On the Use of Soft-Decision Error-Correction Codes in NAND Flash Memory

Guiqiang Dong, *Student Member, IEEE*, Ningde Xie, and Tong Zhang, *Senior Member, IEEE*

***Abstract*—As technology continues to scale down, NAND Flash memory has been increasingly relying on error-correction codes (ECCs) to ensure the overall data storage integrity. Although advanced ECCs such as low-density parity-check (LDPC) codes can provide significantly stronger error-correction capability over BCH codes being used in current practice, their decoding requires soft-decision log-likelihood ratio (LLR) information. This results in two critical issues. First, accurate calculation of LLR demands fine-grained memory-cell sensing, which nevertheless tends to incur implementation overhead and access latency penalty. Hence, it is critical to minimize the fine-grained memory sensing precision. Second, accurate calculation of LLR also demands the availability of a memory-cell threshold-voltage distribution model. As the major source for memory-cell threshold-voltage distribution distortion, cell-to-cell interference must be carefully incorporated into the model. However, these two critical issues have not been ever addressed in the open literature. This paper attempts to address these open issues. We derive mathematical formulations to approximately model the threshold-voltage distribution of memory cells in the presence of cell-to-cell interference, based on which the calculation of LLRs is mathematically formulated. This paper also proposes a nonuniform memory sensing strategy to reduce the memory sensing precision and, thus, sensing latency while still maintaining good error-correction performance. In addition, we investigate these design issues under the scenario when we can also sense interfering cells and hence explicitly estimate cell-to-cell interference strength. We carry out extensive computer simulations to demonstrate the effectiveness and involved tradeoffs, assuming the use of LDPC codes in 2-bits/cell NAND Flash memory.**

***Index Terms*—Cell-to-cell interference, low-density parity check (LDPC), NAND Flash, nonuniform sensing, reverse programming, soft-decision error-correction code (ECC).**

I. Introduction

THE STEADY bit-cost reduction over the past decade enabled the NAND Flash memory to enter increasingly diverse applications, from consumer electronics to personal and enterprise computers. In particular, this trend has made it economically viable to implement solid-state drives (SSDs) using NAND Flash memory, which is expected to fundamentally change the memory and storage hierarchy in future computing systems. The continuous bit-cost reduction of NAND Flash memory mainly relies on aggressive technology scaling, e.g., NAND Flash memory chips at sub-30-nm technology nodes have been recently announced by major NAND Flash manufacturers such as Samsung, Toshiba, and Micron. Aside from technology scaling, the multilevel-per-cell (MLC) technique, i.e., storing more than 1 bit in each memory cell (or floating-gate MOS transistor) by programming the cell threshold voltage into one of several voltage windows, has been widely used to further improve effective storage density and hence reduce the bit cost of NAND Flash memory. Because of its obvious advantages in terms of storage density and, hence, bit cost, MLC NAND Flash memory now largely dominates the global Flash memory market. In the current design practice, most MLC NAND Flash memories store 2 bits/cell, while 3- and even 4-bits/cell NAND Flash memories have been recently reported in the open literature [1]–[5].

Like any other data storage technologies such as magnetic and optical recording, NAND Flash memory must use error-correction codes (ECCs) to ensure the system-level data storage integrity, where BCH codes with classical hard-decision decoding algorithms [6] are being widely used in current design practice. As the industry continues to push the technology scaling envelope and pursue aggressive use of MLC storage, the raw storage reliability of NAND Flash memory inevitably continues to degrade, which could make current design practice inadequate and hence naturally demand the search for more powerful ECCs (e.g., see [7]). Because of their well-proven superior error-correction capability with reasonably low decoding complexity, advanced ECCs, such as low-density parity-check (LDPC) code [8], [9], Reed–Solomon codes with soft-decision decoding algorithms [10], [11], and Turbo code [12], appear to be promising candidates. For example, LSI Corporation, one of leading hard-disk-drive chip vendors, recently announced that LDPC codes have been used in their latest hard-disk-drive read-channel chips at the 40-nm technology node (see [13]), which undoubtedly sheds light on the potential of using LDPC codes in future SSDs.

These advanced ECCs are soft decision in nature, i.e., their decoding demands the log-likelihood-ratio (LLR) information of each bit and their error-correction performance heavily depends on the quality and accuracy of LLRs. As a result, NAND Flash memory chips must carry out fine-grained soft-decision memory-cell sensing (e.g., the threshold voltage of each cell in 2-bits/cell memory is quantized into 4 bits, corresponding to 15 sensing levels). Compared with the current design practice with hard-decision memory sensing, fine-grained soft-decision

Manuscript received May 03, 2010; revised June 26, 2010; accepted August 01, 2010. Date of publication November 11, 2010; date of current version January 28, 2011. This work was supported by the National Science Foundation under Grant CCF-0937799. This paper was recommended by Associate Editor M. Laddomada.

