Noise Recycling Based Multi-level Flash Memory

Gilli Horowitz Hadayo, Yuval Cassuto, and Alejandro Cohen
Faculty of Electrical and Computer Engineering, Technion — Institute of Technology, Haifa, Israel,
Emails: gilli.h@campus.technion.ac.il, ycassuto@ee.technion.ac.il, and alecohen@technion.ac.il

Abstract*—We propose a novel low-complexity Noise-Recycle-based Decoder (NRD) for Multi-Level Cells (MLC) to obtain high storage rates. Our proposed scheme utilizes Block Partition (BP) mapping in multi-level flash memory. Based on multi-stage decoding, NRD method decodes layers sequentially, starting from the MSB (layer 1) to improve noise robustness. Specifically, a digital noise realization is estimated utilizing already decoded layers. This estimated noise is then recycled by subtraction in the subsequent layers pre-decoding to improve Bit Error Rate (BER). Noise Recycling (NR) approach assumes simultaneous reading of an entire MLC, ensuring a fixed correlated noise realization for decoding all layers within a cell. For noise shifts across multiple representation levels, we establish a reliability bound and show via simulations that the proposed NRD solution outperforms Independent Decoding (ID) with both BP and Gray mappings without NR. For a single-level noise shift, we analytically and through simulations demonstrate that the proposed scheme outperforms the baseline ID scheme with BP mapping and no NR, while achieving equal performance to ID with Gray mapping and no NR. We introduce new capacity and reliability bounds for MLC NAND flash memory using BP mapping under single-level noise shifts.

I. Introduction

Flash memory is crucial for modern storage but struggles with rising error rates as *multi-level cell (MLC)* architectures handle greater data volumes. Unlike single-level cells storing one bit, MLCs store m bits per cell, and the stored value is referred to as a *multi-layered symbol (MLS)*. Higher density in cells reduces voltage margins, making MLCs more prone to noise-induced errors. Approaches to improve MLC reliability and storage rate have focused on strategies like: 1) Using error correction methods, such as Bose–Chaudhuri–Hocquenghem (BCH) and Low-Density Parity Check (LDPC) codes [1]–[3]. 2) Adjusting reading/writing voltages to mitigate noise [2], [3] and using more reading voltages with soft decoders for finer MLS estimation [1], [2], [4], [5]. 3) Optimizing verify operations and MLS mappings to reduce inter-cell noise caused by high voltages [5]–[7]. 4) Employing practical encoding strategies, such as independent encoding with low complexity progressive reading [1].

In this work, we introduce a new low-complexity decoding approach that exploits noise correlation between multiple bits of the same MLS. The decoder employs the concept of *noise recycling (NR)* [9], and is called *NR-based decoder (NRD)*. It works by efficiently estimating and correcting noise from previously decoded layers, as illustrated in Fig 1. This significantly enhances reliability and increases storage capacity. The proposed decoder works within the scheme of multi-level coding [8], which in this paper we call multi-*layer* coding. In particular, it uses the *block partition (BP)* mapping, in which the layers are ordered in decreasing significance (from MSB to LSB). The algorithm works purely in the digital domain, assuming readout of "hard" MLS values without any soft information. Therefore, its main task is estimation of digital noise and correction of MLS values between layer decodings. Thanks to the regularity of the BP mapping, the algorithm can work with general noise intensities, but we mostly discuss the special cases where the noise shifts an MLS by up to k levels (in any direction), where $k \in \{1, 2\}$.

