proposed DE, the number of the read voltages in each dominant overlapped region equals the read voltages of MMI. However, the read voltages of the proposed DE are narrow at the high PE cycle.

Algorithm 5 Design of Read Voltages

Input : $\mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_{\mathrm{v}}$ or λ, $d_{\mathrm{c}}, l_{\max}$, NP, $g_{\max}$, F, D, j and CR.
Output : $\mathbf{r}^*$

```
Generate the read voltages for setting a solution vector $\mathbf{x}_{i,1}$,
  $i \in \{1, 2, 3, \ldots, \mathrm{NP}\}$.
for  $g = 1, 2, 3 \ldots, g_{\max}$
  for  $i = 1, 2, 3 \ldots, \mathrm{NP}$
    Compute the mutant vector $\mathbf{m}_{i,g}$, see (40).
    Compute the trial vector $\mathbf{v}_{i,g}$, see (41).
    if the multiple sets of read voltages are used.
      Calculate $\mathrm{f}_1 = \mathrm{P}_{\mathrm{avg}}(\mathbf{x}_{i,g}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_{\mathrm{v}}$ or $\lambda, d_{\mathrm{c}}, l_{\max})$
      and $\mathrm{f}_2 = \mathrm{P}_{\mathrm{avg}}(\mathbf{v}_{i,g}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_{\mathrm{v}}$ or $\lambda, d_{\mathrm{c}}, l_{\max})$,
      see Algorithm 1-3.
    else if the single set of read voltages is used.
      Calculate $\mathrm{f}_1 = \mathrm{PE}^*(\mathbf{x}_{i,g}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_{\mathrm{v}}$ or $\lambda, d_{\mathrm{c}}, l_{\max})$
      and $\mathrm{f}_2 = \mathrm{PE}^*(\mathbf{v}_{i,g}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_{\mathrm{v}}$ or $\lambda, d_{\mathrm{c}}, l_{\max})$,
      see Algorithm 4.
    end if
    Perform the selection process to generate $\mathbf{x}_{i,g+1}$, see (42).
  end for
end for
$\mathbf{r}^* = \mathbf{x}_{i,g_{\max}}, \forall i$.
```

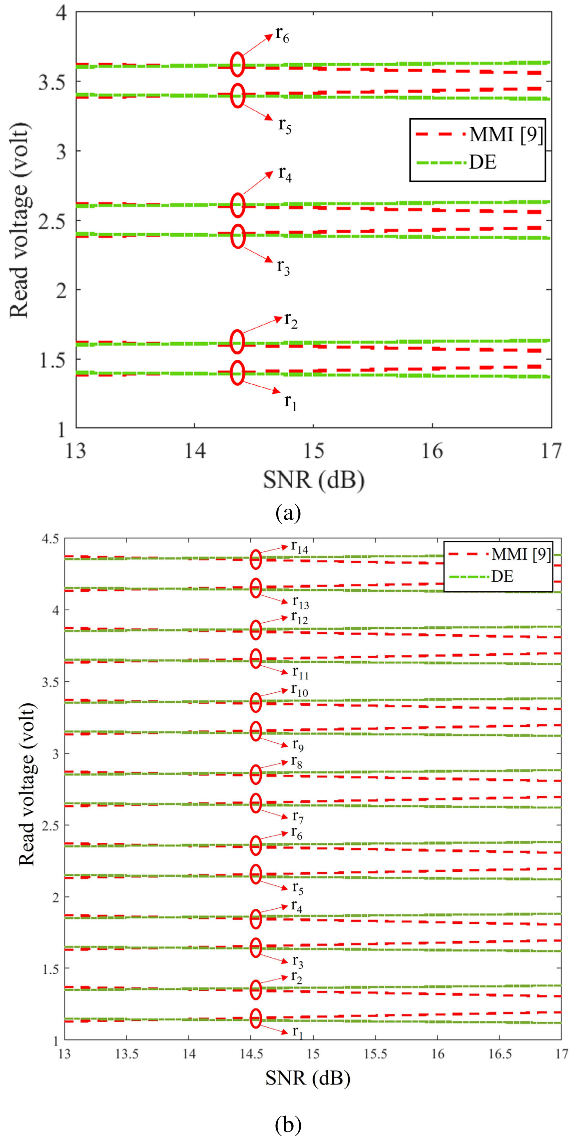
C. DESIGN OF SINGLE SET OF READ VOLTAGES

In this subsection, we consider the NAND flash memory with a limited number of the read voltages set. For example, only a single set of read voltages is used for every PE cycle. In this case, since the optimal read voltages depend on the PE cycle, we propose to optimize the read voltages by maximizing the decoding threshold of LDPC codes. The read voltages should be optimized such that the decoding threshold is maximized as follow

$$\mathbf{r}^* = \arg\max_{\mathbf{r}} \mathrm{PE}^*(\mathbf{r}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_{\mathrm{v}}, d_{\mathrm{c}}, l_{\max}), \tag{44}$$

where the $\mathrm{PE}^*(\cdot)$ is the decoding threshold exploring function in Algorithm 2. The details of read voltages optimization are shown in Algorithm 5.

Tables 1 and 2 demonstrate the optimized read voltages for MLC and TLC NAND flash memory with the Gaussian channel model. The read voltages in Tables 1 and 2 are designed for SNR = 16.130 dB and SNR = 16.862 dB, and the read voltages are on the green dot-dash line in Fig. 8 (a) and (b).



(a)

(b)

FIGURE 8. (a) The comparison of multiple sets of read voltages for MLC NAND flash memory with Gaussian channel model, (b) The comparison of multiple sets of read voltages for TLC NAND flash memory with Gaussian channel model.

Table 3 shows the read voltage for MLC NAND flash memories with an empirical channel model By using the same channel parameter in Fig. 2 (b), the decoding threshold is equal to 28,998 cycles. The read voltages in Table 3 are designed for PE = 28,998 cycles, and the read voltages are on the green dot line in Fig. 9.

VII. DESIGN OF LDPC CODES FOR READ VOLTAGES

This section will present the design of LDPC codes for NAND flash memory. First, we will show the design of irregular LDPC codes for the read voltages optimized by MMI and Entropy. However, in Section V, the read voltages must be designed by considering the structure of LDPC codes. Therefore, we then proposed the joint design of the read voltages and LDPC codes. Note that the MMI and Entropy do not support the joint design since the MMI does not consider the structure of LDPC codes and Entropy requires the Monte Carlo simulation. We begin to explain the design of irregular LDPC codes for the read voltages obtained from MMI and Entropy in subsection A. Then, the design of the protograph LDPC codes for the read voltages will be presented in subsection C.

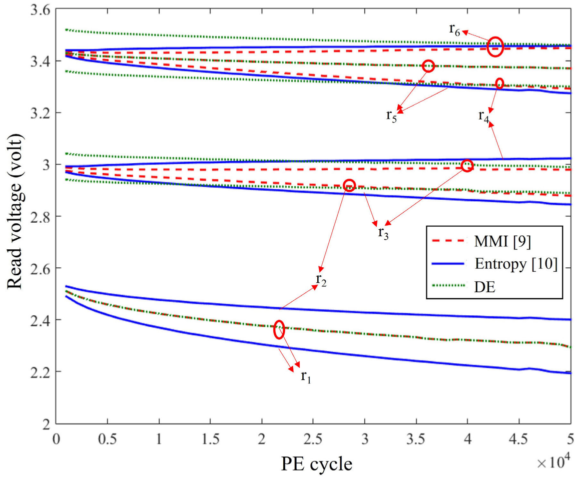


FIGURE 9. The comparison of the multiple sets of read voltages for MLC NAND flash memory with empirical channel model.

