Improving LDPC Decoding Performance for 3D TLC NAND Flash by LLR Optimization Scheme for Hard and Soft Decision

LANLAN CUI and FEI WU, Huazhong University of Science and Technology, China
XIAOJIAN LIU, DERA Co., Ltd, China
MENG ZHANG, RENZHI XIAO, and CHANGSHENG XIE, Huazhong University of Science and Technology, China

Low-density parity-check (LDPC) codes have been widely adopted in NAND flash in recent years to enhance data reliability. There are two types of decoding, hard-decision and soft-decision decoding. However, for the two types, their error correction capability degrades due to inaccurate **log-likelihood ratio (LLR)**. To improve the LLR accuracy of LDPC decoding, this article proposes LLR optimization schemes, which can be utilized for both hard-decision and soft-decision decoding. First, we build a threshold voltage distribution model for 3D **floating gate (FG) triple level cell (TLC)** NAND flash. Then, by exploiting the model, we introduce a scheme to quantize LLR during hard-decision and soft-decision decoding. And by amplifying a portion of small LLRs, which is essential in the layer min-sum decoder, more precise LLR can be obtained. For hard-decision decoding, the proposed new modes can significantly improve the decoder's error correction capability compared with traditional solutions. Soft-decision decoding starts when hard-decision decoding fails. For this part, we study the influence of the reference voltage arrangement of LLR calculation and apply the quantization scheme. The simulation shows that the proposed approach can **reduce frame error rate (FER)** for several orders of magnitude.

CCS Concepts: • **Mathematics of computing** → **Information theory**; **Coding theory**; • **Information systems** → **Information storage technologies**; **Storage class memory**;

Additional Key Words and Phrases: LLR, LDPC, TLC NAND flash, threshold voltage distribution, reference voltage

5

A preliminary version of this work was published as "VaLLR: Threshold Voltage Distribution Aware LLR Optimization to Improve LDPC Decoding Performance for 3D TLC NAND Flash" [Cui et al. 2019] in IEEE international Conference on Computer Design (ICCD'19).
This work was supported in part by Key Area Research and Development Program of Guangdong Province No. 2019B010107001, in part by the NSFC under Grant No. 61821003, No. 61872413, No. U1709220, No. 61902137, in part by National Key Research and Development Program of China No. 2018YFB1003305, No. 2018YFA0701800, in part by the 111 Project (No. B07038), in part by the China Postdoctoral Science Foundation No. 2019M66262, in part by the Postdoctoral Innovative Talents Support Program No. BX20190128, in part by the Excellent Projects for Postdoctoral Science and Technology Activities in Hubei Province, in partby the Key Project of Shandong Wisdom Joint Fund No. ZR2019LZH009.
Authors' addresses: L. Cui, F. Wu (corresponding author), M. Zhang, R. Xiao and C. Xie, Huazhong University of Science and Technology, Wuhan National Laboratory for Optoelectronics, 1037 Luoyu Road, Wuhan, 430074, China; emails: {cuilanlan, wufei, zgmeng, rzxiao, cs_xie}@hust.edu.cn; X. Liu, DERA Co., Ltd, 968 Jinzhong Road, Shanghai, 200050, China; email: liuxiaojian@derastorage.com.

1084-4309/2021/09-ART5 $15.00
https://doi.org/10.1145/3473305

ACM Reference format:
Lanlan Cui, Fei Wu, Xiaojian Liu, Meng Zhang, Renzhi Xiao, and Changsheng Xie. 2021. Improving LDPC Decoding Performance for 3D TLC NAND Flash by LLR Optimization Scheme for Hard and Soft Decision. *ACM Trans. Des. Autom. Electron. Syst.* 27, 1, Article 5 (September 2021), 20 pages.
https://doi.org/10.1145/3473305

1 INTRODUCTION

NAND flash memory is widely used in a variety of computer storage systems as non-volatile storage devices, with advantages of high random read and write performance, low bit cost and large capacity [8, 16]. As the industry continues pushing the technology scaling envelope, 3D **triple-level cell (TLC)** NAND flash has been commercialized as increased bit-density [35]. However, data reliability reduces due to high **raw bit error rates (RBER)** from the continual program/erase (P/E) operations and data retention errors [34]. For ensuring data reliability, **low-density parity-check (LDPC)** [12] with strong error correction capability has attracted a great deal of attention, becoming a more popular **error correction code (ECC)** for 3D TLC NAND flash.

There are two types of LDPC decoding: hard-decision and soft-decision decoding. Hard-decision provides high throughput but coarse **log-likelihood ratio (LLR)** precision, its error correction capability is limited. While soft-decision supports a more complicated decoding algorithm but requires much more read time. When the former failed, the latter is employed. The two schemes are complementary to meet the actual demand in 3D **floating gate (FG)** TLC NAND flash. For both hard-decision and soft-decision decoding, inaccurate LLRs result in poor error-correction capability, thus influencing data reliability.

To improve the error-correction capability, some previous works proposed solutions. Chen et al. [7] developed a non-uniform level placement strategy based on the **multi-Level Cell (MLC)** NAND flash error model to optimize the read reference voltages and improve decoding performance. Li et al. [22] proposed a smart sensing level placement scheme to reduce the LDPC decoding latency for MLC NAND flash. Ho et al. [15] dynamically applied soft-decision voltages to reduce the bit error rate and soft-decision decoding delay according to the shift of threshold voltage. However, for soft-decision decoding, above works focus on researching various interference which is different based on our measured data and the threshold voltage is still symmetrical. For hard-decision decoding, unlike [19, 27], which focus on complex Bit Flipping algorithm, this article improves the simplest hard-decision decoding significantly.

In this paper, we first establish a threshold voltage distribution model with Gaussian distribution of 3D FG TLC NAND flash. Through testing recently released 64-layer 3D FG TLC NAND flash chips, we find that the threshold voltage distributions are fitted to the Gaussian distributions quite well. Besides, the standard deviations of threshold voltage distribution among TLC adjacent states (excluding erase state) are quite different. Another feature of the 3D FG TLC NAND flash is that no serious asymmetric distributions of threshold voltages are noticed. Based on these observations, we choose the Gaussian distribution with fine-tuned mean and standard deviation to describe the states of 3D FG TLC NAND flash. The latest **solid-state drives (SSDs)** use their characteristics to optimize the LLR table, but when the real NAND flash characteristics cannot be obtained or the access cost is high, the LLR table cannot be optimized, or the performing optimization is expensive. Therefore, this article proposes the indirect method of constructing LLR tables. When the above conditions occur, LLR tables can be calculated to meet the needs.

In the case of hard-decision decoding, the cell threshold voltage is determined by comparing a serial of hard-decision reference voltages (HDRV) successively. Then, the corresponding LLR sequence is acquired. In conventional solutions, hard-decision decoding only provides the single precision LLR to adapt the Bose Chaudhuri Hocquenghem (BCH) decoder [2, 10]. The LDPC de-

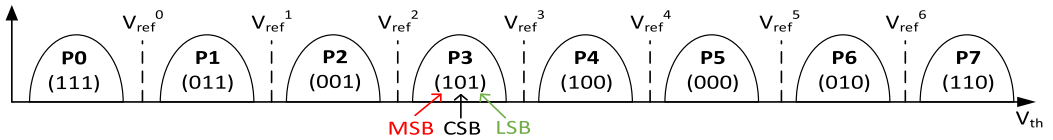

Fig. 1. Threshold voltage description of 3D TLC NAND flash.

coder is a maximum a posteriori (MAP) decoder [1], which is sensitive to the precision of input LLR. To fully develop the potential of LDPC decoder, we introduce two new schemes to get accurate LLRs and build a lookup table for different reference voltage regions and bit positions. Simulation results show a significant improvement for error correction capability of LDPC decoding with our approach.

For soft-decision decoding, more sensing levels are applied around each HDRV, so more accurate LLR can be gained from multiple reads. The arrangement of the soft-decision reference voltage (SDRV) is based on the threshold voltage distribution. In this work, we first decide the boundaries of SDRV, which are the outermost voltages around each HDRV. Since the distortion of threshold voltage distributions in 3D FG TLC NAND flash is within the allowable range of error in our test, symmetrically locating boundaries does not harm the decoding performance. Second, we find that after the SDRVs around each HDRV reach two to four, the error rate stabilizes. The above conclusions are the result of our experiments. Finally, the improved quantization scheme is applied to improve decoding performance.

The major contributions of this article are as follows:

- We fit the threshold voltage distribution based on the measured data. The results show that the fitting of the Gaussian distribution is feasible;
- We introduce a quantization mechanism that can be used for hard-decision and soft-decision;
- For the hard-decision, two novel implementation schemes are proposed and compared the traditional way;
- For the soft-decision, a suitable voltage configuration scheme is explained;
- This article conducts the simulation experiment to verify the effectiveness of the proposed schemes.