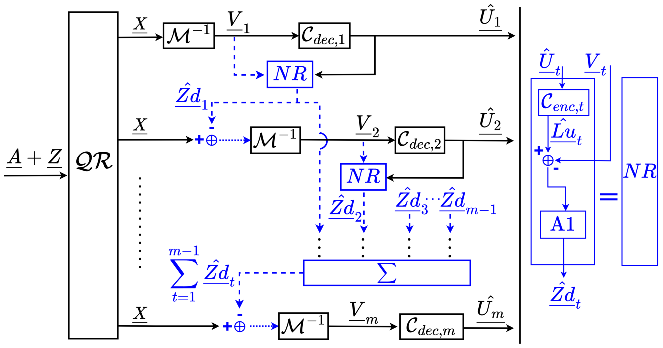

Fig. 1: Our decoding scheme is illustrated with solid black lines representing the ID baseline (see [8]) and dashed blue lines highlighting the components of the proposed Noise-Recycling-based scheme. The quantized reading of a string, denoted as $\mathcal{QR}$, yields noisy digital cell representation levels, denoted as $\underline{X} = (X_1, \ldots, X_N) \in \{1, \ldots, 2^m\}^N$. $\mathcal{M}(\cdot)$ represents the mapping function from the binary vector written to a cell to the cell levels, and $\mathcal{M}^{-1}(\cdot)$ its inverse. The decoding process for the t-th layer is denoted by $\mathcal{C}_{\text{dec},t}$. $\underline{V}_t$ is the input to the t-th decoder, representing the t-th codeword based on $\underline{X}$ and the NR corrections from previous layers. The decoded message from the t-th layer is $\underline{\hat{U}}_t$. After decoding layer t, the estimated digital noise $\underline{\hat{Z}d}_t$ is computed using $\underline{\hat{U}}_t$ and $\underline{V}_t$. This noise is subtracted from the quantized cell readings $\underline{X}$, along with noise estimates from prior layers, improving the BER for subsequent layers. Specifically, $\mathcal{C}_{\text{dec},t}$ operation is performed on the input $\underline{V}_t = \mathcal{M}^{-1}\left(\underline{X} - \sum_{l=1}^{t-1} \underline{\hat{Z}d}_l\right)$, where the estimated digital noise is calculated using the logical steps illustrated by $A1$ in the figure and detailed in the colored box of Algorithm 1.

The novelty of the algorithm is in providing a low-complexity method for multi-stage decoding of multiple bits in MLC flash. Relative to the original NR algorithm [9] that defined the noise correlation explicitly, it takes advantage of an implicit and more subtle noise correlation between the MLS bits. Compared to prior multi-layer coding schemes with multi-stage decoders such as [10], it does not require soft information or likelihood calculations for passing information between layers. Finally, it is also advantageous when comparing to a popular scheme using *Gray mapping* and *independent decoding (ID)* of each layer, thanks to its effectiveness beyond 1-shift noise intensities. In fact, we show analytically that under 1-shift noise intensity the effective layer error rates of NRD are identical to Gray with ID, which is a significant improvement from BP mapping without NRD that is much inferior to ID with Gray in that regime. More importantly, we show that in 2-shift intensities, NRD with BP is superior to ID with Gray: analytically by showing lower bit-error rates in the last layer, and empirically by showing superior block-error rate performance.

The structure of this paper is as follows. In Section II, we describe the system model and the problem formulation. In Section III, we present the proposed NRD algorithm, and in Section IV we provide the analytical results. Finally, we describe an experimental study exemplifying the performance of the proposed method in Section V.

II. System Model & Problem Formulation

A. Flash Memory System

In flash memory systems, a *string* is a series of N cells sharing a common bulk. An MLC with m layers is a transistor that can be programmed to any of 2^m distinct threshold voltage levels. A binary vector $(b_1, b_2, \ldots, b_m) \in \{0,1\}^m$ stored at an MLC is called a *multi-layered symbol (MLS)*. We will also refer to an MLS as the digital representation level of the cell's threshold voltage within $\{1, \ldots, 2^m\}$. This representation is based on a mapping, discussed later, in which b_1 (layer 1) is the most significant bit (MSB) and b_m (layer m) is the least significant bit (LSB).

An MLS value $i \in \{1, \ldots, 2^m\}$ is mapped to a real-valued cell voltage level A by a function $\mathcal{W}$, where $A = \mathcal{W}(i) \triangleq (i - 0.5) \cdot D$, and D is the equal voltage spacing between adjacent MLS representation voltages. A is also called the *write value*, as it directly represents the data written to the cell. The noisy cell voltage of an MLS is $A + Z$, where $Z \sim \mathcal{N}(0, \sigma^2)$ is a Gaussian random variable modeling the cumulative read/write noise affecting the cell [1], [2]. $A + Z$ is not read directly as a real number; instead, we define $2^m - 1$ sensing levels $V_{(RD,\hat{l})} \triangleq \hat{l} \cdot D$ where $\hat{l} \in \{1, \ldots, 2^m - 1\}$. The output digital value is 1 if $A + Z \leq V_{(RD,1)}$, 2^m if $V_{(RD,2^m-1)} < A + Z$, or the value $\hat{l}$ if $V_{(RD,\hat{l}-1)} < A + Z \leq V_{(RD,\hat{l})}$ for any $\hat{l} \geq 2$ (see Fig. 2.c). The quantized reading is achieved via a sense-amp comparator, producing a quantized threshold voltage.

The probability that the wrong MLS is read from a cell due to a noise shift will be called *read error probability* and denoted as P_{read}. The *read error probability of the t-th layer*, denoted as $P_{(\text{read},t)}$, is the probability that noise shifts the MLC threshold voltage level to represent an MLS differing in the t-th bit from the originally written value. As a model of system reliability, we assume that the noise variance is sufficiently small, such that with high probability an MLS value i is read as a value in $\{i - k, \ldots, i, \ldots, i + k\}$, for some integer k. We call this assumption the *k-shift noise limit*. An interesting case in practice is $k = 1$, in which shifts beyond 1 are considered negligible, but we also treat the case $k = 2$ applicable to less reliable high-density memories.