G. Dong and T. Zhang are with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute (RPI), Troy, NY 12180-3590 USA (e-mail: dongguiqiang@gmail.com; tong.zhang@ieee.org).

N. Xie is with Intel Corporation, Hillsboro, OR 97124 USA (e-mail: ningdexie@gmail.com).

Digital Object Identifier 10.1109/TCSI.2010.2071990

memory sensing clearly leads to a longer on-chip memory sensing latency and longer Flash-to-controller data transfer latency. In addition, since sensed data should be temporarily stored in an on-chip page buffer, fine-grained memory sensing also results in a larger on-chip page buffer and, hence, higher silicon cost. Therefore, when using these advanced ECCs in future NAND Flash memory, it is critical to minimize the fine-grained sensing precision (i.e., the number of memory sensing levels) while still achieving sufficient error-correction performance. Aside from higher memory sensing precision, accurate calculation of LLRs also demands the availability of a sufficiently accurate memory-cell threshold-voltage statistical distribution model. It has been well recognized that, as the technology continues to scale down, cell-to-cell interference is the increasingly dominant source for memory-cell threshold-voltage distribution distortion [14]–[17]. Hence, it is important to derive a reasonably accurate threshold-voltage statistical distribution model for memory cells in the presence of cell-to-cell interference. However, to the best of our knowledge, the aforementioned critical issues on the use of advanced ECCs in future NAND Flash memory has not been ever addressed in the open literature.

This paper attempts to fill such a missing link. First, based upon NAND Flash memory erase-and-programming characteristics, we derive mathematical formulations to approximately model threshold-voltage distribution of memory cells in the presence of cell-to-cell interference. We further discuss the calculation of LLRs based on such a mathematical model. Second, we present a nonuniform memory sensing strategy to reduce the memory sensing precision while still maintaining good error-correction performance, compared with straightforward uniform memory sensing. The key is to mainly focus the memory sensing around the boundary of adjacent memory-cell storage states where the entropy tends to be relatively high. Lastly, we investigate how to reformulate the memory-cell threshold-voltage distribution model and LLR calculation when we can also sense the threshold-voltage shift of interfering cells and hence explicitly estimate cell-to-cell interference strength. Using a hypothetical 2-bits/cell NAND Flash memory and rate-19/20 LDPC code as a test vehicle, we carry out extensive computer simulations to evaluate the effectiveness of the developed techniques and involved tradeoffs.

The remainder of this paper is organized as follows. Section II reviews the basics of NAND Flash memory and cell-to-cell interference. In Section III, we derive the mathematical formulation for approximately modeling the threshold-voltage distribution of memory cells in the presence of cell-to-cell interference, study the corresponding LLR calculation, and present the nonuniform memory sensing strategy. In Section IV, we reinvestigate these design issues under the scenario when a certain degree of cell-to-cell interference information is known. Conclusions are drawn in Section V.

II. Background

A. Memory-Cell Programming

Each NAND Flash memory cell is a floating-gate transistor whose threshold voltage can be configured (or programmed) by injecting a certain amount of charges into the floating gate. Hence, data storage in a K-level-per-cell NAND Flash memory is realized by programming the threshold voltage of each memory cell into one of K nonoverlapping voltage windows. Before one memory cell can be programmed, it must be erased (i.e., remove the charges in the floating gate, which sets its threshold voltage to the lowest voltage window). Due to inevitable process variability, the threshold voltage of erased memory cells tends to have a wide Gaussian-like distribution [18]. Hence, we model the threshold-voltage distribution of the erased state as

$$p_e(x) = \frac{1}{\sigma_e\sqrt{2\pi}} e^{-\frac{(x-\mu_e)^2}{2\sigma_e^2}} \tag{1}$$

where μ_e and σ_e are the mean and standard deviation of the erased-state threshold voltage.

When memory cells are programmed, a tight threshold-voltage control is typically realized by using incremental-step-pulse program, i.e., all the memory cells on the same word line are recursively programmed using a program-and-verify approach with a staircase program word-line voltage $V_{\rm pp}$ [19], [20]. Let $\Delta V_{\rm pp}$ denote the incremental program step voltage. Under such a program-and-verify programming strategy, each programmed state (except the erased state) associates with a verify voltage that is used in the verify operations. Denote the verify voltage of the kth programmed state as V_p. During each program-and-verify cycle, the floating-gate threshold voltage V_t is first boosted by up to $\Delta V_{\rm pp}$ and then compared with the corresponding verify voltage. If memory-cell threshold voltage is still lower than the verify voltage, the program-and-verify iteration continues; otherwise, the corresponding bit line is configured so that further programming of this cell is disabled. Therefore, the threshold voltage of the kth programmed state tends to have a uniform distribution over $[V_p, V_p + \Delta V_{\rm pp}]$ with the width of $\Delta V_{\rm pp}$ [21]. Denote V_p and $V_p + \Delta V_{\rm pp}$ for the kth programmed state as $V_l^{(k)}$ and $V_r^{(k)}$. We can model the ideal threshold-voltage distribution of the kth programmed state as

$$p_p^{(k)}(x) = \begin{cases} \frac{1}{\Delta V_{\rm pp}}, & \text{if } V_l^{(k)} \le x \le V_r^{(k)} \\ 0, & \text{else.} \end{cases} \tag{2}$$

We note that certain device and circuit noises, particularly random telegraph noise [21], [22], may introduce exponentially decreasing tails at both sides of the uniform distribution. As well discussed in the open literature [14]–[17], cell-to-cell interference may most severely distort the threshold-voltage distribution of programmed cells.