The rest of the article is organized as follows: Section 2 describes the background and related work. Section 3 introduces the establishment of the threshold voltage model. Topics about hard-decision and soft-decision decoding are discussed in Sections 4 and 5. Section 6 shows the simulation results. Section 7 concludes this article.

2 BACKGROUND AND RELATED WORK

In this section, we introduce the background of 3D TLC NAND flash memory, including sensing techniques, LLR calculation, and decoding algorithm. Then the related work is described.

2.1 3D TLC NAND Flash

Each 3D TLC NAND flash cell stores three bits of information with eight states, represented by P0, P1, P2, P3, P4, P5, P6, and P7, which are assigned to voltage windows by the read reference voltages V_{ref}, $(i = 0, 1, \ldots, 6)$ [3]. Three bits are mapped into a symbol, with the first bit representing the **most significant bit (MSB)**, the second bit representing the **center significant bit (CSB)**, while the last one representing the **least significant bit (LSB)**, as illustrated in Figure 1.

Table 1. The Gray Mapping of Eight States for TLC

State	P0	P1	P2	P3	P4	P5	P6	P7
MSB	1	0	0	1	1	0	0	1
CSB	1	1	0	0	0	0	1	1
LSB	1	1	1	1	0	0	0	0

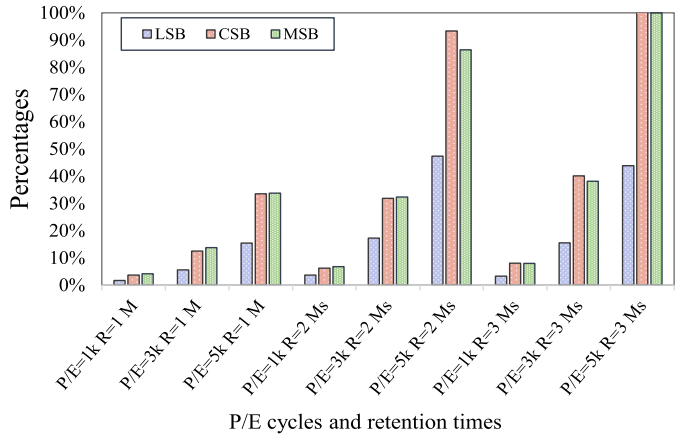

Fig. 2. The RBER comparisons among MSB, CSB, and LSB of 3D TLC NAND flash.

In this work, Gray mapping is adopted to 3D TLC NAND flash. The reason for using Gray mapping is that, when the data is switched between adjacent bits, only one bit changes, which greatly reduces the possibility of errors during state transitions. The mapping relationship we use is shown in Table 1.

The reliability of the three positions varies widely. The reliability of LSB is the highest, while the MSB reliability is the lowest. The errors of LSB are mainly caused by the state transition between P3 and P4. The errors of CSB are caused by the state transition around V_{ref}^{1} and V_{ref}^{5}, while the errors of MSB can be caused by the state transition around V_{ref}^{0}, V_{ref}^{2}, V_{ref}^{4} , and V_{ref}^{6}. Additionally, the distance between a state and a read reference voltage also determines the reliability of bits. The reliability of LSB in P7 (it is far from V_{ref}^{3} read reference voltage) is much higher than that in P4 (it is close in V_{ref}^{3}). In addition, after various disturbances occur, the reliability of the three pages further changes. We mainly consider two causes of errors, P/E cycles and retention. We test the RBER of the LSB, CSB, and MSB under different P/E cycles and retention time. The test is conducted on the hardware platform of the Flash Sorting test board, which is connected with the host through the PCIe interface to perform program, read, and erase operations. The test time can be reduced by baking the chip, where the data is written at a high temperature. For our chip, according to Arrhenius Law [5], placing the chip with data stored at a high temperature of 85°C for 13 hours is equivalent to one month at a normal temperature of 25°C. The RBER is obtained by writing random data and observing errors. Figure 2 gives some representative results. In the figure, "P/E" is the P/E cycle, "P/E = 1k" refers to 1,000 cycles. "R" represents the retention time. "R = 1M" means that retention time is 1 month. The vertical axis is the relative relationship of RBER. From the figure, it can be seen that the RBER of the three pages is uneven. Therefore, different pages should have different LLR values.

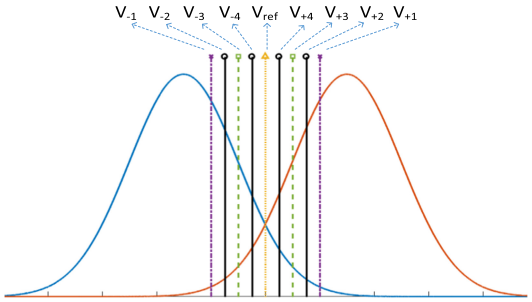

Fig. 3. The strategy of reference voltage placement.

2.2 Sensing Techniques

There are two ways of LDPC decoding: hard-decision and soft-decision decoding. As shown in Figure 3, if memory sensing uses only one reference voltage V_{ref} between two adjacent states, it is called *hard-decision memory sensing*, corresponding to hard-decision decoding. Otherwise, if more sensing levels such as $V_{\pm1}, V_{\pm2}, V_{\pm3}, V_{\pm4}$ are adopted, it is called *soft-decision memory sensing*, corresponding to soft-decision decoding. The more the voltages, the better the decoding performance as the accuracy of the LLR raises. But the latency increases meanwhile. Therefore, it is significant to choose a suitable reference voltage configuration, including boundaries and numbers. In this article, through simulation, we choose the leftmost voltage and the rightmost voltage of each HDRV and define them as the SDRV boundaries which are represented as V_{-1} and V_{+1}.

In most controller logic, hard-decision decoding is scheduled with the highest priority. Only when the hard-decision decoding fails, soft-decision decoding would be managed for data recovery. Therefore, the performance of SSDs is dominantly determined by the decoding failing rate of hard-decision decoding.

2.3 LLR Calculation and Decoding Algorithm

Let V_{th} represent the sensed threshold voltage of a cell, and we simply assume that each bit in a cell has *a priori* probability of 0.5 being 0 or 1. In our work, the corresponding LLR is negative for bit "0" and the LLR of bit "1" is positive. The LLR of the ith bit stored in one cell is calculated through the following formula (1):

$$L\left(\mathrm{b}_i\right) = \log \frac{p\left(b_i = 1|V_{th}\right)}{p\left(b_i = 0|V_{th}\right)} = \log \frac{p\left(V_{th}|b_i = 1\right)}{p\left(V_{th}|b_i = 0\right)} \tag{1}$$

Combining threshold voltage distribution, assuming that the threshold voltage V_{th} falls into the range $(R_l, R_r]$ (where R_l and R_r are two adjacent reference voltages), formula (1) can be written to formula (2) [9]:

$$L\left(\mathrm{b}_i\right) = \log \frac{\int_{R_l}^{R_r} \sum\limits_{P_k \in S_i} p^{(P_k)}\left(x\right) dx}{\int_{R_l}^{R_r} \sum\limits_{P_k} p^{(P_k)}\left(x\right) dx - \int_{R_l}^{R_r} \sum\limits_{P_k \in S_i} p^{(P_k)}\left(x\right) dx} \tag{2}$$

where, S_i denotes the set of states whose ith bit is mapped with 1, and $p^{(P_k)}\left(x\right)$ is the **probability density function (PDF)** of the threshold voltage for the P_k storage state. The numerator of $L\left(\mathrm{b}_i\right)$ represents the probability of all states in S_i falling into the range of $(R_l, R_r]$.

Decoding Algorithm. The sum-product algorithm [12] as a near-optimal decoding algorithm was provided by Gallager, and is also known as the **belief-propagation algorithm (BPA)**. BPA has

an extremely higher decoding complexity, which makes it very difficult to implement in NAND flash [21]. Conversely, the **min-sum algorithm (MSA)** [11, 14] and **layer min-sum algorithm** [6, 25] are introduced to reduce the complexity. HDD manufacturers and NAND-based storage manufacturers often quote UBER values on their datasheets, typically 10^{-13} to 10^{-16} [28]. So, we should simulate the FER whose magnitude is as low as possible. However, that is time-consuming. In our experiment, the layer min-sum algorithm is used, which currently can increase simulation efficiency greatly on a multi-core device using **Single Instruction Multiple Data (SIMD)** and **Single Program Multiple Data (SPMD)** programming models [20].