B. Multi-layer Coding Scheme

The encoding process follows the classic multi-level coding scheme (see [11, Section II.B] and [8]), which we rename here to multi-*layer* coding to avoid confusion with the cell voltage levels. In multi-layer coding each bit b_t of an MLS is part of a different length-N codeword, and these m codewords are encoded independently. Moreover, multi-layer coding allows performing a *multi-stage decoding* process, whereby the decoding outcomes of layer t are used to refine the inputs for the decoders of subsequent layers $> t$. We defer discussing the decoding to Section III, and define the encoder first. Let $\underline{U}_t \in \{0,1\}^{Nu_t}$ be the t-th layer's binary message of length $Nu_t \leq N$. For a message $\underline{U}_t$ with encoding rate $R_t = Nu_t/N$, we define the layer-t encoder function by $\mathcal{C}_{\text{enc},t}(\underline{U}_t, R_t)$; The encoder output, denoted as $\underline{Lu}_t \in \{0,1\}^N$, represents the codeword of the t-th layer. The j-th bit of this codeword is denoted as $Lu_{(t,j)}$. Thus, the MLS value of the j-th cell is $(b_1, b_2, \ldots, b_m) = \left(Lu_{(1,j)}, \ldots, Lu_{(m,j)}\right)$.

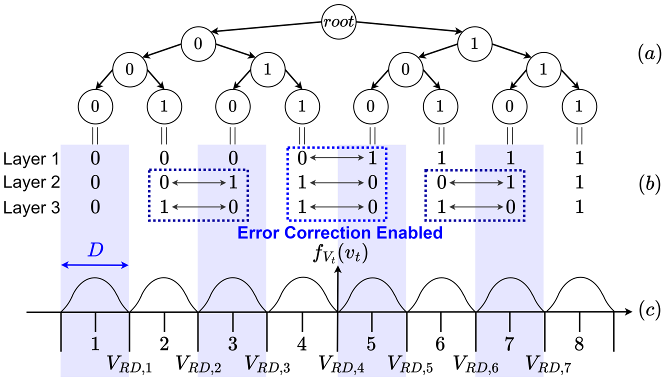

Fig. 2: Illustrated is a TLC (8-level cell) with BP mapping and the corresponding partitioning tree. The 2^3-ary version of the MLS values is enumerated from 1 to 8, and the binary version is displayed as the corresponding column vector positioned above the enumeration. Each MLS value is obtained by traversing the tree based on the already decoded prefix from the preceding layers. Errors that are found and subsequently corrected are framed in blue dashed lines. The two-sided arrows indicate bit flips between adjacent MLS values. The gap between MLS levels, D, is highlighted. The real-valued noise distribution Z within the cell follows a Gaussian distribution centered around each write level, resulting in the final threshold voltage distribution of $A + Z$ based on the voltage level A of the desired MLS. The read voltages $V_{(RD,\hat{l})}$ are also marked.

A key component in multi-layer coding is the *mapping* between the binary representation $(b_1, \ldots, b_m)$ of the MLS and its 2^m-ary digital value that is written to the cell. In this paper we define this mapping by the function $\mathcal{M}(\cdot)$ that maps $(b_1, \ldots, b_m)$ directly to a write value $a \in \{1, \ldots, 2^m\}$, which (according to Section II-A) is in turn mapped to a voltage level $A = (a - 0.5)D$ (see Fig. 2.b and 2.c). Following the encoding, the j-th cell's digital write value is thus $a_j = \mathcal{M}\left(Lu_{(1,j)}, \ldots, Lu_{(m,j)}\right)$. The inverse of this mapping function is denoted as $\mathcal{M}^{-1}(\cdot)$. With a slight abuse of notation, if $a = \mathcal{M}(b_1, \ldots, b_m)$, we write $\mathcal{M}_t^{-1}(a) = b_t$.

C. Problem formulation

Following the multi-layer encoding and mapping of the previous sub-section, each write value A_j is corrupted by a noise value Z_j and the result is mapped to a read value $X_j \in$

$\{1,\ldots,2^m\}$ using 2^m-1 sensing levels (see Section II-A). For decoding, each read value X_j is mapped back to binary: $(Y_{(1,j)},\ldots,Y_{(m,j)}) = \mathcal{M}^{-1}(X_j)$, where $Y_{(t,j)}$ is the read bit at layer t of cell j. Thus, the full binary read word of layer t, without any noise corrections applied, is denoted $\underline{Y}_t$. A decoder in this setting is a function $\mathcal{C}_{dec}$ that gets as input $(\underline{Y}_1,\ldots,\underline{Y}_m)$ (or equivalently $(X_1,\ldots,X_N)$), and outputs estimates of the m information messages. For layer t the estimated codeword is denoted $\underline{\hat{Lu}_t}$. The performance of $\mathcal{C}_{dec}$ is measured by two probabilities: the *system error probability*: $P_{\text{err}} \triangleq P(\exists t \in [m] : \underline{\hat{Lu}_t} \neq \underline{Lu}_t)$ and the *error probability of the t-th layer* $P_{(\text{err},t)} \triangleq P(\underline{\hat{Lu}_t} \neq \underline{Lu}_t)$.