B. Cell-to-Cell Interference

In NAND Flash memory, the threshold-voltage shift of one floating-gate transistor can influence the threshold voltage of its neighboring floating-gate transistors through parasitic capacitance-coupling effect [23]. Such cell-to-cell interference has been well recognized as the major noise source in NAND Flash memory and, hence, the most critical barrier hindering future NAND Flash memory scaling [14]–[17].

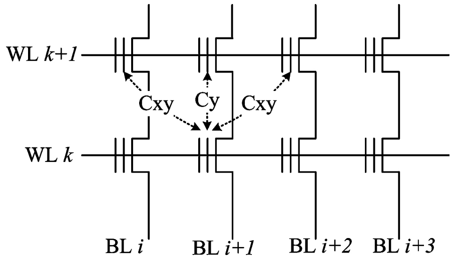


Fig. 1. Illustration of the all-bit-line structure and parasitic coupling capacitances among adjacent cells.

The threshold-voltage shift of a victim cell caused by cell-to-cell interference from neighbor interfering cells which are programmed after this victim cell can be estimated as [23]

$$F = \sum_n \left(\Delta V_t^{(n)} \cdot \gamma^{(n)} \right) \tag{3}$$

where $\Delta V_t^{(n)}$ represents the threshold-voltage shift of one interfering cell which is programmed after the victim cell and the coupling ratio $\gamma^{(n)}$ is defined as

$$\gamma^{(n)} = \frac{C^{(n)}}{C_{\text{total}}} \tag{4}$$

where $C^{(n)}$ is the parasitic capacitance between the interfering cell and the victim cell and C_{total} is the total capacitance of the victim cell.

Cell-to-cell interference significance also depends on NAND Flash memory bit-line structure. In current design practice, there are two different bit-line structures, including conventional even/odd bit-line structure [17], [24] and emerging all-bit-line structure [25], [26]. In even/odd bit-line structure, memory cells on one word line are alternatively connected to even and odd bit lines and they are programmed at different times. An even cell is influenced by five neighboring cells which are programmed after this selected even cell, and an odd cell is influenced by three neighboring cells on the next word line. Therefore, even and odd cells experience largely different amounts of cell-to-cell interference [27]. In the all-bit-line structure, all the memory cells on the same word line are programmed at the same time; therefore, one cell is mainly influenced by three neighboring cells on the adjacent word line, as shown in Fig. 1. Clearly, the all-bit-line structure induces less significant worst-case cell-to-cell interference. In addition, as pointed out in [26], the all-bit-line structure can most effectively support high-speed current sensing to improve the memory read-and-verify speed. Therefore, throughout the remainder of this paper, we mainly consider NAND Flash memory with the all-bit-line structure.

For the cell-to-cell interference modeling in this work, we assume that both the vertical coupling ratio γ_y and diagonal coupling ratio γ_{xy} are random variables with bounded Gaussian distributions

$$p_r(x) = \begin{cases} \frac{c_r}{\sigma_r \sqrt{2\pi}} \cdot e^{-\frac{(x-\mu_r)^2}{2\sigma_r^2}}, & \text{if } |x-\mu_r| \le w_r \\ 0, & \text{else} \end{cases} \tag{5}$$

where μ_r and σ_r are the mean and standard deviation, w_r represents the bounded distribution region, and c_r is chosen to ensure that the integration of this bounded Gaussian distribution equals to one. We set $w_r = 0.1\mu_r$ and $\sigma_r = 0.4\mu_r$ in all the simulations. According to [4], [15], we set the ratio between the means of γ_y and γ_{xy} as 0.08:0.006, i.e., vertical cell-to-cell interference with γ_y tends to play a dominating role. To study a wide range of cell-to-cell coupling strength, we introduce a parameter s called *cell-to-cell coupling strength factor* and have the mean of γ_y equal 0.08 s and the mean of γ_{xy} equal 0.006 s.