2.4 Related Work

In order to ensure the consistency of decoding performance, it is necessary to improve the accuracy of LLR, and many works have studied this issue. Zhang et al. [37] exploited numerical correction characteristics of retention errors, and proposed a retention-error-aware LDPC decoding scheme to improve NAND flash read performance. Zhang et al. [36] proposed a Pair-Bit-Errors-aware LDPC decoding scheme. Through the FPGA hardware test platform, the Pair-Bit error feature of MLC flash memory is obtained and the initial information is pre-processed during the decoding process to reduce the decoding delay. Kim et al. [18] studied the interference characteristics of MLC NAND flash memory and developed interference estimation and interference mitigation scheme. The threshold voltage distribution after interference mitigation is used to calculate LLRs and improve data reliability. Chen et al. [7] developed a non-uniform voltages placement scheme by exploring the error model to optimize the read reference voltages and improve decoding performance for MLC NAND flash. Ge et al. [13] explored the MLC NAND flash channel model in a radiated environment and developed a write voltage optimization scheme using this model. Wang et al. [29, 30] proposed a method to optimize the selection of Word line voltage by maximizing mutual information, optimizing the accuracy of LLR, and reduced the decoding latency. Luo et al. [24] built an online threshold voltage distribution model, showing the shift of threshold voltage with P/E cycles. It can be used to optimize the sensing voltage and decoding. This scheme is highly accurate but complex, and requires rich data to train a high-precision model, which is time-consuming. Ho et al. [15] dynamically applied soft-decision voltages based on the shift of threshold voltage to reduce the bit error rate and soft-decision decoding delay. Xie et al. [33] used lossless data compression to save storage space and provided protection for low-rate LDPC codes to reduce soft-decision detection delay. Li et al. [22] proposed a smart sensing level placement scheme to reduce voltages level and increase LDPC decoding performance for MLC NAND flash. However, little research has been done on hard-decision, which is a crucial part. For soft-decision decoding, the above research relies on various interference. Our test of 3D FG TLC NAND flash shows that the threshold voltage is symmetrical after interfered. The interference patterns for TLC NAND flash changes resulting in the current MLC solutions are no longer applicable, as they are designed based on the interference pattern. TLC NAND flash needs an LDPC algorithm with stronger error correction capability.

3 THRESHOLD VOLTAGE MODEL

For different retention times and P/E cycles, there are various threshold voltage distributions. We test the recently released 64-layer 3D FG TLC NAND flash under the different retention periods and P/E cycles. In total, 64 chips are tested. These chips are scattered in different parts of the SSD, with high representativeness. For FG-type chips, the test results are similar. However, due to the different working principles of Charge Trap (CT)-type chips, the results are different. For each chip, 2 dies are selected, which contains 15 blocks, and each block contains 744 pages, to ensure that the test data represents the overall situation. There are some researches on the threshold

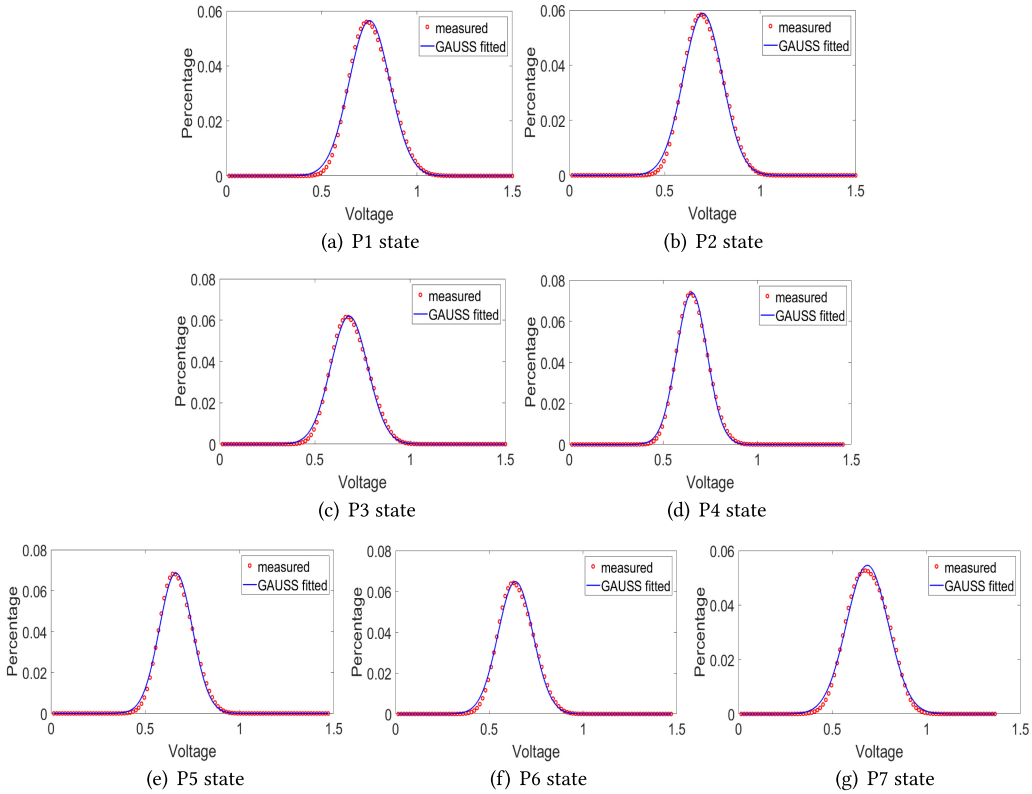

Fig. 4. The Gaussian distribution fits the threshold voltage distribution of each state (except erase state) according to the measured data.

voltage distribution model, such as Gaussian-based Model [4], Normal-Laplace-based Model [26], and Student's t-based Model [23]. Although the last two methods have high accuracy, the computation complexity and storage overhead cannot be ignored. The overhead includes time and storage overhead. Normal-Laplace-based model and student's-based model are computationally complex, and the latency is 89.3% and 31.3% higher than the Gaussian-based model, respectively [23]. These three models only need to store the key parameters. The storage overhead is very small and can be ignored. This article uses the normal distribution model to continue the expansion of the following content. If someone wants higher accuracy and the increased overhead is within an acceptable range, one can use the other two models. The subsequent research process is applicable to various threshold voltage distribution models. In this article, we choose the Gaussian distribution, which is a popular fitting method [3] to fit the threshold voltage distribution. In addition, the mean and standard deviation are fine tuned to improve accuracy. Fine tuning means that it is not a standard Gaussian distribution; the mean and the standard deviation are obtained by fitting. Figure 4 shows the fitting results for 5,000 P/E cycles, and retention time is 30 days. Table 2 characterizes the overall evaluation of the fitting accuracy under all cases in this article. Usually, the **sum of squares due to error (SSE)**, **coefficient of determination (R-square)**, and **root mean squared error (RMSE)** are used to judge the quality of the fitting. After counting the results in various situations, all three parameters indicate a good fitting, with SSE around 0.06, R-square > 0.950, and RMSE $\leq$ 0.09. Moreover, we find that interference only causes a uniform drop of voltage and keeps the

Table 2. The Table of Threshold Voltage Fitting Parameters

Parameter	P1	P2	P3	P4	P5	P6	P7
SSE	0.06	0.04	0.05	0.07	0.07	0.04	0.06
R-Square	0.975	0.960	0.982	0.984	0.976	0.950	0.986
RMSE	0.05	0.04	0.05	0.04	0.06	0.09	0.05

Table 3. RMSE of Blocks within the Same Chip

Number	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12	B13	B14	B15
C1	0.03	0.02	0.03	0.03	0.03	0.02	0.03	0.03	0.04	0.03	0.03	0.02	0.02	0.02	0.03
C2	0.04	0.03	0.03	0.03	0.03	0.02	0.03	0.04	0.05	0.03	0.02	0.02	0.03	0.04	0.02
C3	0.04	0.05	0.04	0.02	0.04	0.04	0.03	0.04	0.04	0.03	0.02	0.03	0.02	0.04	0.03
C4	0.10	0.05	0.05	0.12	0.06	0.07	0.09	0.04	0.03	0.03	0.02	0.02	0.02	0.03	0.04
C5	0.11	0.12	0.12	0.10	0.10	0.09	0.11	0.12	0.05	0.05	0.12	0.09	0.11	0.11	0.06

distribution still in a symmetrical form. Therefore, we choose Gaussian distribution to fit the threshold voltage distribution. The mean and standard deviation for various P/E cycles and retention times are shown in the appendix.

Within the same chip, the threshold voltage fitting accuracy of different blocks is different. In order to show the difference in accuracy, we calculate the RMSE of the blocks. The error range of our threshold voltage fitting model is [0.01,0.12]. The fitting accuracy is low on some blocks, and the RMSE reaches 0.12. However, this kind of situation is rare, and overall, the model fits well.