Our goal is to design a novel decoding scheme that enhances noise robustness in multi-level flash memory with minimal computational complexity. That is, by efficiently exploiting the noise corelation within the cells, we aim to minimize P_{err} and $P_{(\text{err},t)}$ to increase overall decoding reliability or increase the total storage rate $R=\sum_{t=1}^{m} R_t$.

III. Proposed Noise-Recycling (NR) Decoder

We now present our proposed algorithm, NRD, for the multi-layer decoding problem defined in Section II-C. Before that, we note a common alternative solution of using a Gray code [1], [2], [10] for the mapping function $\mathcal{M}(\cdot)$, and performing independent decoding (ID) in each layer. In our solution, we employ a powerful multi-stage decoding process, and choose the mapping to allow effective transfer of decoding outcomes to subsequent layers. For the mapping function $\mathcal{M}(\cdot)$ we choose the standard binary mapping, which maps binary vectors to MLS values $1,\ldots,2^m$ in increasing order where b_1 is taken as the MSB and b_m as the LSB (See Fig. 2.b). In multi-layer coding literature, this mapping is called *Block Partitioning (BP)* [8], also referred to by other names in modulation and coding contexts.

Our proposed decoder uses as building blocks binary decoding functions for the codes used in the m layers; for layer t we denote this function by $\mathcal{C}_{\text{dec},t}(\underline{V}_t, R_t)$, where $\underline{V}_t$ is a binary length-N word, possibly corrupted by errors. These decoders are simple hard-decision decoders providing at the output an estimated binary message.

The full formal description of the proposed decoder is given in Algorithm 1 and is illustrated in Fig. 1. For clarity, we also describe its main ideas in text. In the following sections, we present key results, outline essential operations and analyze NRD performance under the noise-limit model defined in Section II-A.

Algorithm 1 Noise Recycling Decoding for MLC with BP[*]

```
 1: Read data from string
 2: $\underline{a}^{(1)} = (a_1,\ldots,a_N) := \underline{X} \in \{1,\ldots,2^m\}^N$
 3: Decode layer by layer
 4: Init: $S_j = 0, \quad \forall j \in \{1,\ldots,N\}$
 5: for $t$ from 1 to $m-1$ do
 6:     $\underline{\hat{U}}_t = \mathcal{C}_{\text{dec},t}(\ \mathcal{M}_t^{-1}\left(\underline{X} - \sum_{l=1}^{t-1} \underline{\hat{Zd}_l}\right),\ R_t)$    $\Leftarrow$ Apply NR
 7:     Re-encode the estimated message
 8:     $\underline{\hat{Lu}_t} = \mathcal{C}_{\text{enc},t}(\underline{\hat{U}}_t\ ,\ R_t)$
 9:     For each cell, calculate $\hat{Zd}_{(t,j)}$ for NR in next layer
10:     for $j$ from 1 to $N$ do
11:         if $\hat{Lu}_{(t,j)} - \mathcal{M}_t^{-1}\left(X_j - \sum_{l=1}^{t-1} \hat{Zd}_{(l,j)}\right) < 0$ then
12:         NR Check 1: Positive shift → Negative correction
13:             $a_j^{(t+1)} = S_j + 2^{m-t}$
14:         else if $\hat{Lu}_{(t,j)} - \mathcal{M}_t^{-1}\left(X_j - \sum_{l=1}^{t-1} \hat{Zd}_{(l,j)}\right) > 0$ then
15:         NR Check 2: Negative shift → Positive correction
16:             $a_j^{(t+1)} = S_j + 2^{m-t} + 1$
17:         else
18:         NR Check 3: No shift detected → No correction
19:             $a_j^{(t+1)} = a_j^{(t)}$
20:         end if
21:         Update subset shift and $\hat{Zd}_{(t,j)}$
22:         $S_j += \left(2^{m-t}\right) \cdot \hat{Lu}_{(t,j)}$
23:         $\hat{Zd}_{(t,j)} = a_j^{(t)} - a_j^{(t+1)}$ $\Leftarrow$ NR Correction for next layer
24:     end for
25: end for
26: Decode last layer
27: $\underline{\hat{U}}_m = \mathcal{C}_{\text{dec},m}(\ \mathcal{M}_t^{-1}\left(\underline{X} - \sum_{l=1}^{m-1} \underline{\hat{Zd}_l}\right),\ R_m)$    $\Leftarrow$ Apply NR
28: return $\underline{\hat{\mathbf{U}}} = (\underline{\hat{U}}_1, \underline{\hat{U}}_2, \ldots, \underline{\hat{U}}_m)$
```

[*] NRD operates in the discrete domain. i.e., without any soft information.