TABLE 1. A single set of read voltages for MLC NAND flash memory with gaussian channel model.

$\mathbf{r}^*$						SNR*
r_1	r_2	r_3	r_4	r_5	r_6	
1.3750	1.6250	2.375	2.625	3.375	3.625	16.130

A. DESIGN OF IRREGULAR LDPC CODES

Given the optimized read voltages, the irregular LDPC codes can be designed by maximizing the decoding threshold. This design method can be viewed as the design of irregular LDPC code for a single set of read voltages. We define $\Lambda = [\lambda_2, \lambda_3, \ldots, \lambda_{d_{\text{vmax}}}]$ as the variable node degree distribution, where the summation of degree distribution must be equal to 1. The variable node degree distribution is a nonnegative number and must satisfy the desired code rate R. Therefore, the irregular LDPC codes for given read voltages can be obtained from

$$\{\Lambda^*, d_c^*\} = \arg \max_{\{\Lambda, d_c\}} \text{PE}^*(\mathbf{r}^*, \text{p}_{11}, \text{p}_{10}, \text{p}_{00}, \text{p}_{01}, \Lambda, d_c, l_{\max}), \tag{45}$$

subject to $\sum_{i=2}^{d_{\text{vmax}}} \lambda_i = 1$, $R = 1 - \frac{1/d_c}{\sum_{i=2}^{d_{\text{vmax}}} \lambda_i / i}$ and $\lambda_2, \lambda_3, \ldots, \lambda_{d_{\text{vmax}}} \geq 0$.

Since differential evolution is an optimization tool with non-constraints, we then construct several constraints using the penalty function [34]. The penalty function D($\mathbf{x}$) will force the population of the solution to be the value in the feasible region. For inequality constraints, let $g_k(\mathbf{x})$ be the k-th

TABLE 2. A single set of read voltages for TLC flash memory with gaussian channel model.

$\mathbf{r}^*$							SNR*
r_1	r_2	r_3	r_4	r_5	r_6	r_7	
1.2375	1.2625	1.7375	1.7625	2.2375	2.2625	2.7375	
r_8	r_9	r_{10}	r_{11}	r_{12}	r_{13}	r_{14}	
2.7625	3.2375	3.2625	3.7375	3.7625	4.2375	4.2625	16.962

TABLE 3. A single set of read voltages for MLC flash memory with empirical channel model.

$\mathbf{r}^*$						PE*
r_1	r_2	r_3	r_4	r_5	r_6	
2.3940	2.9290	3.0090	3.3400	3.4000	3.4800	28,998

TABLE 4. Irregular LDPC codes for read voltages optimized by MMI, and entropy.

$\boldsymbol{\Lambda}^*$	MMI	Entropy
λ_2	0.0000	0.0000
λ_3	0.4458	0.4230
λ_4	0.3768	0.2622
λ_5	0.0191	0.2222
λ_6	0.0000	0.0000
λ_7	0.0000	0.0599
λ_8	0.0199	0.0092
λ_9	0.0177	0.0127
λ_{10}	0.1207	0.0107
d_c	38	38
PE*	30,928	33,614

constraint function of $\mathbf{x} = [x_1, x_2, \ldots, x_D]$. If the $\mathrm{g}_k(\mathbf{x}) \geq 0$, the constraint function will be $\mathrm{g}_k(\mathbf{x}) = \max\{0, \mathrm{g}_k(\mathbf{x})\}$. For the equality constraints, let $\mathrm{h}_l(\mathbf{x})$ be the l-th constraint function of $\mathbf{x}$. If $\mathrm{h}_l(\mathbf{x}) = 0$, the penalty function will be rewritten by $\mathrm{h}_l(\mathbf{x}) = |\mathrm{h}_l(\mathbf{x})|$. To design the LDPC codes, the penalty functions of inequality constraints and equality constraints are $\mathrm{g}_1(\lambda_2), \mathrm{g}_2(\lambda_3), \ldots, \mathrm{g}_1(\lambda_{d_{\mathrm{vmax}}}) \geq 0$, $\mathrm{h}_1(\lambda) = \lambda_2 + \ldots + \lambda_{d_{\mathrm{vmax}}} - 1 = 0$ and $\mathrm{h}_2(\lambda) = 1 - R - \frac{1/d_{\mathrm{c}}}{\lambda_2/2 + \ldots + \lambda_{d_{\mathrm{vmax}}}/d_{\mathrm{vmax}}} = 0$. Then, the penalty function is

$$\mathrm{PF}(\lambda) = F_p\left(\sum_{k=1}^{d_{\mathrm{vmax}}-1} \mathrm{g}_k(\lambda_{k+1}) + \sum_{l=1}^{2} \mathrm{h}_l(\lambda)\right). \tag{46}$$

where F_p is the penalty parameter. The details of the design of irregular LDPC codes are given in Algorithm 6. Table 4 shows the irregular LDPC codes designed for the read voltages in Fig. 9.

Algorithm 6 Design of Irregular LDPC Codes

```
Input : $\mathbf{r}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, R, l_{\max}, \mathrm{NP}, g_{\max}, F, D, j$ and CR.
Output : $\Lambda^*$ and $d_{\mathrm{c}}^*$.
1: Generate the degree distribution for setting a solution vector
     $\mathbf{x}_{i,1} = \{\Lambda_{i,g}, d_{\mathrm{c}_{i,g}}\}$.
2: for  $g = 1, 2, 3 \ldots, g_{max}$
3:   for  $i = 1, 2, 3 \ldots, \mathrm{NP}$
4:       Compute the mutant vector $\mathbf{m}_{i,g}$, see (40).
5:       Compute the trial vector $\mathbf{v}_{i,g}$, see (41).
6:       Compute $\mathrm{f}_1 = \mathrm{PE}^*(\mathbf{r}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, \mathbf{x}_{i,g}, l_{\max}) - \mathrm{PF}(\mathbf{x}_{i,g})$
         and $\mathrm{f}_2 = \mathrm{PE}^*(\mathbf{r}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, \mathbf{x}_{i,g}, l_{\max}) - \mathrm{PF}(\mathbf{v}_{i,g})$.
7:       Perform the selection process to generate $\mathbf{x}_{i,g+1}$, see (42).
8:   end for
9: end for
10: $\Lambda^* = \mathbf{x}_{i,g_{\max}},\ \forall i$ and $d_{\mathrm{c}}^* = \mathbf{x}_{i,g_{\max}},\ \forall i$.
```