Five representative results are listed in Table 3. C1 represents that the number of the chip is 1. B1 represents block number 1. C1-C3 show results appear in most cases, with small errors and RMSE < 0.05. In some chips, there are differences in the fitting accuracy of different blocks. Like the result of C4, most of the blocks have high accuracy, and a small part of the blocks have large fitting errors. The worst result is the fitting situation corresponding to C5, and the fitting accuracy of most blocks is a little low. But the last two situations are rare, accounting for less than 9%.

The main reason for the low accuracy of the model is that the top fitting of the Gaussian distribution is poor. The tail of the distribution is important in decoding, and a large number of errors are generated at the intersection of the tails between the two adjacent states. The top of the distribution generally does not cause errors in decoding, because the LLR value corresponding to this part is very large. A large LLR value means that it has a large probability in decoding, and this value is almost impossible to be an error. Therefore, the inaccuracy of the top fitting does not have a serious impact on the reference voltage selection and decoding. Even with the fitting results of C5, the decoding performance is not bad.

To verify the versatility of the model, we randomly select 14 blocks in the same chip to obtain the distribution. The RMSE between the data of the 15th block and the fitted distribution is calculated. Through calculation and observation, in most chips, the data consistency of different blocks is high, the fitted model is stable and the cross-validation performance is good. There are gaps in the data of different blocks in some chips, so the cross-validation results are slightly worse. In general, the RMSE of the 15th block is less than 0.085, which is acceptable.

In our test, most of the area of erase state is undetectable, and its standard deviation is large, which is recognized. Referring to [3], we assume its average voltage is -1V. Figure 5 demonstrates a example of the fitted threshold voltage distribution, it's configuration is same with Figure 4.

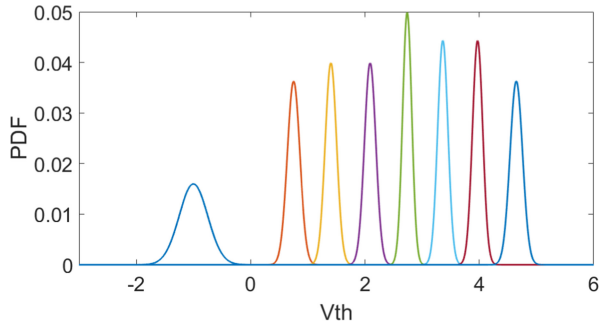

Fig. 5. Threshold voltage description of 3D TLC NAND flash based on measured data.

4 HARD-DECISION DECODING

The LDPC decoding is essentially a process in which LLRs are continuously updated during iteration. Each LLR consists of a sign and a value. The decoding results are based on the signs of the LLRs. For hard-decision sensing, the threshold voltage window of the cell is determined through a series of comparisons with $V_{ref}^{i}, (i = 0, 1, \ldots, 6)$. Then, the corresponding LLR is obtained as a hard-decision input to the decoder in the SSD controller, according to the three bits carried by the window. In the traditional method, the input LLR values are all the same, only the signs of LLRs are different. This is also the commonly used method.

With the fitted threshold voltage distribution model, we select the HDRV based on the principle of minimizing RBER. The probability of the bit being judged incorrectly P_{err}^{s} when the voltage is between the two states P_s $(s = 0, 1, \ldots, 6)$ and P_{s+1} can be expressed as formula (3). V_{ref}^{s} $(s = 0, 1, \ldots, 6)$, which minimizes P_{err}^{s} $(s = 0, 1, \ldots, 6)$, is the optimum HDRV between the P_s state and the P_{s+1} state.

$$P_{err}^{s} = \sum_{i=0}^{s} \int_{V_{ref}^{s}}^{+\infty} p^{(P_i)}(x) + \sum_{i=s+1}^{7} \int_{-\infty}^{V_{ref}^{s}} p^{(P_i)}(x) \tag{3}$$

During the process of choosing the optimal HDRV, it is found that we can decide the optimal $V_{ref}^{1}, V_{ref}^{2}, \ldots, V_{ref}^{6}$ according to formula (4). V_{ref}^{s} $(s = 0, 1, \ldots, 6)$, which minimizes P_{err}^{s} $(s = 0, 1, \ldots, 6)$, is the optimum HDRV. There are two reasons. On the one hand, the standard deviation of each state is relatively small. On the other hand, the errors are mainly caused by the transition of the adjacent states

$$P_{err}^{s} = \int_{V_{ref}^{s}}^{+\infty} p^{(P_s)}(x) + \int_{-\infty}^{V_{ref}^{s}} p^{(P_{s+1})}(x). \tag{4}$$

The proposed method is to deal with situations where the real flash memory characteristics cannot be obtained or the access cost is high. When the proposed scheme is started, the test is performed every 500 P/E cycles to obtain the threshold voltage distribution. The three tables in the appendix show the details of the mean and standard deviation of the threshold voltage distribution under different P/E cycles. It can be seen that the changes in the mean and standard deviation between 500 or 1,000 P/E cycles are small. In order to increase the accuracy of the model, we obtain the test data and the threshold voltage distribution every 500 P/E cycles. In the early life of the NAND flash, the threshold voltage distribution changes little within 500 P/E cycles. At the end of the life (for example, after P/E cycle > 3000), the threshold voltage distribution changes more within 500 P/E cycles. The accuracy of the threshold voltage model is slightly reduced. Based on

this, the P/E cycle interval of the test can be increased in the early life of the NAND flash. When the NAND flash reaches the late stage, a test can be performed in a lower P/E interval (such as 300) to obtain the threshold voltage distribution. Although the reduced accuracy is still acceptable, the above operation increases the accuracy of the model and improves the decoding performance. Next, the HDRV is calculated and updated.

This article examines the two improvement schemes of hard-decision decoding. Unlike traditional methods, which only focus on the sign of the LLR, these two mechanisms also take the value of the LLR into account. We adopt a similar idea of soft-decision decoding, and let the magnitude of the LLR value represent the reliability of the LLR sign. As shown in Figure 2, the probabilities that three bits are judged as wrong are different and the LLR values should be differentiated to increase accuracy. The first method, called method 1, is to roughly adjust the LLR value based on the level of the reliability of the three bits. Within the scope of the LLR quantization value, search the LLR and simulate the decoding error rate to find the best value. The value corresponding to the lowest error rate is the LLR finally used. Since this method is to break through the original hard-decision scheme based on the soft-decision mechanism, it is straightforward and can be understood as a trial operation. Although this way matches the actual situation more closely, it is not accurate enough.

The second way, called method 2, reveals the difference between the LLR values of the three bits more accurately, calculating LLR directly based on HDRV and threshold voltage distribution. This is a unique innovation of this article. Method 2 is the theoretical result of method 1 essentially. Method 2 is specifically described below.

Once the HDRV is determined, we can calculate the floating-point LLRs using formula (2) based on fitted threshold voltage distribution model. Here, although the model is not very fine grained, it is enough to calculate LLR, which is the integral of PDF in the reference voltage interval. In the hardware implementation of the LDPC decoding algorithm, floating-point LLRs are generally converted to fixed-point LLRs. But the conversion loses accuracy. We propose a quantization scheme that can reduce the loss of quantization.

The floating-point LLRs are quantized by two steps, which are quantization and saturation. Assuming q is the quantized bit width, the maximum modulus of quantized LLRs is $max = 2^{(q-1)} - 1$. Let x denote the floating-point LLR. Set the quantized LLR as $q(x)$, and let $Q(x)$ denote the saturation result. The quantization is carried out using formula (5).

$$q(x) = \left\lfloor \beta \times \frac{x}{\min(x)} + \gamma \right\rfloor \tag{5}$$

Here, β and γ are two constants. In general, β is set to 1, and γ is to ensure that LLR $\neq 0$ during the decoding process, because a correction coefficient less than 1 is multiplied in min-sum algorithm.

Formula (6) is used to get the final fixed-point LLR.

$$Q(x) = \begin{cases} -\max, & q(x) \leq -\max \\ q(x), & -\max < q(x) \leq \max \\ \max, & q(x) > \max \end{cases} \tag{6}$$

At different stages of the 3D TLC NAND flash, the threshold voltage distribution can be fitted online according to formulas (5) and (6). Based on the threshold voltage distribution, the floating-point LLR value can be obtained by the integration of the threshold voltage distribution function on the reference voltage interval. Depending on the formula, the final table can be obtained by converting the floating-point LLR into a fixed-point LLR. We save the LLR value as a lookup table, which can be used in the next same situation. The conversion from the floating-point LLR to the fixed-point LLR is pre-calculated to simplify implementation complexity. The table is only 1 KB for AsLDPC, storage overhead is negligible.