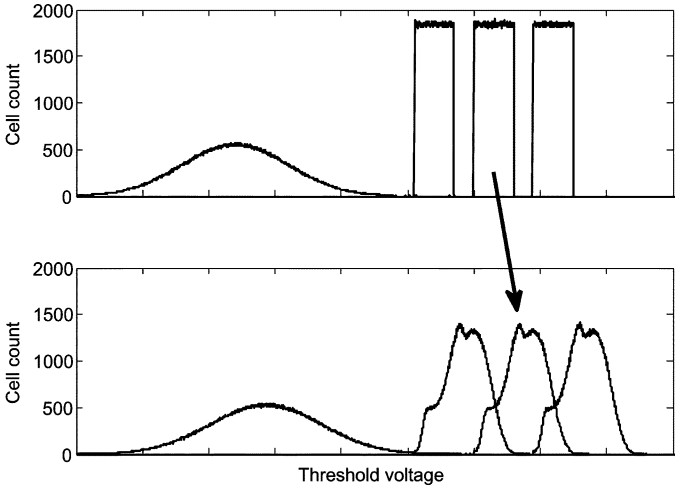


Fig. 2. Simulated threshold-voltage distributions before and after cell-to-cell interference in 2-bits/cell NAND Flash memory when cell-to-cell coupling strength factor s is 1.5.

Example 2.1: Let us consider the 2-bits/cell NAND Flash memory. We set the normalized σ_e and μ_e of the erased state as 0.35 and 1.2, respectively. For the three programmed states, we set the normalized program step voltage ΔV_{pp} as 0.3 and the normalized verify voltages V_p as 2.55, 3.0, and 3.45 V, respectively. The cell-to-cell coupling strength factor s is set as 1.5. We carry out computer simulations to obtain the cell threshold-voltage distribution before and after the occurrence of cell-to-cell interference, as shown in Fig. 2, where the coupling ratios are modeled as bounded Gaussian variables, as discussed previously.

III. Using Soft-Decision ECCs

This paper studies the use of soft-decision ECCs such as LDPC codes in NAND Flash memory. As pointed out earlier, current NAND Flash memories use hard-decision ECCs such as BCH codes, compared with which the use of soft-decision ECCs incurs two critical issues.

1) NAND Flash chips should support fine-grained soft-decision memory-cell threshold-voltage quantization (e.g., the threshold voltage of each cell in 2-bits/cell memory is quantized into 4 bits). This will directly result in a longer on-chip memory sensing latency, larger on-chip page buffer, and longer Flash-to-controller data transfer latency. Suppose each memory cell in 2-bits/cell NAND Flash memory is sensed with an l-bit precision, compared with the hard-decision memory sensing, the on-chip sensing latency approximately increases by $(2^l - 1)/3$ times because of the fully sequential NAND Flash memory sensing operations, the on-chip page buffer size increases by $l/2$ times, and the Flash-to-controller data transfer latency

increases by $l/2$ times. Clearly, it is highly desirable to minimize the memory-cell sensing precision while still maintaining sufficiently good ECC decoding performance.

2) The fine-grained memory-cell sensing results should be translated to LLR for each bit as the input to soft-decision ECC decoders. The calculation of LLR strongly depends on the threshold-voltage statistical distributions of all the states. In most communication and data storage systems, because of the use of sophisticated equalization and noise whitening before ECC decoding, LLR is typically calculated by assuming noise as random Gaussian variables, which can largely simplify the LLR calculation. However, threshold-voltage distribution in NAND Flash memory is not necessarily close to a Gaussian distribution (e.g., see Fig. 2 in Example 2.1). Therefore, we should particularly investigate how to more accurately calculate LLR given the fine-grained memory sensing results. Clearly, better LLR calculation quality can further contribute to the minimization of memory-cell sensing precision.

In this section, we will first derive the mathematical formulation for calculating LLR based upon the NAND Flash threshold-voltage distribution and cell-to-cell interference model presented in Section II, and then, we will present a nonuniform memory sensing strategy that can reduce the memory sensing precision while maintaining good error-correction performance.

A. Mathematical Formulation of LLR Calculation

Let $V_{\rm th}$ represent the sensed threshold voltage of one memory cell, and we simply assume that each bit in a cell has *a priori* probability of 0.5 being 0 or 1, i.e., all the storage states in one memory cell have equal *a priori* probability. Therefore, the LLR of the ith bit stored in one cell is

$$L(b_i) = \log\frac{p(b_i=1|V_{\rm th})}{p(b_i=0|V_{\rm th})} = \log\frac{p(V_{\rm th}|b_i=1)}{p(V_{\rm th}|b_i=0)}. \tag{6}$$

Let N_b represent the number of bits stored in each NAND Flash memory cell; hence, there are $K = 2^{N_b}$ storage states. Let $p^{(k)}(x)$ denote the probability density function of the threshold voltage of the kth storage state, where $0 \leq k \leq K-1$ and the state 0 corresponds to the erased state and state $K-1$ corresponds to the state with the highest threshold voltage. Let $\mathcal{S}_i$ denote the set of states whose ith bit is 1. Hence, given the threshold voltage $V_{\rm th}$ of a cell, we can calculate the LLR of each bit as

$$L(b_i) = \log\frac{\sum\limits_{k\in\mathcal{S}_i} p^{(k)}(V_{\rm th})}{\sum\limits_{k} p^{(k)}(V_{\rm th}) - \sum\limits_{k\in\mathcal{S}_i} p^{(k)}(V_{\rm th})}. \tag{7}$$