B. DESIGN OF PROTOGRAPH LDPC CODES

An example base graph of an AR4JA code with a code rate of 0.9 is

$$\mathbf{B}_{\mathbf{AR4JA}} = \begin{pmatrix} 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 3 & 1 & 1 & 0 & 1 & 3 & 1 & 3 & \cdots & 1 \\ 1 & 2 & 2 & 0 & 1 & 1 & 3 & 1 & \cdots & 3 \end{pmatrix}, \tag{47}$$

where the first column is defined as a punctured node. To design the protograph code, the elements of the base graph $\mathrm{b}(i, j)$ are optimized. In this work, the numbers of rows and columns of the base matrix are set to be 3 and 21, respectively. The degree-1 and degree-2 variable nodes are added to the base graph to improve the decoding threshold and linear-minimum-distance property [35]. The 6-th to 21-th columns are identical to the AR4JA code. Therefore, our base graph is expressed by

$$\mathbf{B} = \begin{pmatrix} \mathrm{b}(1,1) & \mathrm{b}(1,2) & \mathrm{b}(1,3) & 1 & 0 & 0 & \cdots & 0 \\ \mathrm{b}(2,1) & \mathrm{b}(2,2) & \mathrm{b}(2,3) & 0 & 1 & 3 & \cdots & 1 \\ \mathrm{b}(3,1) & \mathrm{b}(3,2) & \mathrm{b}(3,3) & 0 & 1 & 1 & \cdots & 3 \end{pmatrix}, \tag{48}$$

where $\mathrm{b}(i, j) \in \{0, 1, 2, 3\}$. The decoding threshold of the protograph code is estimated from the DE as presented in Section IV-A). To design the protograph code, the elements of the base graph are searched in a way that the decoding threshold is maximized, i.e.

$$\mathbf{b}^* = \arg\max_{\{\mathbf{b}\}} \ \mathrm{PE}^*(\mathbf{r}^*, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, \mathbf{b}, \mathbf{B}, l_{\max}), \tag{49}$$

where $\mathbf{b} = [\mathrm{b}(1,1), \mathrm{b}(2,1), \mathrm{b}(3,1), \mathrm{b}(1,2), \mathrm{b}(2,2), \ldots, \mathrm{b}(3,3)]$. By using the differential evolution, the optimal base graph is obtained, easily. The design of the protograph LDPC code is shown in Algorithm 7. The designed base graph for the read voltages given from the MMI method is:

$$\mathbf{B}_{\mathbf{Design}} = \begin{pmatrix} 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 3 & 0 & 3 & 0 & 1 & 3 & 1 & 3 & \cdots & 1 \\ 1 & 3 & 0 & 0 & 1 & 1 & 3 & 1 & \cdots & 3 \end{pmatrix}.$$

VIII. JOINT DESIGN OF READ VOLTAGES AND LDPC CODES

In our read voltages optimization, we found that the optimized read voltages are related to the structure of the LDPC code. Therefore, the read voltages and LDPC code must be jointly designed to achieve the capacity of NAND flash memory. Using the DE algorithm, both the read voltages and LDPC codes can be designed easily.

Algorithm 7 Design of Protograph LDPC Codes

```
Input : $\mathbf{r}, p_{11}, p_{10}, p_{00}, p_{01}, R, l_{\max}, \mathrm{NP}, g_{\max}, F, D, j$ and CR.
Output : $\mathbf{b}^*$.
Generate the degree distribution for setting a solution vector
    $\mathbf{x}_{i,1} = \mathbf{b}_{i,g}$.
for $g = 1, 2, 3 \ldots, g_{max}$
    for $i = 1, 2, 3 \ldots, \mathrm{NP}$
        Compute the mutant vector $\mathbf{m}_{i,g}$, see (40).
        Compute the trial vector $\mathbf{v}_{i,g}$, see (41).
        Compute $f_1 = \mathrm{PE}^*(\mathbf{r}, p_{11}, p_{10}, p_{00}, p_{01}, \mathbf{x}_{i,g}, l_{\max})$
         and $f_2 = \mathrm{PE}^*(\mathbf{r}, p_{11}, p_{10}, p_{00}, p_{01}, \mathbf{x}_{i,g}, l_{\max})$
        Perform the selection process to generate $\mathbf{x}_{i,g+1}$, see (42).
    end for
end for
$\mathbf{b}^* = \mathbf{x}_{i,g_{\max}}, \ \forall i$.
```

A. JOINT DESIGN OF READ VOLTAGES AND IRREGULAR LDPC CODES

To achieve the superior FER performance of NAND flash memory, we will propose the joint optimization of read voltages and LDPC codes from

$$\{\Lambda^*, d_c^*, \mathbf{r}^*\} = \arg \max_{\{\Lambda, d_c, \mathbf{r}\}} \ \mathrm{PE}^*(\mathbf{r}, p_{11}, p_{10}, p_{00}, p_{01}, \Lambda, d_c, l_{\max}). \tag{50}$$

The details of the joint design are presented in Algorithm 8. The results of the joint design for MLC NAND flash memory with an empirical channel model are shown in Table 5. As a result, the decoding threshold of the joint design is greater than that of the other design in Table 4.

Algorithm 8 Joint Design of Read Voltages and Irregular LDPC Codes

```
Input : $p_{11}, p_{10}, p_{00}, p_{01}, R, l_{\max}, \mathrm{NP}, g_{\max}, F, D_1, D_2, j$ and CR.
Output : $\Lambda^*, d_c^*$ and $\mathbf{r}^*$.
Generate the degree distribution and read voltages for setting
  a solution vector $\mathbf{x}_{i,1}^{(1)} = \{\Lambda_{i,g}, d_{c_{i,g}}\}$ and
  $\mathbf{x}_{i,1}^{(2)} = \mathbf{r}, i \in \{1, 2, 3, \ldots, \mathrm{NP}\}$.
for $g = 1, 2, 3 \ldots, g_{max}$
    for $i = 1, 2, 3 \ldots, \mathrm{NP}$
        Compute the mutant vector $\mathbf{m}_{i,g}^{(1)}$ and $\mathbf{m}_{i,g}^{(2)}$, see (40).
        Compute the trial vector $\mathbf{v}_{i,g}^{(1)}$ and $\mathbf{v}_{i,g}^{(2)}$, see (41).
        Compute $f_1 = \mathrm{PE}^*(\mathbf{x}_{i,g}^{(2)}, p_{11}, p_{10}, p_{00}, p_{01}, \mathbf{x}_{i,g}^{(1)}, l_{\max})$
            $-\mathrm{PF}(\mathbf{x}_{i,g}^{(1)})$ and
          $f_2 = \mathrm{PE}^*(\mathbf{v}_{i,g}^{(2)}, p_{11}, p_{10}, p_{00}, p_{01}, \mathbf{v}_{i,g}^{(1)}, l_{\max}) - \mathrm{PF}(\mathbf{v}_{i,g}^{(1)})$.
        Perform the selection process to generate $\mathbf{x}_{i,g+1}$, see (42).
    end for
end for
$\Lambda^* = \mathbf{x}_{i,g_{\max}}^{(1)}, \ \forall i, d_c^* = \mathbf{x}_{i,g_{\max}}^{(1)}, \ \forall i$ and $\mathbf{r}^* = \mathbf{x}_{i,g_{\max}}^{(2)}, \ \forall i$.
```

TABLE 5. Read voltages and irregular LDPC codes optimized by DE for MLC NAND flash memory with empirical channel model.