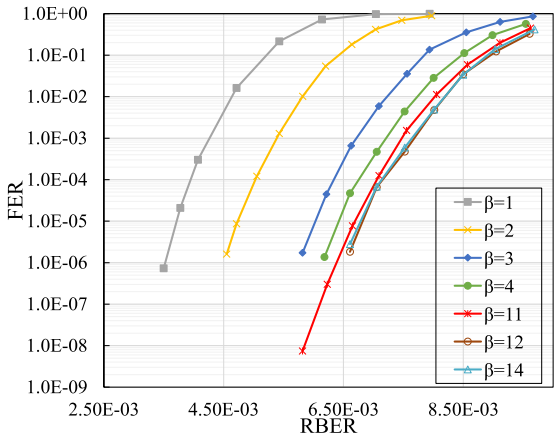

Fig. 6. The impacts of parameter β on performance in hard-decision decoding.

The floating-point LLRs in different reference voltage regions vary greatly. Considering that the LDPC decoder [17] is only sensitive to small LLRs, it is reasonable to preserve their precisions by amplifying them with factors β and γ. Inevitably, some large LLRs are saturated. By choosing a proper value of β and γ, we can mitigate the negative effect caused by saturation and keep the precision as well. In this work, we optimize β and γ using simulations. β is the main parameter that is decided by numerical approximation to control the size of the LLR. γ is used to ensure that the final value is non-zero. Once there is no 0 in the LLRs, set γ to 0.

We compare the performance of different βs in hard-decision decoding, as shown in Figure 6. We can see that when using general parameter $\beta = 1$ ($\gamma = 1.5$), the performance is poor. Amplifying some LLRs by increasing β can improve performance because of the sensitivity to the small LLRs of the decoder. But the performance gain is not obvious when it reaches a certain level. In detail, when $\beta = 2$ ($\gamma = 0$), the performance improvement is obvious. After $\beta > 3$ ($\gamma = 0$), the improvement is gradually stable, and when $\beta = 11$ ($\gamma = 0$), the FER is quite low. Continuing to adjust β is not meaningful. Therefore, $\beta = 11$ and $\gamma = 0$ are final choices. Due to the underestimation of the size of the tail distribution based on the Gaussian model, β of this distribution is larger than that of the more accurate distribution. Properly amplifying the LLR is beneficial in different distribution models as the decoder is sensitive to small values. The accuracy of the LLR could be measured by calculating correlation coefficients between our LLRs and the original LLRs [32]. By calculating, this value of our hard-decision LLR sequence is 0.9, showing a high degree of correlation with the original LLR sequence. The correlation coefficient of Normal-Laplace-based Model and Student's t-based Model are 0.94 and 0.925. These two models are better, but not much different from the Gaussian distribution model.

5 SOFT-DECISION DECODING

Our previous research [36] focuses on pair-bit error characteristics. When this feature does not exist, the solution could not be used. This article analyzes and studies the influencing factors in the entire decoding process, including the following three parts (see 5.1–5.3).

5.1 Threshold Voltage Distribution

After exceeding the scope of the error correction capability of the hard-decision decoding, it is necessary to switch to soft-decision decoding. The soft-decision sensing is to place more reference

Table 4. The Optimal Voltage Difference Δ of Six Overlap Regions

Δ_s	$0-0.05$	$0.05-0.09$	$0.09-0.15$	$0.15-0.18$
Δ	0.12	0.125	0.13	0.135

voltages on both sides of each HDRV. Through multiple reads, the more precise threshold voltage decision region for the cell is gained, and a more accurate LLR sequence is obtained. Soft-decision decoding has a stronger error correction capability, so it can correct the higher RBER. Like hard-decision decoding, threshold voltage distribution is obtained periodically. After that, the relevant threshold voltage can be characterized.

5.2 Reference Voltage

The calculation of floating-point LLRs needs to know threshold voltage distribution and read reference voltages. The former is recognized. Afterward, we need to select a suitable SDRV configuration between adjacent states, including two steps:

(1) The setting of SDRV boundaries consists of the leftmost voltage and the rightmost voltage around each HDRV (defined in Section 2.2).
(2) After determining the SDRV boundaries, select an appropriate number.

For different P/E cycles and retention time, the principle of voltage selection is the same. First, we choose SDRV boundaries. Since the standard deviation of erase state is large, the reference voltage between P0 state and P1 state is considered separately. Through simulation, we can determine the boundaries of the reference voltage.

Set V_{-1}^{i} and V_{+1}^{i}, $(i = 1, 2, \ldots, 6)$ as the leftmost and rightmost SDRV of each HDRV, respectively. The reference voltage differences between SDRV and HDRV are $\Delta_{left}^{i} = V_{ref}^{\mathrm{i}} - V_{-1}^{\mathrm{i}}$ and $\Delta_{right}^{i} = V_{+1}^{\mathrm{i}} - V_{ref}^{\mathrm{i}}$, respectively. The optimal Δ is determined by the optimal SDRV and the HDRV. The acquisition of the HDRV is explained in Section 4. So here we describe the selection of the SDRV. We choose the leftmost and rightmost SDRV between every two adjacent states through scanning voltages from the HDRV in the first state to the HDRV in the second state. The voltage resulting in the fewest errors is the final SDRV. Accordingly, the distance between the SDRV and the HDRV is optimal Δ. Since the Δ is highly related to the difference between the standard deviations of the two states, we collate their relationship, which can be seen in Table 4. Δ_s is the value that standard deviation of the first state minus standard deviation of the second state. For example, when Δ_s is from 0 to 0.05, the optimal Δ is 0.12.

During simulation search, we observe:

(1) Although there is a relatively big difference in standard deviations among the various overlap regions, about 3 dB, the performance can reach optimum when V_{-1}^{i} and V_{+1}^{i} are symmetric about V_{ref}^{i};
(2) In the case of $\Delta_{left}^{i} = \Delta_{right}^{i}$, we choose an overall $\Delta^{\mathrm{i}} = \Delta_{left}^{i} = \Delta_{right}^{i}$. As presented in Figure 7, the performance that this Δ^{i} is used for six overlap regions is fine, and it is almost identical to the performance that each region using their respective optimal Δ^{i}.

According to Observation (2), we see that the decoder is not sensitive to voltage difference Δ^{i}. So we set the voltage difference Δ^{i} to be the same for the six regions before next step. Furthermore, between the P0 state and the P1 state, the voltage differences Δ_{left}^{1} and Δ_{right}^{1} are asymmetric, and according to the experiments, their difference is within 0.2.

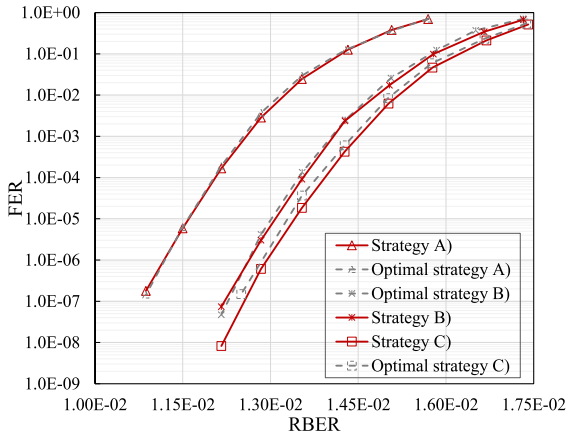

Fig. 7. The performance comparison between the mode that optimal voltage difference Δ is adopted and the mode that the selected voltage difference Δ is adopted in soft-decision decoding.

We know that increasing the number of reference voltages for soft-decision decoding improves the decoding performance but increases the read latency. To explore this issue, more reference voltages are sequentially added in the middle of the determined reference voltages according to the principle of dichotomy. As shown in Figure 3, this process is the conversion from strategy A) → strategy B) → strategy C).

Strategy A): Three reference voltages are applied between adjacent states: V_{ref}, V_{-1}, V_{+1}.

Strategy B): Five reference voltages are applied between adjacent states: $V_{ref}, V_{-1}, V_{+1}, V_{-3}, V_{+3}$.

Strategy C): Nine reference voltages are applied between adjacent states: V_{ref}, V_{-1}, V_{+1}, $V_{-3}, V_{+3}, V_{-2}, V_{+2}, V_{-4}, V_{+4}$.

5.3 LLR Quantization

After the reference voltages are selected, we first calculate the LLRs using formula (2), and then quantize the floating-point LLRs. The quantification scheme which is expounded by formulas (5) and (6) is also applicable to soft-decision decoding. Similarly, adjusting factors β and γ can greatly reduce the error rate. For the above three strategies, LLR quantization scheme without optimizing parameters is applied to simulate the traditional situation. The performance after optimizing β and γ represents the proposed scheme. By comparing the error rate of these two types, the effectiveness of the new method can be verified. In this part, the correlation coefficient between the adjusted LLR and the origin LLR is 0.95, explaining the high accuracy of our LLR.