A. Algorithm Description

The input to Algorithm 1, $\underline{X}$, is assigned to a vector variable $\underline{a}^{(1)}$, where $a_j^{(t)}$ is the estimated MLS value of cell j before decoding layer t. The main idea of the algorithm is that after decoding each layer t, the digital noise $\underline{\hat{Zd}_t}$ is estimated based on the decodings of layers 1 to t. Since it is the same noise sample Z_j that affects the bits of all layers in the j-th cell, a noise estimate from decoding outcome of one layer can be recycled for subsequent layers[1]. The sum of noise estimates from layers $1,\ldots,t$ is in turn subtracted from the input $\underline{X}$ (line 6), before decoding the $(t+1)$-th layer. This reduces the BER of these layers. The noise estimation is done as follows: when a layer-t bit is corrected from 1 to 0 (line 11), we infer that the noise shifted the MLS level to the right, and thus the noise estimate is the distance to the highest representation level with $b_t = 0$ and prefix $(b_1,\ldots,b_{t-1}) = (\hat{Lu}_{(1,j)},\ldots,\hat{Lu}_{(t-1,j)})$. This level is $S_j + 2^{m-t}$ (line 13). S_j is a variable that holds the integer value of the prefix: $\mathcal{M}(\hat{Lu}_{(1,j)},\ldots,\hat{Lu}_{(t-1,j)},0,\ldots,0)$. Meaning, it serves as the path to the root in the BP mapping tree (see Fig. 2.a, [8]), and is updated after decoding each layer. Similarly, when the correction is from 0 to 1 (line 14), we infer that the noise shifted to the left, and thus the noise estimate is the distance to the lowest level with $b_t = 1$ and the same prefix. This level is $S_j + 2^{m-t} + 1$ (line 16). That is, assuming the message $\underline{\hat{U}}_t$ is decoded correctly, any difference between $\mathcal{M}^{-1}\left(\underline{X} - \sum_{l=1}^{t-1} \underline{\hat{Zd}_l}\right)$ and $\underline{\hat{Lu}_t} = \mathcal{C}_{\text{enc},t}(\underline{\hat{U}}_t\ ,\ R_t)$ identifies noisy cells. This difference, used in the marked box of Algorithm 1, provides the *indexes* of noisy cells and the *direction* of the required correction for the next layer. Correctness proofs for the proposed NR algorithm, due to the space limitation, are given in [11, Section IX].

The proposed NR scheme based on BP mapping can handle noise shifts between any MLS values. Assuming correct decoding of previous layers, this process improves BER. For example, let us consider a noise shift under 1-shift noise limit in a TLC with BP mapping (illustrated in Fig. 2). If the original write value was 4, but the noise shifts it to 5 (MLC values differing in b_1, and all other bits), the shift is

[1] Note, in MLC, the analog noise estimation is given by $\underline{\hat{Z}_t} = \underline{\hat{Zd}_t} \cdot D$.

detected after decoding the first layer (see Fig. 2.b). Using NRD, the shift is corrected, preventing errors in subsequent layers. For an example of noise correction under 2-shift noise limit refer to [11, Section V.A]. Algorithm 1 involves only re-encoding and basic arithmetic, similar to [9] when the channel correlation is fixed and given. Since encoder complexity for linear codes is typically much lower than that of the decoder, the added complexity of NRD is negligible. For a full complexity analysis, see [11, Section IV.C].

1) Simplified NRD (assuming 1-shift noise limit): Under the 1-shift noise limit, we can simplify Algorithm 1. Since we know that an error results in a shift of exactly 1 level, instead of a full noise estimate $\underline{\hat{Zd}}_t$, we may implement a ± 1 correction for a corrected bit, and such a correction can happen in at most one layer in every decoding instance. That is, we can skip all the lines in the shaded box and substitute directly $\underline{\hat{Zd}}_t = \mathcal{M}_t^{-1}\left(\underline{X} - \sum_{l=1}^{t-1} \underline{\hat{Zd}}_l\right) - \underline{\hat{Lu}}_t$.

IV. Analytical Results

We will now focus on analyzing our proposed NRD scheme. Although our scheme works under any Signal-to-Noise Ratio (SNR), the number of possible errors to analyze increases exponentially with the possible MLS shifts that can be caused by noise. Therefore, we will only provide bounds under k-shift noise limits for $k \in \{1, 2\}$. The proofs for the bounds in Theorems 1-7 given in this section are deferred to an extended version [11], due to lack of space.

For a *k-shift noise limit*, we assume that the probability of a shift of exactly k levels equals the tail probability of shifting $\geq k$ levels. Given the Gaussian noise distribution with standard $Q(l,h)$ function,[2] when $k = 1$ let $\xi^- = Q(-\infty, -D/2)$ denote the probability of a leftward shift causing a reading error. By symmetry, the probability of a rightward shift is identical ($\xi^+ = \xi^- \triangleq \xi$), making the total read error probability 2ξ. For $k = 2$ we define $\xi_1^- = Q(-3D/2, -D/2)$ to be the probability of a left shift of one level, and from symmetry $\xi_1^+ = \xi_1^- \triangleq \xi_1$ for the right shift. Similarly, let $\xi_2^- = Q(-\infty, -3D/2)$ denote the probability of a two-level shift to the left, and from symmetry $\xi_2^+ = \xi_2^- \triangleq \xi_2$ for the right shift. Thus, the total probability of a read error is $2(\xi_1 + \xi_2)$, and we note that $\xi = \xi_1 + \xi_2$.