Λ^*		$\mathbf{r}^*$	
λ_2	0.0000	r_1	2.3770
λ_3	0.2762	r_2	2.9250
λ_4	0.3157	r_3	3.0150
λ_5	0.0458	r_4	3.3250
λ_6	0.0471	r_5	3.3950
λ_7	0.0095	r_6	3.4850
λ_8	0.0916		
λ_9	0.0002		
λ_{10}	0.2138		
d_c	45		
PE*	35,383		

B. JOINT DESIGN OF READ VOLTAGES AND PROTOGRAPH LDPC CODES

The read voltages and the base graph are optimized in a way that the decoding threshold is maximized, i.e.

$$\{\mathbf{b}^*, \mathbf{r}^*\} = \arg \max_{\{\mathbf{b}, \mathbf{r}\}} \ \mathrm{PE}^*(\mathbf{r}, p_{11}, p_{10}, p_{00}, p_{01}, \mathbf{b}, \mathbf{B}, l_{\max}). \tag{51}$$

The read voltages and the base graph can be found by differential evolution. The procedure of joint design is shown in Algorithm 9.

Algorithm 9 Joint Design of Read Voltages and Protograph LDPC Codes

```
Input : $p_{11}, p_{10}, p_{00}, p_{01}, R, l_{\max}, \mathrm{NP}, g_{\max}, F, D_1, D_2, j$ and CR.
Output : $\mathbf{b}^*$ and $\mathbf{r}^*$.
Generate the degree distribution and read voltages for setting
  a solution vector $\mathbf{x}_{i,1}^{(1)} = \mathbf{b}_{i,g}$ and $\mathbf{x}_{i,1}^{(2)} = \mathbf{r}, i \in \{1, 2, 3, \ldots, \mathrm{NP}\}$.
for $g = 1, 2, 3 \ldots, g_{max}$
    for $i = 1, 2, 3 \ldots, \mathrm{NP}$
        Compute the mutant vector $\mathbf{m}_{i,g}^{(1)}$ and $\mathbf{m}_{i,g}^{(2)}$, see (40).
        Compute the trial vector $\mathbf{v}_{i,g}^{(1)}$ and $\mathbf{v}_{i,g}^{(2)}$, see (41).
        Compute $f_1 = \mathrm{PE}^*(\mathbf{x}_{i,g}^{(2)}, p_{11}, p_{10}, p_{00}, p_{01}, \mathbf{x}_{i,g}^{(1)}, l_{\max})$ and
         $f_2 = \mathrm{PE}^*(\mathbf{v}_{i,g}^{(2)}, p_{11}, p_{10}, p_{00}, p_{01}, \mathbf{v}_{i,g}^{(1)}, l_{\max})$.
        Perform the selection process to generate $\mathbf{x}_{i,g+1}$, see (42).
    end for
end for
$\mathbf{b}^* = \mathbf{x}_{i,g_{\max}}^{(1)}, \ \forall i$ and $\mathbf{r}^* = \mathbf{x}_{i,g_{\max}}^{(2)}, \ \forall i$.
```

The joint design of read voltages and protograph are:

$$\mathbf{B}_{\mathbf{Joint}} = \begin{pmatrix} 1 & 2 & 3 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 3 & 1 & 0 & 1 & 3 & 1 & 3 & \cdots & 1 \\ 1 & 2 & 3 & 0 & 1 & 1 & 3 & 1 & \cdots & 3 \end{pmatrix},$$

and $\mathbf{r}^* = [2.2355, 2.8965, 3.0124, 3.3221, 3.3981, 3.4837]$.

IX. MULTIPLE DECODING

The increasing number of read voltages can increase the error-correcting capability of LDPC codes. However, the reading speed of NAND flash memory will be decreased. Therefore, in practical implementation, the reading speed can be confined by using multiple decoding [8]. A few read voltages are used at the first decoding. If the LDPC decoder cannot correct the erroneous bits, the memory cell will be reread with different read voltages to obtain more precise soft information. For example, in the first read, the LDPC decoder can use the LLR estimated from the read voltages $\mathbf{r}_f = [r_{f_1}, r_{f_2}, r_{f_3}]$. Thus, the memory cell provides hard information. If the first decoding is failed, the memory cell will be reread with additional read voltages $\mathbf{r}_s = [r_{s_1}, r_{s_2}, r_{s_3}]$. Here, the soft information of the second decoding will be computed from the read voltages $\mathbf{r} = [r_{f_1}, r_{f_2}, r_{f_3}, r_{s_1}, r_{s_2}, r_{s_3}]$.

A. DESIGN OF READ VOLTAGES FOR NAND FLASH MEMORY WITH MULTIPLE DECODING

There are two approaches for optimizing the read voltages. In the first approach, the read voltages $\mathbf{r}_f$ of the first decoding are selected so that the decoding threshold is at the highest level. We will optimize the single set of read voltages in such a way that the PE*($\mathbf{x}$) of the first decoding is maximized, namely,

$$\mathbf{r}_f^* = \arg\max_{\mathbf{r}_f} \; \mathrm{PE}^*(\mathbf{r}_f, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_\mathrm{v}, d_\mathrm{c}, l_{\max}), \quad (52)$$

where $\mathbf{r}_f^*$ represents the optimal read voltages by performing Algorithm 5. If the first decoding is failed, the additional read voltages $\mathbf{r}_s$ will be applied. Thus, the memory cells seem to be read with $\mathbf{r} = [\mathbf{r}_f^*, \mathbf{r}_s]$, where the $\mathbf{r}_s$ can be optimized from

$$\mathbf{r}_s^* = \arg\max_{\mathbf{r}_s} \; \mathrm{PE}^*(\mathbf{r}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_\mathrm{v}, d_\mathrm{c}, l_{\max}). \quad (53)$$

In the second approach, we will optimize the read voltage by considering the performance of the second decoding. This approach allows the first decoding to have inferior FER performance. The read voltages $\mathbf{r}^*$ are firstly designed by

$$\mathbf{r}^* = \arg\max_{\mathbf{r}} \; \mathrm{PE}^*(\mathbf{r}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_\mathrm{v}, d_\mathrm{c}, l_{\max}). \quad (54)$$

To find the optimal read voltages $\mathbf{r}_f^*$, we will select the $\mathbf{r}_f$ from $\mathbf{r}^*$ in such a way that the PE*($\mathbf{x}$) is maximized, i.e.,

$$\mathbf{r}_f^* = \arg\max_{\mathbf{r}_f \in \mathbf{r}^*} \; \mathrm{PE}^*(\mathbf{r}_f, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, d_\mathrm{v}, d_\mathrm{c}, l_{\max}). \quad (55)$$

Therefore, the optimal read voltages $\mathbf{r}_s^*$ are the rest of $\mathbf{r}^*$. The optimized read voltages from the first approach and the second approach are shown in Table 6. Note that PE*($\cdot$) can be replaced by the $\mathrm{P}_{\mathrm{avg}}(\cdot)$ if multiple sets of read voltages are used. For the protograph LDPC codes, PE*($\cdot$) is obtained from the density evolution for the protograph LDPC code given in section III-A).

B. JOINT DESIGN OF READ VOLTAGES AND LDPC CODES FOR NAND FLASH MEMORY WITH MULTIPLE DECODING

In this subsection, we will explain the joint design for multiple decoding. We will show the joint design of read voltages and irregular LDPC code only. For the joint design of read voltages and protograph LDPC code, the proposed design can be applied straightforwardly. Our joint design can be divided into 2 approaches. In the first approach, the read voltages $\mathbf{r}_f = [r_{f_1}, r_{f_2}, r_{f_3}]$ and LDPC code with d_c and $\Lambda = [\lambda_2, \lambda_3, \ldots, \lambda_{d_{\mathrm{vmax}}}]$ are jointly optimized by using Algorithm 5, i.e.,

TABLE 6. The single set of read voltages for (3, 30) LDPC code for multiple decoding.