6 SIMULATION RESULTS

In this section, we first introduce the simulation setup, and then give the results and analysis for hard-decision decoding and soft-decision decoding, respectively.

6.1 Simulation Setup

In this article, the LDPC code we use for simulations is a 2KB Quasi-Cyclic LDPC (QC-LDPC) that we construct with a code rate of 0.9. The configuration of quantization is a 6-bit sign quantization for initial LLRs, that is, LLRs belong to [−31,31] after quantification. The layer min-sum algorithm is used, which is accelerated through the SIMD instruction set. Moreover, the correction coefficient is 0.75 and the maximum number of iterations is 10.

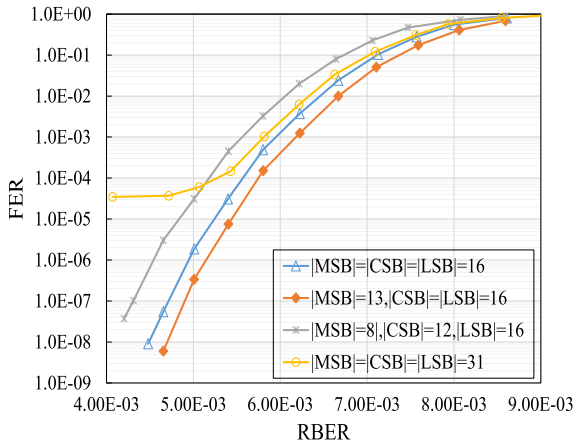

Fig. 8. Performance comparisons of the proposed scheme and traditional methods in hard-decision decoding.

6.2 Hard-Decision Results

For the first way, we can see the results in Figure 8. First, we find that different LLR values can influence decoding performance in the traditional method. For example, when the absolute values of LLR are all set to 31, the error floor appears. When the LLR is reduced by half, this problem disappears. This phenomenon also confirms that the improper selection of fixed-point LLRs has a serious impact on performance. Then we adjust the LLR values based on different error rates, as we propose in method 1. The baseline of LLR is equal to 16. Through experiments, it is found that different LLR combinations lead to various results. Among them, when the MSB is equal to 13 and the CSB is equal to 16, the performance is optimal, which is improved by an order of magnitude compared with the same value 16 when it is used for the three bits. But when the value is set unsuitable, such as the MSB is equal to 8 and the CSB is equal to 12, the performance is worse than the original one.

For a more precise way (proposed method 2), we get the LLR through calculation. By exploiting the proposed scheme, we calculate the LLR of each bit in different hard-decision voltage regions according to the threshold voltage model and hard-decision voltages. The selected quantization parameters are $\beta = 11, \gamma = 0$. We save the fixed-point LLR as an offline table, then look up the table directly when decoding.

It should be noted that we use 16 and 31 as the same LLR modulus in traditional solutions to compare the decoding performance. For 31, the error floor appears prematurely, while when using 16, the performance is good. As shown in Figure 9, by calculating the LLRs of various bit positions in different reference voltage regions, and improving the fixed-point LLRs according to the proposed scheme, the performance is significantly improved compared with the traditional solutions. For example, at RBER = 5.8×10^{-3}, the FER that using 16 in traditional method is about 1×10^{-3}, while the FER of the proposed quantization scheme is 7.5×10^{-9}. The performance improvement is from LLR quantization, so the two schemes are suitable for different P/E cycles and retention time. We compare our algorithm performance with the **adapted Probability based Gradient Descent Bit Flipping (A-PGDBF)** algorithm [19]. Although A-PGDBF shows good performance, our algorithm has a lower FER.

6.3 Soft-Decision Results

We compare the performance with strategy A), strategy B), and strategy C), as illustrated in Figure 10. Compared with strategy A), strategy B) has a large performance gain, while strategy C)'s

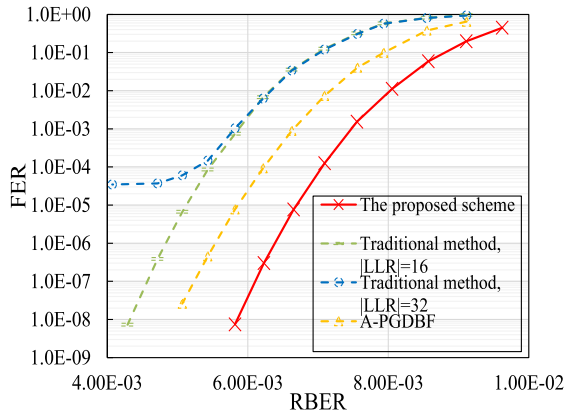

Fig. 9. Performance comparisons of the proposed scheme and traditional methods in hard-decision decoding.

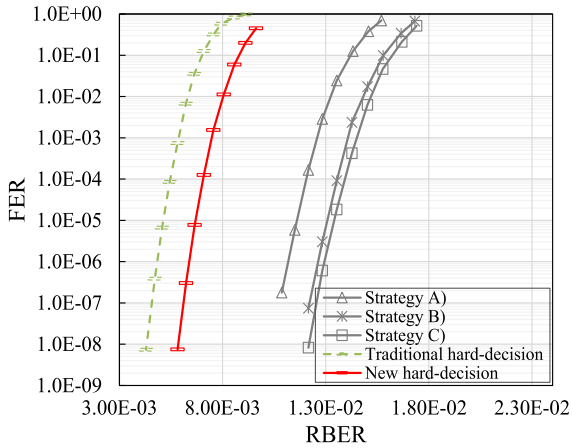

Fig. 10. The performance comparisons among different strategies of voltage number in soft-decision decoding, and their performance comparisons with hard-decision decoding.

performance gain is small compared with strategy B). When increasing the number of reference voltages, a more accurate voltage region that cell is located can be obtained. Therefore, a more accurate LLR can be gained, improving the decoding performance. Although increasing the number of reference voltages can improve the decoding performance, the improvement becomes insignificant after reaching a certain level. In a word, using three or five reference voltages is a good solution for soft-decision decoding.

In addition, as shown in Figure 10, we can see that our new scheme has significantly narrowed the gap between the performance of traditional hard-decision decoding and soft-decision decoding. There is a relatively big gap between the performance of strategy A) and strategy B). Thus, we adjust the parameters to improve the performance of strategy A) using numerical approximation. β as the main parameter needs to be analyzed in detail, as shown in Figure 11.

From Figure 11, we can see that decreasing β can result in degrading performance, and increasing it can improve performance. But when it increases to a certain degree, performance starts to decline. Finally, the selected quantization parameters are $\beta = 2(\gamma = 0)$ (when $\beta < 2, \gamma = 1.5$, else,

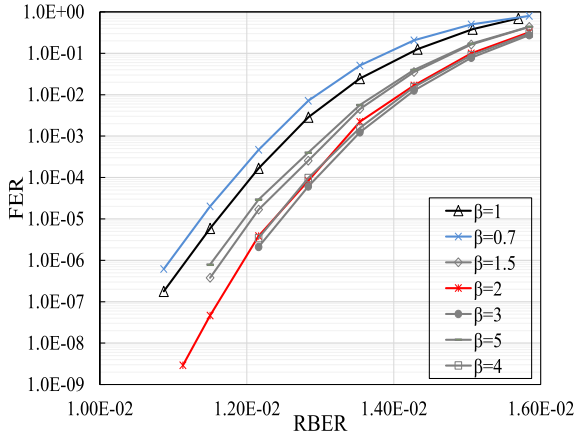

Fig. 11. The impacts of parameter β on performance in soft-decision decoding.

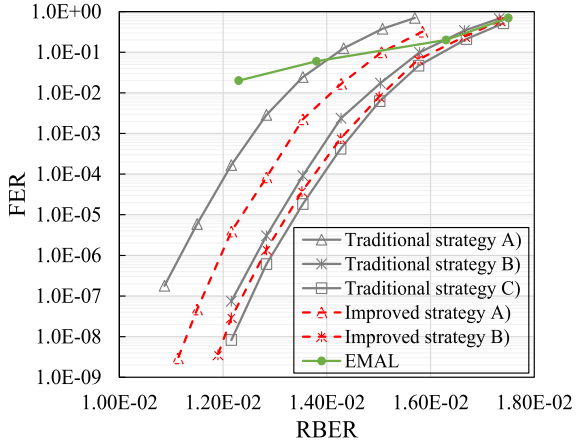

Fig. 12. The performance comparisons of EMAL, ordinary LLR and improved LLR for strategy A) and strategy B) in soft-decision decoding.