Clearly, LLR calculation demands the knowledge of the probability density functions of all the K states. Based on the discussions in Section II, LLR calculation is trivial if cell-to-cell interference has not occurred yet. Hence, we only consider LLR calculations for cells in the presence of cell-to-cell interference. In this context, we should first know the probability density function of cell-to-cell interference, denoted as $p_c(x)$. Suppose a victim cell suffers cell-to-cell interference from N interfering cells. Let x_n and y_n denote the threshold voltage of the nth interfering cell before and after being programmed, and let γ_n denote the coupling ratio between the nth interfering cell and the victim cell. The overall cell-to-cell interference can be expressed as

$$F = \sum_{n=0}^{N-1} \gamma_n(y_n - x_n) = \sum_{n=0}^{N-1} \gamma_n y_n - \sum_{n=0}^{N-1} \gamma_n x_n. \tag{8}$$

As discussed earlier, the threshold voltage of the erased state has a Gaussian distribution. Hence, all the x_n's are random Gaussian variables, and $x' = \sum_{n=0}^{N-1} \gamma_n x_n$ is still a Gaussian variable $\mathcal{N}(\mu', \sigma')$, where

$$\mu' = \sum_{n=0}^{N-1} \gamma_n \mu_e \quad \sigma' = \left(\sum_{n=0}^{N-1} \gamma_n^2 \sigma_e^2\right)^{1/2}. \tag{9}$$

Let $p_{x'}(x)$ denote the probability density function of the Gaussian variable x'. After a cell is programmed, its threshold voltage ideally follows a uniform distribution, as discussed earlier. Hence, each $\gamma_n y_n$ can be simplified as a scaled uniform distribution. Let $y' = \sum_{n=0}^{N-1} \gamma_n y_n$, and assuming that each γ_n is a constant, we can obtain its probability density function $p_{y'}(y)$ using the results presented in [28] that derives a closed form for the probability density function of the sum of n independent nonidentically distributed uniform random variables. Hence, the probability density function of the overall cell-to-cell interference strength $F = y' - x'$ can be estimated as

$$p_c(x) = \int\limits_t p_{y'}(x+t) p_{x'}(t) dt. \tag{10}$$

After obtaining the cell-to-cell interference strength, the last step is to estimate the distribution of victim-cell threshold-voltage shift induced by cell-to-cell interference. If the victim cell stays in the erased state, its threshold-voltage distribution in the presence of cell-to-cell interference can be expressed as

$$p^{(0)}(x) = \int\limits_t p_e(t) p_c(x-t) dt. \tag{11}$$

If the victim cell is programmed to the kth programmed state, its threshold-voltage distribution in the presence of cell-to-cell interference can be expressed as

$$p^{(k)}(x) = \int\limits_t p_p^{(k)}(t) p_c(x-t) dt. \tag{12}$$

In the following, we further elaborate on the scenario when the all-bit-line structure is being used. In this context, as discussed previously, one victim cell has three interfering cells, one along the vertical direction with γ_y and two along the two diagonal directions with γ_{xy}. According to [4] and [15], γ_{xy} is one order of magnitude less than γ_y. Hence, to simplify the following mathematical derivations, we ignore the cell-to-cell interference from the two diagonal directions and also simply fix γ_y as a constant. We note that we still take into account of the cell-to-cell interference from the two diagonal directions and

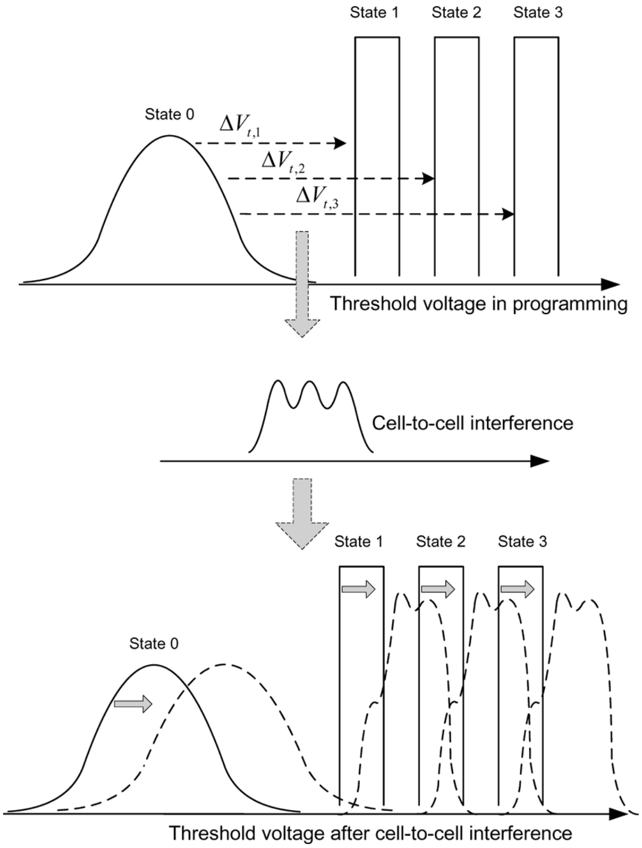

State 1
State 2
State 3
State 0
$\Delta V_{t,1}$
$\Delta V_{t,2}$
$\Delta V_{t,3}$
Threshold voltage in programming
Cell-to-cell interference
State 1
State 2
State 3
State 0
Threshold voltage after cell-to-cell interference