First approach				Second approach			
$\mathbf{r}_f^*$		$\mathbf{r}^*$		$\mathbf{r}_f^*$		$\mathbf{r}^*$	
r_{f_1}	2.4180	r_{f_1}	2.4180	r_{f_1}	2.3940	r_{f_1}	2.3940
r_{f_2}	2.9640	r_{f_2}	2.9640	r_{f_2}	2.9290	r_{f_2}	2.9290
r_{f_3}	3.4070	r_{f_3}	3.4070	r_{f_3}	3.4000	r_{f_3}	3.4000
		r_{s_1}	2.9240			r_{s_1}	3.0090
		r_{s_2}	3.3470			r_{s_2}	3.3400
		r_{s_3}	3.4870			r_{s_3}	3.4800
PE*	17,481	PE*	28,367	PE*	16,725	PE*	28,998

$$\{\Lambda^*, d_\mathrm{c}^*, \mathbf{r}_f^*\} = \arg\max_{\{\Lambda, d_c, \mathbf{r}_f\}} \; \mathrm{PE}^*(\mathbf{r}_f, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, \Lambda, d_\mathrm{c}, l_{\max}). \quad (56)$$

The read voltages $\mathbf{r}_s = [r_{s_1}, r_{s_2}, r_{s_3}]$ of second decoding are selected by using Algorithm 5, i.e.,

$$\mathbf{r}_s^* = \arg\max_{\mathbf{r}_s} \; \mathrm{PE}^*(\mathbf{r}, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, \Lambda^*, d_\mathrm{c}^*, l_{\max}), \quad (57)$$

where the $\mathbf{r}_f$ and d_c^*, Λ^* are fixed. The design results are shown in Table 6.

In the second approach, the read voltages $\mathbf{r} = [r_{f_1}, r_{f_2}, r_{f_3}, r_{s_1}, r_{s_2}, r_{s_3}]$ and LDPC code with d_c and $\Lambda = \{\lambda_2, \lambda_3, \ldots, \lambda_{d_{\mathrm{vmax}}}\}$ are jointly optimized using (53). Then, the read voltages $\mathbf{r}_f = [r_{f_1}, r_{f_2}, r_{f_3}]$ of the first decoding are obtained by selecting 3 read voltages from $\mathbf{r}^*$, whereby the PE*($\mathbf{x}$) is maximized, namely,

$$\mathbf{r}_f^* = \arg\max_{\mathbf{r}_f \in \mathbf{r}^*} \; \mathrm{PE}^*(\mathbf{r}_f, \mathrm{p}_{11}, \mathrm{p}_{10}, \mathrm{p}_{00}, \mathrm{p}_{01}, \Lambda^*, d_\mathrm{c}^*, l_{\max}). \quad (58)$$

and the second read voltages are the rest of $\mathbf{r}^*$ i.e., $\mathbf{r}_s^* = \mathbf{r}^* \setminus \mathbf{r}_f^*$. The design results of the second approach are shown in Table 7.

X. SIMULATIONS AND RESULTS

For computer-based simulation, we set the channel parameters as described in section II. The (3,30) LDPC codes are constructed by using the PEG algorithm [31]. The number of information bits is 4,096 for regular and irregular codes. For the AR4JA code, OARA code, proposed protograph code, and joint design of read voltage and protograph code, the number of information bits is 4,000. The number of iterative decoding is 10 iterations for all simulations except the simulations in Fig. 13 which used 50 iterations.

A. DESIGN OF READ VOLTAGES

For MLC and TLC NAND flash memory with Gaussian channel model, the FER performance of (3, 30) LDPC code with the read voltages optimized by MMI and DE are shown in Fig.10. We use the read voltages presented in Section VB. The DE provides better FER performance than MMI for both

TABLE 7. Joint design of read voltages and irregular LDPC code for multiple decoding.

First approach				Second approach			
$\mathbf{\Lambda}^*$		$\mathbf{r}_f^*$		$\mathbf{\Lambda}^*$		$\mathbf{r}_f^*$	
λ_2	0	r_{f_1}	2.3980	λ_2	0	r_{f_1}	2.3770
λ_3	0.3215	r_{f_2}	2.9590	λ_3	0.2762	r_{f_2}	2.9250
λ_4	0.2430	r_{f_3}	3.4010	λ_4	0.3157	r_{f_3}	3.3950
λ_5	0			λ_5	0.0458		
λ_6	0.1217			λ_6	0.0471		
λ_7	0.0437			λ_7	0.0095		
λ_8	0.0296			λ_8	0.0916		
λ_9	0			λ_9	0.0002		
λ_{10}	0.2405			λ_{10}	0.2138		
d_c	45	PE*	22,817	d_c	45	PE*	22,251
		$\mathbf{r}^*$				$\mathbf{r}^*$	
		r_{f_1}	2.3980			r_{f_1}	2.3770
		r_{f_2}	2.9590			r_{f_2}	2.9250
		r_{f_3}	3.4010			r_{f_3}	3.3950
		r_{s_1}	2.8990			r_{s_1}	3.0150
		r_{s_2}	3.3210			r_{s_2}	3.3250
		r_{s_3}	3.4910			r_{s_3}	3.4850
		PE*	34,317			PE*	35,383

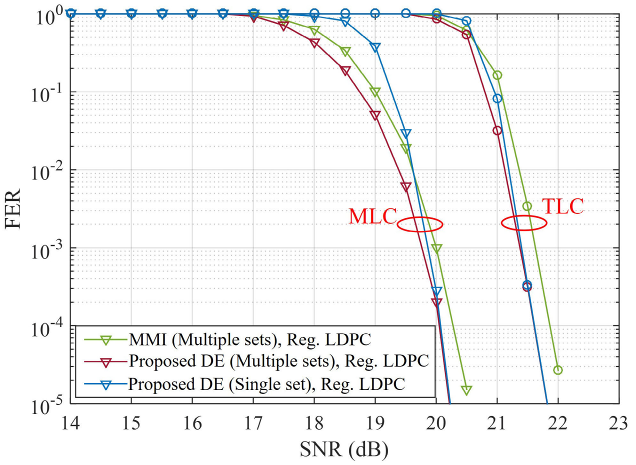


FIGURE 10. FER comparisons of the read voltages optimized by MMI and DE for MLC and TLC Gaussian channels.

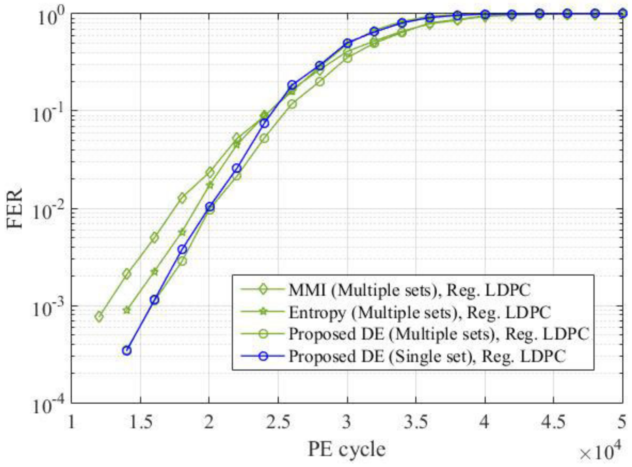


FIGURE 11. FER comparisons of the read voltages optimized by MMI, Entropy, and DE for MLC empirical channel.