$\gamma = 0$). We use formula (6) for saturation. As shown in Figure 12, compared with traditional situation, FER performance of strategy A) is improved by two orders of magnitude and the acceptable RBER is also increased obviously. For example, at FER = 10^{-7}, the acceptable RBER of strategy A) is about 1.08×10^{-2}, and for strategy B), that is about 1.23×10^{-2}. The RBER of the improved strategy A) is about 1.18×10^{-2}, which is approximately the middle of strategy A) and B). For hard and soft decisions, MSB has a lower LLR value.

The effectiveness of strategy B) is also verified. Since the performance gap between strategy B) and strategy C) is quite small, the performance improvement of strategy B) cannot be obvious. Similar to the previous parameter selection principle, after adjusting the fixed-point LLRs, where $\beta = 2(\gamma = 0)$, the error rate is reduced, which is almost the same as that of strategy C). The simulation results are shown in Figure 12. Overall, after the fixed-point LLRs are optimized, the superior performance can be obtained by using only five reference voltages. For various P/E cycles and retention time, the performance curves show consistent trends. Besides, in order to prove the effectiveness of the proposed scheme, we compared the results with **error modes aware LDPC**

(EMAL) [31] at the error rate level. From Figure 12, we can see that EMAL's FER declines quickly at first, and it has the same performance as our strategy C). However, after RBER reaches to 1.6×10^{-2}, EMAL appears as the error floor. In other words, the advantage of EMAL is that the error rate is excellent when the RBER is low, but when the RBER gradually increases, its advantage in reducing the error rate is not as good as our mechanism. When RBER reaches 1.3×10^{-2}, the performance of EMAL is worse than our scheme using three reference voltages. In the test interval, our algorithm has a lower FER in most RBER. Overall, our mechanism has better performance. For both hard-decision decoding and soft-decision decoding, the LLR table is calculated before decoding and saved into decoder. When the algorithm is implemented, the overhead is principally the storage overhead of the LLR table, which is only 1 KB and can be ignored.

6.4 Overhead Analysis

The proposed method generates space and time overhead. The space overhead comes from storing the mean and standard variance of the threshold voltage distribution, the voltage position and the LLR table. These are the same as the cost of other methods, only the value is changed. For our algorithm, the storage overhead is 2 KB and can be ignored. The time cost includes the time for fitting the distribution and for calculating the reference voltage and the LLR value. The latter two contents are the same as the calculation process of the conventional method. The latency of fitting the distribution comes from reading the data and fitting process, which takes 28 ms each time on average. The delay to each read/write operation is 52 ns.

7 CONCLUSION

In order to improve the decoding performance of LDPC for 3D TLC NAND flash, this article first establishes the threshold voltage distribution model of 3D TLC NAND flash based on the measured data, and finds that the fitting of Gaussian distribution is good. Then, according to the model, after selecting the optimal HDRV, we propose schemes to calculate the fixed-point LLRs of three bits in different HDRV regions offline. For hard-decision decoding, the decoding performance is significantly improved compared with traditional solutions. For the soft-decision decoding, we first select the appropriate voltage configuration, and then calculate the LLR lookup table. In the experiment, we find that, although the standard deviations of various states are different, the performance that the SDRVs are placed symmetrically around the HDRV can be optimal, and the same voltage difference can be used among different overlap regions. Based on the above conditions, we also study the effect of the reference voltages number on the performance, and find that when the number of reference voltages achieves five, the performance improvement becomes less obvious. In addition, we use the new quantization scheme to compensate for the performance loss. Simulation results show that the decoding performance is increased by two orders of magnitude when using three reference voltages.

APPENDIX

A THE MEAN AND STANDARD DEVIATION FOR VARIOUS CASES

We enlarge the mean and standard deviation to a certain extent, which is from Reference [3], to more prominently show their difference under various P/E cycles and retention time. The mean for P0 is a hypothetical value (see Tables A1 through A3).

Table A1. Normalized Mean (Top) and Standard Deviation (Bottom) for Threshold Voltage Distribution of Various P/E Cycles at 15 Days

P/E Cycles	P0	P1	P2	P3	P4	P5	P6	P7
500	−85	64.5	121.9	181.6	236.4	289.5	341.2	401.5
1,000	−85	64.8	122.1	181.8	236.6	289.6	341.5	402.1
2,000	−85	65.4	122.5	182.0	236.8	289.9	341.7	402.8
3,000	−85	65.2	122.4	181.8	236.7	289.9	341.8	402.2
4,000	−85	65.0	122.3	181.7	236.7	289.9	342.1	402.9
5,000	−85	64.6	121.8	181.2	236.3	289.6	341.9	404.1
P/E Cycles	P0	P1	P2	P3	P4	P5	P6	P7
500	18.2	9.1	8.9	9.1	7.5	7.8	7.6	9.0
1,000	18.4	9.2	9.0	9.1	7.6	7.8	7.7	9.1
2,000	18.7	9.4	9.1	9.0	7.8	7.9	7.8	9.3
3,000	18.7	9.5	9.2	9.0	7.8	7.9	7.8	9.4
4,000	19.2	9.6	9.3	9.1	7.9	8.0	7.9	9.6
5,000	19.3	9.7	9.4	9.1	8.0	8.1	8.1	9.8

Table A2. Normalized Mean (Top) and Standard Deviation (Bottom) for Threshold Voltage Distribution of Various P/E Cycles at 30 Days

P/E Cycles	P0	P1	P2	P3	P4	P5	P6	P7
500	−85	65.1	122.2	181.8	236.5	289.4	341.0	401.4
1,000	−85	65.6	122.6	182.1	236.8	289.7	341.3	402.1
2,000	−85	65.6	122.5	181.9	236.7	289.7	341.5	402.8
3,000	−85	65.3	122.3	181.7	236.6	289.7	341.6	402.2
4,000	−85	65.0	122.1	181.4	236.4	289.6	341.8	402.2
5,000	−85	64.8	121.8	181.2	236.2	289.5	341.8	404.1
P/E Cycles	P0	P1	P2	P3	P4	P5	P6	P7
500	18.4	9.2	9.0	9.1	7.5	7.8	7.6	9.2
1,000	18.6	9.3	9.0	9.1	7.7	7.8	7.7	9.3
2,000	18.9	9.4	9.1	9.0	7.8	7.9	7.8	9.5
3,000	19.2	9.6	9.3	9.1	7.9	8.0	7.8	9.7
4,000	19.4	9.7	9.4	9.1	8.0	8.0	7.9	9.5
5,000	19.5	9.7	9.4	9.1	8.0	8.1	8.1	10.0

Table A3. Normalized Mean (Top) and Standard Deviation (Bottom) for Threshold Voltage Distribution of Various P/E Cycles at 90 Days

P/E Cycles	P0	P1	P2	P3	P4	P5	P6	P7
500	−85	65.6	122.5	182.0	236.6	289.4	340.9	401.4
1,000	−85	66.0	122.8	182.2	236.8	289.6	341.2	402.1
2,000	−85	65.9	122.6	181.9	236.7	289.7	341.4	402.8
3,000	−85	65.5	122.4	181.7	236.5	289.6	341.5	402.2
4,000	−85	65.2	122.1	181.4	236.4	289.6	341.7	402.2
5,000	−85	64.9	121.8	181.1	236.1	289.4	341.6	404.2
P/E Cycles	P0	P1	P2	P3	P4	P5	P6	P7
500	18.6	9.3	9.1	9.1	7.6	7.8	7.6	9.3
1,000	18.8	9.4	9.1	9.1	7.7	7.9	7.7	9.5
2,000	18.1	9.6	9.2	9.1	7.8	8.0	7.9	9.6
3,000	19.4	9.7	9.3	9.1	7.9	8.0	7.9	9.8
4,000	19.6	9.8	9.5	9.2	8.0	8.1	8.0	9.6
5,000	19.7	9.9	9.5	9.2	8.1	8.2	8.1	9.7

REFERENCES

[1] Lalit Bahl, John Cocke, Frederick Jelinek, and Josef Raviv. 1974. Optimal decoding of linear codes for minimizing symbol error rate (corresp.). *IEEE Transactions on Information Theory* 20, 2 (1974), 284–287.

[2] Raj Chandra Bose and Dwijendra K. Ray-Chaudhuri. 1960. On a class of error correcting binary group codes. *Information and Control* 3, 1 (1960), 68–79.

[3] Yu Cai, Saugata Ghose, Erich F. Haratsch, Yixin Luo, and Onur Mutlu. 2017. Error characterization, mitigation, and recovery in flash-memory-based solid-state drives. *Proc. IEEE* 105, 9 (2017), 1666–1704.

[4] Yu Cai, Erich F. Haratsch, Onur Mutlu, and Ken Mai. 2013. Threshold voltage distribution in MLC NAND flash memory: Characterization, analysis, and modeling. In *2013 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 1285–1290.