Fig. 3. Using 2-bits/cell memory as an example to illustrate the procedure for estimating victim-cell threshold-voltage distribution in the presence of cell-to-cell interference.

model the coupling ratios as bounded Gaussian variables for all the computer simulations presented throughout this paper.

Let $\Delta V_{t,k}$ represent the threshold-voltage shift when one interfering cell is being programmed to the kth state, and let $p_\Delta^{(k)}(x)$ represent the probability density function of $\Delta V_{t,k}$. As shown in Fig. 3, in the following, we first derive $p_\Delta^{(k)}(x)$ and then $p_c(x)$, based on which we derive $p^{(k)}(x)$ for calculating LLR according to (7).

When the interfering cell is being programmed to the state k (where $1 \leq k \leq K-1$), the probability density function of its threshold-voltage shift can be obtained as

$$\begin{aligned} p_\Delta^{(k)}(x) &= \int_t p_e(t-x)p_p^{(k)}(t)dt \\ &= \int_{V_l^{(k)}}^{V_r^{(k)}} \frac{1}{\Delta V_{\text{pp}}} \frac{1}{\sigma_e\sqrt{2\pi}} e^{-\frac{(t-x-\mu_e)^2}{2\sigma_e^2}} dt \\ &= \frac{1}{2\Delta V_{\text{pp}}}\left(erf\left(\frac{V_r^{(k)}-x-\mu_e}{\sqrt{2}\sigma_e}\right)\right. \\ &\qquad \left. - erf\left(\frac{V_l^{(k)}-x-\mu_e}{\sqrt{2}\sigma_e}\right)\right) \end{aligned} \quad (13)$$

where

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt \quad (14)$$

is the error function. Given the probability density function $p_\Delta^{(k)}(x)$ of threshold voltage shift of one interfering cell and the corresponding coupling ratio γ_y, the cell-to-cell interference experienced by the victim cell can be obtained as

$$p_c^{(k)}(x) = \frac{p_\Delta^{(k)}(x/\gamma_y)}{\gamma_y}. \quad (15)$$

If a cell should stay in the erased state, it will not experience threshold voltage shift from programming operation, and thus will not induce cell-to-cell interference to its neighbors, therefore the corresponding interference can be modeled as

$$p_c^{(0)}(x) = \delta(x) \quad (16)$$

where $\delta(x)$ represents the Dirac delta function.

Assuming that the interfering cells have the same probability to be programmed to each state (i.e., each state has a probability of $1/K$), the overall distribution of cell-to-cell interference can be modeled as

$$\begin{aligned} p_c(x) =& \frac{1}{2K\gamma_y\Delta V_{\text{pp}}} \\ & \times \sum_{v=1}^{K-1}\left(erf\left(\frac{V_r^{(v)}-x/\gamma_y-\mu_e}{\sqrt{2}\sigma_e}\right)\right. \\ & \qquad \left. - erf\left(\frac{V_l^{(v)}-x/\gamma_y-\mu_e}{\sqrt{2}\sigma_e}\right)\right) \\ & \qquad + \frac{\delta(x)}{K}. \end{aligned} \quad (17)$$

We can then estimate the overall victim-cell threshold-voltage distribution after the occurrence of cell-to-cell interference. If the victim cell is in the erased state, its threshold-voltage distribution after the occurrence of cell-to-cell interference can be modeled as

$$\begin{aligned} p^{(0)}(x) &= \int_t p_e(t)p_c(x-t)dt \\ &= \frac{1}{C}\int_t e^{-\frac{(t-\mu_e)^2}{2\sigma_e^2}} \\ &\quad \times \left(\sum_{v=1}^{K-1}\left(erf\left(\frac{V_r^{(v)}-\frac{x-t}{\gamma_y}-\mu_e}{\sqrt{2}\sigma_e}\right)\right.\right. \\ &\qquad \left. -erf\left(\frac{V_l^{(v)}-\frac{x-t}{\gamma_y}-\mu_e}{\sqrt{2}\sigma_e}\right)\right) \\ &\qquad \left. + 2\gamma_y\Delta V_{\text{pp}}\delta(x-t)\right) dt \end{aligned} \quad (18)$$

where $C = 2\sqrt{2\pi}\sigma_e\gamma_y K\Delta V_{\text{pp}}$.