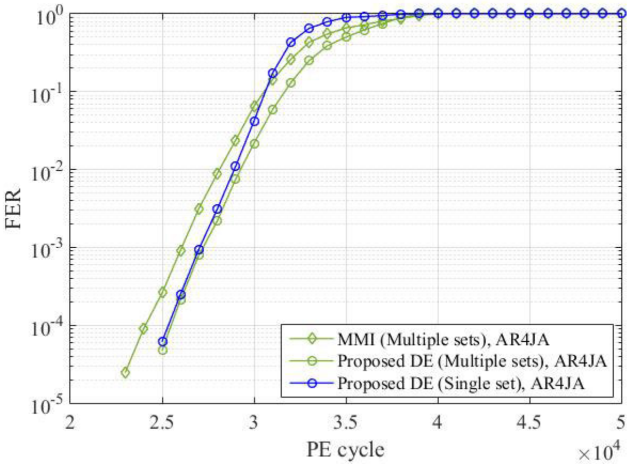


FIGURE 12. FER comparisons of the read voltages optimized by MMI, and DE for MLC empirical channel.

MLC and TLC NAND flash memory. Compared to the multiple sets of read voltage, the FER performance of the single set of the read voltage of DE at the high PE cycle is inferior. However, at the low PE cycle, the FER performance of the single set of read voltage is converged to the FER performance of the multiple sets of the read voltage. Fig. 11 shows the FER performance comparisons of the read voltages obtained by MMI [9], Entropy [10], and proposed DE for MLC NAND flash memory with the empirical channel model. We use the read voltages presented in Fig. 9 and Table 3. Since the read voltages optimized by DE are well designed for regular LDPC code with $d_v = 3$ and $d_c = 30$, the FER performance of DE is lower than that of MMI and Entropy. At FER $= 10^{-3}$, the FER performance gains are equal to 2,000 PE cycles and 1,000 PE cycles, respectively. For the single set of read voltages, the FER performance is higher than that of multiple sets of read voltages at a high PE cycle and exhibits the same FER performance at a low PE cycle. The FER performance gain between a single set and multiple sets of read voltage is equal to 1,000 PE cycles. Fig. 12 shows the FER performances of MMI and DE for the AR4JA code. For multiple sets of read voltages, the FER performance of DE is better than that of MMI around 1,000 PE cycles at 10^{-4}.

We also demonstrate that our design of read voltages can perform well for any decoder. The FER performances of the (3, 6) LDPC code with BP, MS, NMS, and OMS decoders are compared in Fig. 13. Note that the maximum number of iterations is set to be 50 for the convergence of the decoding algorithm. The parametric design such as α and β are re-designed for MLC empirical channel. The results show that the read voltages optimized by DE provide better FER performance than MMI [9]. Particularly, the OMS with MMI method [9] offers an FER performance similar to the NMS with DE method.

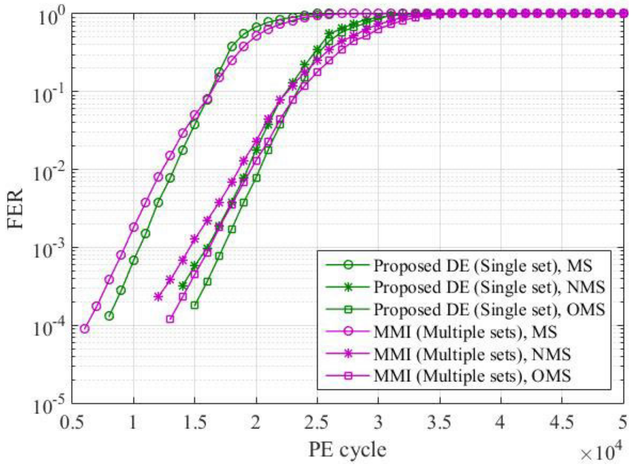


FIGURE 13. FER comparisons of regular (3, 6) LDPC codes with MS, NMS, and OMS decoder in MLC empirical channel.

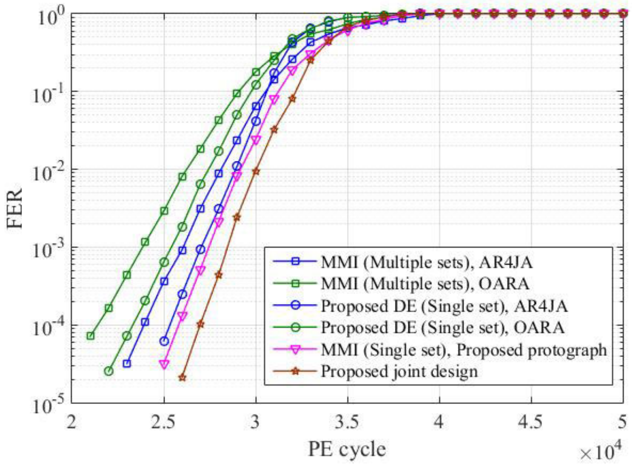


FIGURE 15. FER comparisons of protograph LDPC codes and joint design for MLC empirical channel.

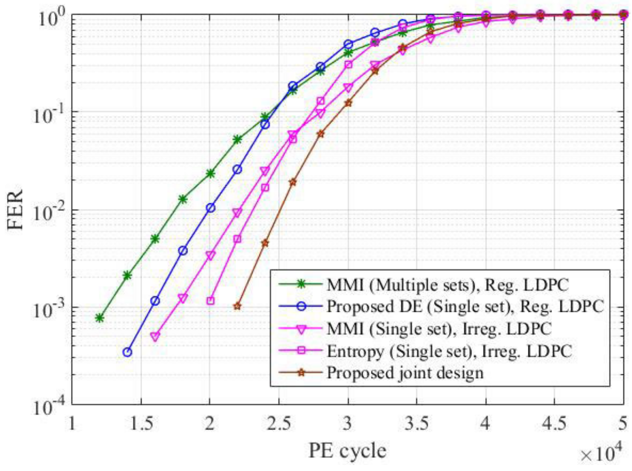


FIGURE 14. FER comparisons of irregular LDPC codes and proposed joint design for MLC empirical channel.

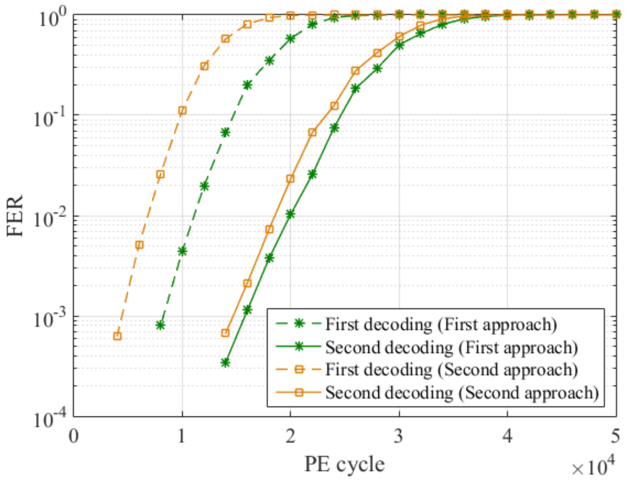


FIGURE 16. FER comparison of the read voltages optimized by DE for multiple decoding.