Let $\mathcal{C}_{\text{enc},t}(\cdot, R_t)$ represent a non-random structured code—such as LDPC [1], [2], BCH [2], or Hamming [5], characterized by a fixed, equal error probability for each codeword.[3] This code is assumed to be at least d_t*-error-correcting code*, meaning it guarantees to correct at least up to d_t bit errors in a codeword of length N. Thus, similar to [10] and [12], the error probability is upper bounded by

$$P_{\text{err}} \leq \sum_{t=1}^{m} P_{(\text{err},t)} \leq \sum_{t=1}^{m} 1 - \sum_{k=0}^{d_t} \binom{N}{k} (1 - P_{(\text{read},t)})^{N-k} P_{(\text{read},t)}^{k}. \tag{1}$$

1) Analytical Results Under 1-shift noise limit: The achievability results for ID and NRD are given by the following theorems.

Theorem 1. Consider an ID scheme with BP mapping under 1-shift noise, the t-th layer read error probability determining the upper bound on P_{err} in (1) is

$$P_{(\text{read},t)} = \frac{\xi \cdot 2^{t-1}}{2^{m-1}} + \frac{\xi \cdot (2^{t-1} - 1)}{2^{m-1}} \triangleq P_{(\text{read},t)}^{\text{ID (BP)}}.$$

Theorem 2. Consider an NRD scheme with BP mapping under 1-shift noise, the t-th layer read error probability determining the upper bound on P_{err} in (1) is

$$P_{(\text{read},t)} = \frac{\xi \cdot 2^{t-1}}{2^{m-1}} \triangleq P_{(\text{read},t)}^{\text{NRD (BP)}}.$$

Theorem 2 can be interpreted as a read error probability reduction, compared to Theorem 1. In MLC, each reading voltage for layer t distinguishes two MLS levels that differ in the t-th bit. With BP mapping, once levels differ in the t-th bit, they continue to differ in all subsequent bits. For ID with BP mapping, no noise correction is applied, so such a noise shift affects $P_{(\text{read},\tau)}$ for all $\tau \geq t$. However, for NRD with BP mapping, the error is found and corrected in layer t. As a result, the noise shift impacts $P_{(\text{read},\tau)}$ only for $\tau = t$. This reduces the read error probability per layer.

Corollary 1. Consider a scheme with BP mapping under 1-shift noise. The upper bound on the system error probability is strictly smaller when using NRD rather than ID.

The resulting capacity bounds are the following. The first result is a direct consequence of [8], while the next one is our main analytical result for NRD with BP mapping.

Theorem 3. Consider an ID scheme with BP mapping under 1-shift noise. The capacity of the t-th layer is

$$C^t = 1 - H_2\left(P_{(\text{read},t)}^{\text{ID (BP)}}\right).$$

The proof of Theorem 3 is obtained using similar techniques for parallel channels in [8] with the derived $P_{(\text{read},t)}^{\text{ID (BP)}}$ in Theorem 1. The complete proof is in [11, Appendix D].

Theorem 4. Consider an NRD scheme with BP mapping under 1-shift noise. The capacity of the t-th layer is

$$C^t = 1 - H_2\left(P_{(\text{read},t)}^{\text{NRD (BP)}}\right).$$

Our main analytical analysis for the capacity proof in Theorem 4 can be interpreted as a generalization of analysis for parallel channels (e.g., as in [8] and Theorem 3) to correlated channels and channels with feedback [15, Chapts. 9.4-9.6], using the derived $P_{(\text{read},t)}^{\text{NRD (BP)}}$ in Theorem 2. The complete proof is in [11, Section VIII.A]. Now, given that $C^{\text{Tot}} = \sum_{t=1}^{m} C^t$, from Theorems 1-4 we obtain the following corollary.

Corollary 2. Consider a scheme with BP mapping under 1-shift noise limit. The system capacity with NRD is strictly greater than that achieved with ID.

Proof Sketch:, In the interval $[0, \frac{1}{2}]$, the binary entropy function is monotonically increasing. Hence, under 1-shift noise limit, i.e., for $\xi \leq \frac{1}{4}$ (see [11, Section VIII.B]), $C^{t,\text{NRD (BP)}} \geq C^{t,\text{ID (BP)}}$. Equality holds only when $t = 1$, as in this case the error probabilities are identical. For all

[2] The standard function is given by $Q(l,h) \triangleq \int_l^h \frac{1}{\sqrt{2\pi\sigma_n^2}} e^{-\frac{z^2}{2\sigma_n^2}}\, dz$.

[3] If codewords have unequal error probabilities, the worst-case distribution can serve as an upper bound for error probability analysis.