If the victim cell is in the kth programmed state, its threshold-voltage distribution after the occurrence of cell-to-cell interference can be modeled as

$$\begin{aligned} p^{(k)}(x) &= \int_t p_p(t)p_c(x-t)dt \\ &= \int_{V_l^{(k)}}^{V_r^{(k)}} \left(\frac{1}{2K\Delta V_{\rm pp}^2} \right. \\ &\quad \times \sum_{v=1}^{K-1} \left(erf\left(\frac{V_r^{(v)} - \frac{x-t}{\gamma_y} - \mu_e}{\sqrt{2}\sigma_e} \right) \right. \\ &\qquad \left. - erf\left(\frac{V_l^{(v)} - \frac{x-t}{\gamma_y} - \mu_e}{\sqrt{2}\sigma_e} \right) \right) \\ &\quad \left. + \frac{\delta(x-t)}{K} \right) dt. \end{aligned} \tag{19}$$

Ideally, given the previously obtained distribution functions and the sensed threshold voltage $V_{\rm th}$ of memory cells in the presence of cell-to-cell interference, we can use (7) to calculate the corresponding LLR. In practice, threshold-voltage sensing is realized by the comparison with a series of reference voltages. Assume that the threshold voltage $V_{\rm th}$ falls into the range $(R_l, R_r]$ (where R_l and R_r are two adjacent reference voltages), we can estimate the corresponding LLR of the ith bit as

$$L(b_i) = \log \frac{\int_{R_l}^{R_r} \sum_{k \in \mathcal{S}_i} p^{(k)}(x)dx}{\int_{R_l}^{R_r} \sum_k p^{(k)}(x)dx - \int_{R_l}^{R_r} \sum_{k \in \mathcal{S}_i} p^{(k)}(x)dx}. \tag{20}$$

In the aforementioned mathematical formulation, we assume that the erased-state distribution parameters μ_e and σ_e are known and we treat the coupling ratio γ_y as a known constant. As pointed out earlier, γ_y is essentially a random variable; hence, we should use the average value of γ_y in practice. In addition, given all these parameters, we can precalculate all the possible LLRs, and hence, we only need to carry out table lookup in the run time to obtain the LLR. Let K_s denote the number of reference voltages being used in memory sensing, the LLR lookup table only contains $N_b(K_s+1)$ entries for N_b-bit/cell NAND Flash memory.

Example 3.1: Let us consider the use of LDPC code in 2-bits/cell NAND Flash memory with the all-bit-line structure and 4-kB page. We set the normalized σ_e and μ_e of the erase state as 0.35 and 1.2, respectively. For the three programmed states, we set the normalized verify voltages V_p as 2.55, 3.0, and 3.45, respectively, and the normalized program step voltage $\Delta V_{\rm pp}$ as 0.3. We construct a rate-19/20 (34 520, 32 794) quasi-cyclic (QC) LDPC code [29] with column weight 4 and girth 6. Min-sum decoding algorithm is used to carry out LDPC code decoding. We use the standard gray code mapping, i.e., we map "11," "10," "00," and "01" to the states 0, 1, 2, and 3, respectively. We sense the cell threshold voltages with floating point precision and quantify each LLR to 6 bits. For the purpose of comparison, we also consider a rate-19/20 (32 767, 31 133, 109) binary BCH code. Fig. 4 shows the simulated page error rate (PER) versus cell-to-cell coupling strength factor s, where it is observed that the LDPC code offers obvious performance gain over BCH.

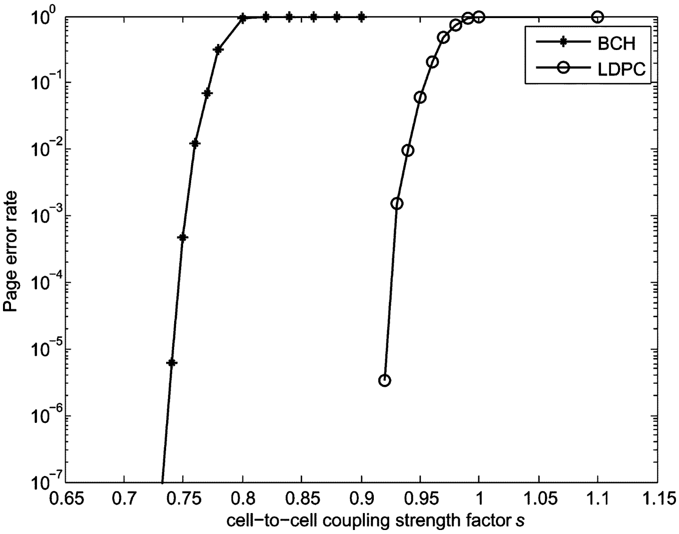


Fig. 4. Simulated PER versus cell-to-cell coupling strength factor s when LDPC and BCH codes are being used.