B. JOINT DESIGN OF READ VOLTAGES AND LDPC CODES

In Fig. 14, we evaluate the FER performance of irregular LDPC codes and our joint design as presented in Tables 4 and 5. We found that the FER performance of irregular LDPC code with Entropy is lower than that of irregular LDPC code with MMI at the small PE cycle. The joint design provides low FER compared to irregular LDPC code with Entropy at any PE cycle. These results correspond to the PE^* in Tables 4 and 5. Fig. 15 shows the FER performance of the AR4JA code [18], OARA code [19], the protograph LDPC code and the joint design of read voltages and protograph LDPC code. The simulation results demonstrate that the FER performance of protograph code is superior to the AR4JA code. Since the OARA code is designed for the BICM-ID system, the FER performance of the OARA code is inferior to the AR4JA code for the BICM system. At FER $= 10^{-4}$, the performance gain of our joint design is 4,300 cycles compared to the AR4JA code for MMI.

C. MLC NAND FLASH MEMORY WITH MULTIPLE DECODING

In this subsection, we will simulate the performance of NAND flash memory with multiple decoding. In Fig. 16, we compare the FER performance of the read voltages optimized by DE as presented in Table 6. In the first approach, the performances of the first decoding are mainly considered. Therefore, the FER performance of the first approach is lower than that of the second approach. At FER $= 10^{-3}$, the difference between the first and second approaches is 4,000 cycles. Since the second approach aims to enhance the FER performance of the second decoding. The second approach then provides a better FER performance than the first approach. The performance gain is 1,000 cycles at FER $= 10^{-3}$. Corresponding to Table 6, the PE* of the first approach is higher than the second approach for the first decoding and the PE* of the second approach is higher than the first approach for the second decoding.

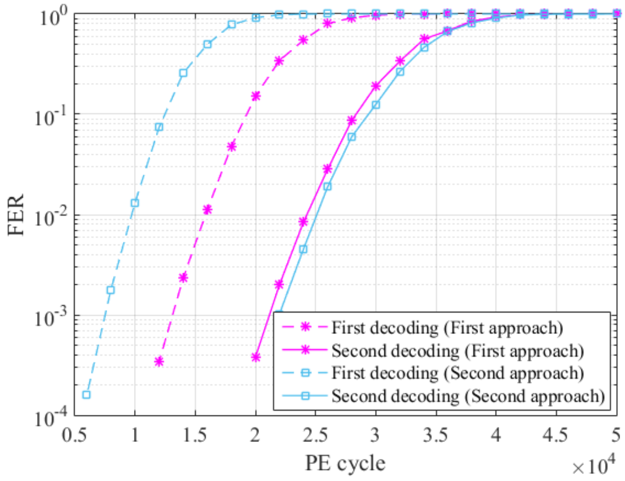


FIGURE 17. FER comparisons of joint design for multiple decoding.

We further simulate the joint design of read voltages and irregular LDPC code presented in Table 7. The FER performance of the joint design is shown in Fig. 17. For the first decoding, the joint design with the first approach outperforms the second approach. The performance gain is 6,000 cycles at FER $= 10^{-3}$. For the second decoding, the joint design with the second approach outperforms the first approach. The performance gain is 1,000 cycles at FER $= 10^{-3}$. These results coincide with the PE* in Table 7, where the first decoding of the second approach provides the lowest PE^* and the second decoding of the second approach provides the highest PE*.

XI. CONCLUSION

We present the DE method for the design of read voltages. Our DE can identify the suitable read voltages for MLC and TLC NAND flash memory. The simulation results show that the DE outperforms the MMI and Entropy in terms of FER. The DE provides performance improvements between 1,000 and 3,000 cycles at FER $= 10^{-3}$. We also present the joint design of read voltages and LDPC codes for MLC and TLC NAND flash memory. The performance improvement obtained from our joint design is around 2,500 and 4,600 cycles when compared to Entropy and MMI, respectively.

REFERENCES

[1] K. Kanda, M. Koyanagi, T. Yamamura, K. Hosono, M. Yoshihara, T. Miwa, Y. Kato, A. Mak, S. L. Chan, F. Tsai, and R. Cernea, "A 120 mm^2 16 Gb 4-MLC NAND flash memory with 43 nm CMOS technology," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2008, pp. 430–431.

[2] S.-H. Chang, S. K. Lee, S. J. Park, M. J. Jung, J. C. Han, I. S. Wang, J. H. Lee, J. H. Kim, W. K. Kang, T. K. Kang, and H. S. Byun, "A 48 nm 32 Gb 8-level NAND flash memory with 5.5 MB/s program throughput," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2009, pp. 240–241.

[3] Y. Takai, M. Fukuchi, R. Kinoshita, C. Matsui, and K. Takeuchi, "Analysis on heterogeneous SSD configuration with quadruple-level cell (QLC) NAND flash memory," in *Proc. IEEE 11th Int. Memory Workshop (IMW)*, May 2019, pp. 1–4.

[4] X. Wang, G. Dong, L. Pan, and R. Zhou, "Error correction codes and signal processing in flash memory," in *Flash Memories*. Rijeka, Croatia: InTech, Sep. 2011, doi: 10.5772/19083.

[5] G. Dong, Y. Pan, N. Xie, C. Varanasi, and T. Zhang, "Estimating information-theoretical NAND flash memory storage capacity and its implication to memory system design space exploration," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 20, no. 9, pp. 1705–1714, Sep. 2012.

[6] Q. Xu, P. Gong, T. M. Chen, J. Michael, and S. Li, "Modelling and characterization of NAND flash memory channels," *Measurement*, vol. 70, pp. 225–231, Jun. 2015.

[7] G. Dong, N. Xie, and T. Zhang, "On the use of soft-decision error-correction codes in NAND flash memory," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 2, pp. 429–439, Feb. 2011.

[8] Q. Li, Q. Wang, Q. Xu, and Z. Huo, "A fast read retry method for 3D NAND flash memories using novel valley search algorithm," *IEICE Electron. Exp.*, vol. 15, no. 22, 2018, Art. no. 20180921.

[9] J. Wang, K. Vakilinia, T.-Y. Chen, T. Courtade, G. Dong, T. Zhang, H. Shankar, and R. Wesel, "Enhanced precision through multiple reads for LDPC decoding in flash memories," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 5, pp. 880–891, May 2014.

[10] C. A. Aslam, Y. L. Guan, and K. Cai, "Read and write voltage signal optimization for multi-level-cell (MLC) NAND flash memory," *IEEE Trans. Commun.*, vol. 64, no. 4, pp. 1613–1623, Apr. 2016.

[11] K. Wei, J. Li, L. Kong, F. Shu, and Y. Li, "Read-voltage optimization for finite code length in MLC NAND flash memory," in *Proc. IEEE Inf. Theory Workshop (ITW)*, Nov. 2018, pp. 1–5, doi: 10.1109/ITW.2018.8613523.