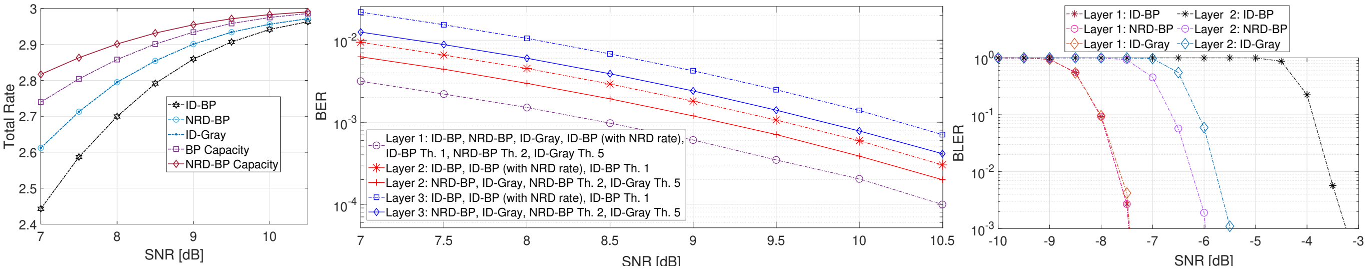

Fig. 3: Performance evaluation. Capacity (left) and Achievability (middle) under 1-Shift-Noise in a TLC under a Gaussian noise [13], and Achievability (right) under 2-Shift-Noise in a DLC under Student's-T noise, which mimics Gaussian noise but with more outliers [14].

subsequent layers ($t > 1$), the inequality becomes strict, highlighting the advantage of NR-based decoding. □

Now, we compare NRD with BP mapping to ID with Gray mapping in a flash memory system.

Theorem 5. Consider an ID scheme with Gray mapping under 1-shift noise, the upper bound on P_{err} in (1) is equal to that of an NRD scheme with BP mapping.

Note, since all read voltages are utilized during the reading process, both mapping methods require the same quantity and values of read voltages to read an entire MLS. Each read voltage, necessary for layer t distinguishes neighboring MLS levels that differ in the t-th bit. Thus, BP and Gray mappings offer equivalent potential capabilities to find errors. However, BP mapping (without NRD) is less noise robust than Gray mapping, and gains from additional post-decoding error correction provided by NR for 1-shift noise. Theorem 5 shows that for $k = 1$ BP mapping with NR performs equally to Gray mapping, while BP mapping without NR performs worse (analysis for $k > 1$ presented next).

Remark 1. It is important to note that if the 1-shift noise limit is violated, i.e., noise shifts across multiple representation levels, both mappings (BP and Gray) can gain from post-decoding error correction, highlighting the value of our NR-based scheme for large noises as shown in the next section.

2) Analytical Results Under 2-shift noise limit: The achievability is given by the following theorems.

Theorem 6. Consider an ID scheme with Gray mapping under 2-shift noise, the upper bound on error probability P_{err} in (1) is given by the read error probability of the t-th layer

$$P_{(\text{read},t)} = \begin{cases} \frac{(\xi_1+2\xi_2)\cdot 2^{t-1}}{2^{m-1}} \triangleq P^{\text{ID (Gray)}}_{(\text{read},t)} & ,\text{if } t < m, \\ \xi_1 + \xi_2\left(2 - \frac{1}{2^{m-1}}\right) \triangleq P^{\text{ID (Gray)}}_{(\text{read},m)} & ,\text{if } t = m. \end{cases}$$

A straightforward algebraic comparison of Theorems 5 and 6 shows a higher read error probability in the t-th layer due to increased noise.

Theorem 7. Consider an NRD scheme with BP mapping under 2-shift noise, the upper bound on error probability P_{err} in (1) is given by the read error probability of the t-th layer

$$P_{(\text{read},t)} = \begin{cases} P^{\text{ID (Gray)}}_{(\text{read},t)} & ,\text{if } t < m, \\ \xi_1 + \xi_2 \triangleq P^{\text{NRD (BP)}}_{(\text{read},m)} & ,\text{if } t = m. \end{cases}$$

As with 1-shift noise limit, the upper bounds on error probability for layers $t < m$ are the same between ID with Gray mapping and NRD with BP mapping. However, NRD with BP mapping shows better performance in the last, most noise-sensitive layer. Ongoing work focuses on NR improvements under k-shift noise for $k \geq 3$.

V. Performance Evaluation

Here, we evaluate the proposed scheme by simulation results in Matlab. We focus on analyzing TLCs ($m = 3$) and DLCs ($m = 2$), though our approach extends to any m. In Fig. 3, we present the performance evaluation for the following simulations, when comparing to *binary reflected Gray mapping*, as done in [2], [16].