[5] Yu Cai, Yixin Luo, Erich F. Haratsch, Ken Mai, and Onur Mutlu. 2015. Data retention in MLC NAND flash memory: Characterization, optimization, and recovery. In *2015 IEEE 21st International Symposium on High Performance Computer Architecture (HPCA)*. 551–563.

[6] Jinghu Chen, Ajay Dholakia, Evangelos Eleftheriou, Marc P. C. Fossorier, and Xiao-Yu Hu. 2005. Reduced-complexity decoding of LDPC codes. *IEEE Transactions on Communications* 53, 8 (2005), 1288–1299.

[7] Shih-Liang Chen, Bo-Ru Ke, Jian-Nan Chen, and Chih-Tsun Huang. 2011. Reliability analysis and improvement for multi-level non-volatile memories with soft information. In *2011 48th ACM/EDAC/IEEE Design Automation Conference (DAC)*. IEEE, 753–758.

[8] Yoshiaki Deguchi, Shun Suzuki, and Ken Takeuchi. 2018. Write and read frequency-based word-line batch VTH modulation for 2-D and 3-D-TLC NAND flash memories. *IEEE Journal of Solid-State Circuits* 99 (2018), 1–10.

[9] Guiqiang Dong, Ningde Xie, and Tong Zhang. 2010. On the use of soft-decision error-correction codes in NAND flash memory. *IEEE Transactions on Circuits and Systems I: Regular Papers* 58, 2 (2010), 429–439.

[10] G. Forney. 1965. On decoding BCH codes. *IEEE Transactions on Information Theory* 11, 4 (1965), 549–557.

[11] Marc P. C. Fossorier, Miodrag Mihaljevic, and Hideki Imai. 1999. Reduced complexity iterative decoding of low-density parity check codes based on belief propagation. *IEEE Transactions on Communications* 47, 5 (1999), 673–680.

[12] Robert Gallager. 1962. Low-density parity-check codes. *IRE Transactions on Information Theory* 8, 1 (1962), 21–28.

[13] Guangjun Ge and Liuguo Yin. 2017. LDPC coding scheme for improving the reliability of multi-level-cell NAND flash memory in radiation environments. *China Communications* 14, 8 (2017), 10–21.

[14] Frédéric Guilloud, Emmanuel Boutillon, and Jean-Luc Danger. 2003. λ-min decoding algorithm of regular and irregular LDPC codes. In *3rd International Symposium on Turbo Codes and Related Topics*. 451–454.

[15] Kin-Chu Ho, Po-Chao Fang, Hsiang-Pang Li, Cheng-Yuan Michael Wang, and Hsie-Chia Chang. 2013. A 45nm 6b/cell charge-trapping flash memory using LDPC-based ECC and drift-immune soft-sensing engine. In *2013 IEEE International Solid-State Circuits Conference Digest of Technical Papers*. IEEE, 222–223.

[16] Hoda Aghaei Khouzani and Chengmo Yang. 2016. Towards a scalable and write-free multi-version checkpointing scheme in solid state drives. In *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)*. IEEE, 37–48.

[17] Jonghong Kim and Wonyong Sung. 2013. Rate-0.96 LDPC decoding VLSI for soft-decision error correction of NAND flash memory. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 22, 5 (2013), 1004–1015.

[18] Taehyung Kim, Gyuyeol Kong, Xi Weiya, and Sooyong Choi. 2013. Cell-to-cell interference compensation schemes using reduced symbol pattern of interfering cells for MLC NAND flash memory. *IEEE Transactions on Magnetics* 49, 6 (2013), 2569–2573.

[19] Khoa Le and Fakhreddine Ghaffari. 2018. On the use of hard-decision LDPC decoders on MLC NAND flash memory. In *2018 15th International Multi-Conference on Systems, Signals & Devices (SSD)*. IEEE, 1453–1458.

[20] Bertrand Le Gal and Christophe Jego. 2015. High-throughput multi-core LDPC decoders based on x86 processor. *IEEE Transactions on Parallel and Distributed Systems* 27, 5 (2015), 1373–1386.

[21] Huanlin Li, Yanyan Cao, and Jeffrey C. Dill. 2010. Analysis of error-prone patterns for LDPC codes under belief propagation decoding. In *2010 Military Communications Conference (2010 MILCOM)*. IEEE, 2056–2061.

[22] Qiao Li, Liang Shi, Chun Jason Xue, Qingfeng Zhuge, and Edwin H-M Sha. 2017. Improving LDPC performance via asymmetric sensing level placement on flash memory. In *2017 22nd Asia and South Pacific Design Automation Conference (ASP-DAC)*. IEEE, 560–565.

[23] Yixin Luo. 2018. Architectural techniques for improving NAND flash memory reliability. *arXiv preprint arXiv:1808.04016* (2018).

[24] Yixin Luo, Saugata Ghose, Yu Cai, Erich F. Haratsch, and Onur Mutlu. 2016. Enabling accurate and practical online flash channel modeling for modern MLC NAND flash memory. *IEEE Journal on Selected Areas in Communications* 34, 9 (2016), 2294–2311.

[25] Mohammad M. Mansour and Naresh R. Shanbhag. 2002. Low-power VLSI decoder architectures for LDPC codes. In *International Symposium on Low Power Electronics and Design*. IEEE, 284–289.

[26] Thomas Parnell, Nikolaos Papandreou, Thomas Mittelholzer, and Haralampos Pozidis. 2014. Modelling of the threshold voltage distributions of sub-20nm NAND flash memory. In *2014 IEEE Global Communications Conference*. IEEE, 2351–2356.

[27] Santini Paolo, Battaglioni Massimo, Baldi Marco, and Chiaraluce Franco. 2019. Hard-decision iterative decoding of LDPC codes with bounded error rate. In *2019 IEEE International Conference on Communications (ICC)*. IEEE, 1–6.

[28] C. van Ingen and J. Gray. 2005. *Empirical Measurements of Disk Failure rates and Error Rates*. Technical Report. MSR-TR-2005.

[29] Jiadong Wang, Thomas Courtade, Hari Shankar, and Richard D. Wesel. 2011. Soft information for LDPC decoding in flash: Mutual-information optimized quantization. In *2011 IEEE Global Telecommunications Conference (GLOBECOM 2011)*. IEEE, 1–6.

[30] Jiadong Wang, Kasra Vakilinia, Tsung-Yi Chen, Thomas Courtade, Guiqiang Dong, Tong Zhang, Hari Shankar, and Richard Wesel. 2014. Enhanced precision through multiple reads for LDPC decoding in flash memories. *IEEE Journal on Selected Areas in Communications* 32, 5 (2014), 880–891.

[31] Fei Wu, Meng Zhang, Yajuan Du, Weihua Liu, Zuo Lu, Jiguang Wan, Zhihu Tan, and Changsheng Xie. 2020. Using error modes aware LDPC to improve decoding performance of 3-D TLC NAND flash. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 39, 4 (2020), 909–921.

[32] Yunxiang Wu, Yu Cai, and Erich F. Haratsch. 2017. Fixed point conversion of LLR values based on correlation. US Patent 9,582,361.

[33] Ningde Xie, Guiqiang Dong, and Tong Zhang. 2010. Applying transparent lossless data compression to improve the feasibility of using advanced error correction codes in solid-state drives. In *2010 IEEE Workshop on Signal Processing Systems*. IEEE, 31–35.

[34] Qin Xiong, Fei Wu, Zhonghai Lu, Yue Zhu, You Zhou, Yibing Chu, Changsheng Xie, and Ping Huang. 2017. Characterizing 3D floating gate NAND flash. In *2017 ACM SIGMETRICS/International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS 2017) (University of Illinois Urbana-ChampaignUrbana-Champaign, IL, 5 June 2017 through 9 June 2017)*. Association for Computing Machinery (ACM), 31–32.

[35] Cristian Zambelli, Giuseppe Cancelliere, Fabrizio Riguzzi, Evelina Lamma, Piero Olivo, Alessia Marelli, and Rino Micheloni. 2017. Characterization of TLC 3D-NAND flash endurance through machine learning for LDPC code rate optimization. In *2017 IEEE International Memory Workshop (IMW)*. IEEE, 1–4.

[36] Meng Zhang, Fei Wu, Yajuan Du, Weihua Liu, and Changsheng Xie. 2019. Pair-bit errors aware LDPC decoding in MLC NAND flash memory. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 38, 12 (2019), 2312–2320.

[37] Meng Zhang, Fei Wu, Xubin He, Ping Huang, Shunzhuo Wang, and Changsheng Xie. 2016. REAL: A retention error aware LDPC decoding scheme to improve NAND flash read performance. In *2016 32nd Symposium on Mass Storage Systems and Technologies (MSST)*. IEEE, 1–13.

Received October 2020; revised May 2021; accepted June 2021