[12] C. Duangthong, W. Phakphisut, and P. Supnithi, "Read voltage optimization in MLC NAND flash memory via the density evolution," in *Proc. 26th Int. Conf. Telecommun. (ICT)*, Apr. 2019, pp. 361–365.

[13] Y. Yeh, A. Fazeli, and P. H. Siegel, "Optimal placement of read thresholds for coded NAND flash memory," in *Proc. IEEE Int. Conf. Commun.*, Jun. 2021, pp. 1–7, doi: 10.1109/ICC42927.2021.9500536.

[14] Z. Mei, K. Cai, and X. He, "Deep learning-aided dynamic read thresholds design for multi-level-cell flash memories," *IEEE Trans. Commun.*, vol. 68, no. 5, pp. 2850–2862, May 2020.

[15] M. P. C. Fossorier, M. Mihaljevic, and H. Imai, "Reduced complexity iterative decoding of low-density parity check codes based on belief propagation," *IEEE Trans. Commun.*, vol. 47, no. 5, pp. 673–680, May 1999.

[16] J. Chen and M. P. C. Fossorier, "Near optimum universal belief propagation based decoding of low-density parity check codes," *IEEE Trans. Commun.*, vol. 50, no. 3, pp. 406–414, Mar. 2002.

[17] J. Chen and M. P. C. Fossorier, "Density evolution for two improved BP-based decoding algorithms of LDPC codes," *IEEE Commun. Lett.*, vol. 6, no. 5, pp. 208–210, May 2002, doi: 10.1109/4234.1001666.

[18] D. Divsalar, S. Dolinar, C. Jones, and K. Andrews, "Capacity-approaching protograph codes," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 6, pp. 876–888, Aug. 2009, doi: 10.1109/JSAC.2009.090806.

[19] Y. Bu, Y. Fang, G. Han, S. Mumtaz, and M. Guizani, "Design of protograph-LDPC-based BICM-ID for multi-level-cell (MLC) NAND flash memory," *IEEE Commun. Lett.*, vol. 23, no. 7, pp. 1127–1131, Jul. 2019, doi: 10.1109/LCOMM.2019.2915223.

[20] G. Caire, G. Taricco, and E. Biglieri, "Bit-interleaved coded modulation," *IEEE Trans. Inf. Theory*, vol. 44, no. 3, pp. 927–946, May 1998, doi: 10.1109/18.669123.

[21] Y. Fang, G. Zhang, G. Cai, F. C. M. Lau, P. Chen, and G. Han, "Root-protograph-based BICM-ID: A reliable and efficient transmission solution for block-fading channels," *IEEE Trans. Commun.*, vol. 67, no. 9, pp. 5921–5939, Sep. 2019, doi: 10.1109/TCOMM.2019.2923778.

[22] Y. Luo, S. Ghose, Y. Cai, E. F. Haratsch, and O. Mutlu, "Enabling accurate and practical online flash channel modeling for modern MLC NAND flash memory," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 9, pp. 2294–2311, Sep. 2016, doi: 10.1109/JSAC.2016.2603608.

[23] Y. Cai, Y. Luo, E. F. Haratsch, K. Mai, and O. Mutlu, "Data retention in MLC NAND flash memory: Characterization, optimization, and recovery," in *Proc. IEEE 21st Int. Symp. High Perform. Comput. Archit. (HPCA)*, San Francisco, CA, USA, Feb. 2015, pp. 551–563.

[24] K.-T. Park, M. Kang, D. Kim, S.-W. Hwang, B. Y. Choi, Y.-T. Lee, C. Kim, and K. Kim, "A zeroing cell-to-cell interference page architecture with temporary LSB storing and parallel MSB program scheme for MLC NAND flash memories," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 919–928, Apr. 2008.

[25] R.-A. Cernea, L. Pham, F. Moogat, S. Chan, B. Le, Y. Li, S. Tsao, T. Y. Tseng, K. Nguyen, J. Li, and J. Hu, "A 34 MB/s MLC write throughput 16 Gb NAND with all bit line architecture on 56 nm technology," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 186–194, Jan. 2009.

[26] C. M. Compagnoni, M. Ghidotti, A. L. Lacaita, A. S. Spinelli, and A. Visconti, "Random telegraph noise effect on the programmed threshold-voltage distribution of flash memories," *IEEE Electron Device Lett.*, vol. 30, no. 9, pp. 984–986, Sep. 2009.

[27] N. Mielke, H. Belgal, A. Fazio, Q. Meng, and N. Righos, "Recovery effects in the distributed cycling of flash memories," in *Proc. IEEE Int. Rel. Phys. Symp.*, Mar. 2006, pp. 29–35.

[28] T. J. Richardson, M. A. Shokrollahi, and R. L. Urbanke, "Design of capacity-approaching irregular low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 619–637, Feb. 2001.

[29] F. Angarita, J. Valls, V. Almenar, and V. Torres, "Reduced-complexity min-sum algorithm for decoding LDPC codes with low error-floor," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 7, pp. 2150–2158, Jul. 2014, doi: 10.1109/TCSI.2014.2304660.

[30] J. Hagenauer, "The exit chart–introduction to extrinsic information transfer in iterative processing," in *Proc. 12th Eur. Signal Process. Conf.*, Sep. 2004, pp. 1541–1548.

[31] X.-Y. Hu, E. Eleftheriou, and D. M. Arnold, "Regular and irregular progressive edge-growth tanner graphs," *IEEE Trans. Inf. Theory*, vol. 51, no. 1, pp. 386–398, Jan. 2005.

[32] R. Storn and K. Price, "Differential evolution—A simple and efficient heuristic adaptive scheme for global optimization over continuous spaces," *J. Global Optimiz.*, vol. 11, pp. 341–359, Dec. 1997.

[33] J. Hou, P. H. Siegel, L. B. Milstein, and H. D. Pfister, "Capacity-approaching bandwidth-efficient coded modulation schemes based on low-density parity-check codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 9, pp. 2141–2155, Sep. 2003.

[34] C. Deng, C. Liang, B. Zhao, and A. Deng, "New penalty function with differential evolution for constrained optimization," in *Proc. 7th World Congr. Intell. Control Autom.*, Chongqing, China, 2008, pp. 5304–5307.

[35] D. Divsalar, C. Jones, S. Dolinar, and J. Thorpe, "Protograph based LDPC codes with minimum distance linearly growing with block size," in *Proc. IEEE Global Telecommun. Conf.*, Nov. 2005, p. 5, doi: 10.1109/GLOCOM.2005.1577834.

CHATUPORN DUANGTHONG received the B.S. and M.S. degrees in telecommunication engineering from the King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, in 2015 and 2017, respectively. His research interests include channel coding, storage technology, and signal processing.

WATID PHAKPHISUT received the B.Eng. degree in telecommunication engineering, the M.Eng. degree in data storage technology, and the D.Eng. degree in electrical engineering from the King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, in 2009, 2011, and 2015, respectively. His research interests include channel coding and artificial intelligence for communication and storage systems.

PARAMOTE WARDKEIN received the M.E. and D.E. degrees in electrical engineering from the King Mongkut's Institute of Technology Ladkrabang (KMITL), Bangkok, Thailand, in 1990 and 1997, respectively. He is currently an Associate Professor with the Department of Telecommunications Engineering, Faculty of Engineering, KMITL. His current research interests include analog/digital communications, circuit design, system analysis, queuing theory, and signal processing.