Simulation for Capacity and Achievability Under 1-Shift Noise: In this simulation we use the simplified version of NRD, presented in Section III-A1. For ease of comparison, we use BCH codes, which are suitable for flash memory systems and compatible with ID schemes [2], [10]. Their computable minimum distance simplifies comparison to theoretical bounds[4].

Both the left and middle plots in Fig. 3 present results for a fixed *block-error rate (BLER)* of $P_{(\text{err},t)} = 0.001$ across all layers in a TLC. The left plot shows the rate as a function of SNR, where NRD with BP (dark blue) outperforms ID with BP (black) and matches ID with Gray (light blue). Both BP-based schemes fall short of capacity (NRD in red, ID in purple), likely due to BCH code limitations, though NRD's rate advantage remains evident. The middle plot evaluates the BER for each layer, showing alignment of all schemes with their predicted $P_{(\text{read},t)}$, derived from Theorems 1, 2, and 5. These results highlight the performance benefits of the NR-based scheme with BP mapping, achieving parity with ID using Gray mapping.

Simulation for Performance Under 2-Shift-Noise: In this simulation NRD is implemented based on Algorithm 1. This simulation models a strained MLC, achieved either by increasing the number of levels (m) or reducing the MLS margins (D). For simplicity, we opted for the latter. To evaluate performance under significant noise, we introduced Student's-T distributed noise, increasing large noise events, and used LDPC coding. For fixed rates $R_1 = 0.25$ and $R_2 = 0.15$ in a DLC, as presented in the right plot in Fig. 3, NRD with BP (circles) achieves a lower BLER compared to ID with BP (line with asterisks) and ID with Gray (diamonds), confirming our theoretical results.

[4] Our method extends beyond BCH codes and is applicable to other, potentially stronger, structured codes, paving the way for future research.

References

.

[1] N. Wong, E. Liang, H. Wang, S. V. Ranganathan, and R. D. Wesel, "Decoding flash memory with progressive reads and independent vs. joint encoding of bits in a cell," in *2019 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 2019, pp. 1–6.

[2] 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 Journal on Selected Areas in Communications*, vol. 32, no. 5, pp. 880–891, 2014.

[3] A. Asmani, S. Galijasevic, and R. D. Wesel, "Write voltage optimization to increase flash lifetime in a two-variance Gaussian channel," in *2024 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2024, pp. 1143–1148.

[4] J. Wang, T. Courtade, H. Shankar, and R. D. Wesel, "Soft information for LDPC decoding in flash: Mutual-information optimized quantization," in *2011 IEEE Global Telecommunications Conference-GLOBECOM 2011*. IEEE, 2011, pp. 1–6.

[5] K. Mizrachi, I. Bloom, and Y. Cassuto, "Memory reliability for cells with strong bit-coupling interference," in *Proceedings of the International Symposium on Memory Systems*, 2017, pp. 196–204.

[6] Y. Kim, J. Kim, J. J. Kong, B. K Vijaya Kumar, and X. Li, "Verify level control criteria for multi-level cell flash memories and their applications," *EURASIP Journal on Advances in Signal Processing*, vol. 2012, pp. 1–13, 2012.

[7] G. Hemink and A. Goda, "NAND Flash technology status and perspectives," *Semiconductor Memories and Systems*, pp. 119–158, 2022.

[8] U. Wachsmann, R. Fischer, and J. Huber, "Multilevel codes: theoretical concepts and practical design rules," *IEEE Transactions on Information Theory*, vol. 45, no. 5, pp. 1361–1391, 1999.

[9] A. Cohen, A. Solomon, K. R. Duffy, and M. Médard, "Noise recycling," in *2020 IEEE International Symposium on Information Theory (ISIT)*. IEEE, 2020, pp. 315–320.

[10] H. Imai and S. Hirakawa, "A new multilevel coding method using error-correcting codes," *IEEE Transactions on Information Theory*, vol. 23, no. 3, pp. 371–377, 1977.

[11] G. Horowitz Hadayo, Y. Cassuto, and A. Cohen, "Noise Recycling Based Multi-level Flash Memory," *arXiv preprint*.

[12] A. Solomon and Y. Cassuto, "Error-correcting WOM codes: Concatenation and joint design," *IEEE Transactions on Information Theory*, vol. 65, no. 9, pp. 5529–5546, 2019.

[13] Wikipedia contributors, "Normal distribution — Wikipedia, the free encyclopedia," 2024, [Online; accessed December 2024]. [Online]. Available: https://en.wikipedia.org/wiki/Normal_distribution

[14] ——, "Student's t-distribution — Wikipedia, the free encyclopedia," 2024, [Online; accessed December 2024]. [Online]. Available: https://en.wikipedia.org/wiki/Student%27s_t-distribution

[15] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. New-York: Wiley, 2006.

[16] J.-P. Thiers, D. N. Bailon, and J. Freudenberger, "Bit-labeling and page capacities of TLC non-volatile flash memories," in *2020 IEEE 10th International Conference on Consumer Electronics (ICCE-Berlin)*. IEEE, 2020, pp. 1–6.