B. Proposed Nonuniform Memory Sensing

As pointed out earlier, it is highly desirable to reduce the number of memory sensing levels in order to reduce the implementation and latency overhead when soft-decision ECCs are being used. This section investigates the potential of using nonuniform memory sensing to achieve this objective. Conventional design practice tends to simply use a uniform fine-grained soft-decision memory sensing strategy, as shown in Fig. 5, where the soft-decision sensing reference voltages uniformly distribute within each pair of hard-decision reference voltages [30]. Intuitively, since most overlaps between two adjacent states occur around the corresponding hard-decision reference voltage (i.e., the boundary of two adjacent states), as shown in Fig. 5, it may be desirable to sense such region with a higher precision and leave the remainder region with less sensing precision or even no sensing. This naturally leads to a nonuniform memory sensing strategy. Given a sensed threshold voltage $V_{\rm th}$, its entropy can be obtained as

$$H(V_{\rm th}) = \sum_k P({\rm state} = k|V_{\rm th}) \log \frac{1}{P({\rm state} = k|V_{\rm th})} \tag{21}$$

where

$$P({\rm state} = k|V_{\rm th}) = \frac{p^{(k)}(V_{\rm th})}{\sum_v p^{(v)}(V_{\rm th})}. \tag{22}$$

Given the $V_{\rm th}$ of a Flash memory cell, there is always just one or two items being dominating among all the $KP({\rm state} = k|V_{\rm th})$ items for the calculation of $H(V_{\rm th})$. Outside of the dominating overlap region, there is only one dominating item very close to 1 with all the other items being almost 0; hence, the entropy will be very small. On the other hand, within the dominating overlap region, there are two relatively dominating items among all the $KP({\rm state} = k|V_{\rm th})$ items, and both of them are close to 0.5

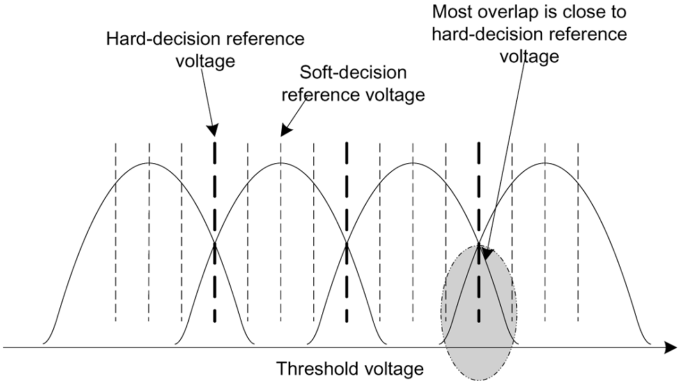

Hard-decision reference voltage
Soft-decision reference voltage
Most overlap is close to hard-decision reference voltage
Threshold voltage

Fig. 5. Illustration of the straightforward uniform soft-decision memory sensing. Note that soft-decision reference voltages are uniformly distributed between any two adjacent hard-decision reference voltages.

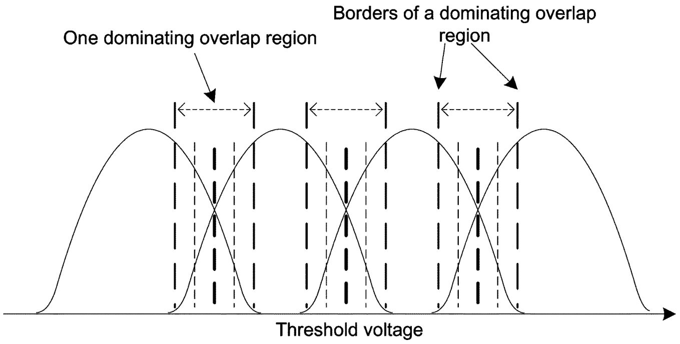

Borders of a dominating overlap region
One dominating overlap region
Threshold voltage

Fig. 6. Illustration of the proposed nonuniform sensing strategy. The dominating overlap region is around hard-decision reference voltage, and all the sensing reference voltages only distribute within those dominating overlap regions.

if $V_{\rm th}$ locates close to the hard-decision reference voltage, i.e., the boundary of two adjacent states, which will result in a relatively large entropy value $H(V_{\rm th})$. Clearly, the region with large entropy tends to demand a higher sensing precision. Therefore, it is intuitive to apply a nonuniform memory sensing strategy, as shown in Fig. 6. Associated with each hard-decision reference voltage at the boundary of two adjacent states, we define a so-called *dominating overlap region* and we carry out uniform memory sensing only within each dominating overlap region.

Selection of the dominating overlap region clearly involves a design tradeoff: If we reduce the size of each dominating overlap region, we can accordingly reduce the memory sensing levels, which nevertheless may result in soft-decision ECC decoding performance degradation. Given the sensed $V_{\rm th}$ of a memory cell, the value of entropy $H(V_{\rm th})$, as in (21), is mainly determined by two largest probability items, and this translates into the ratio between the two largest probability items. Therefore, such a design tradeoff can be adjusted by a probability ratio R, i.e., by letting $[B_l^{(k)}, B_r^{(k)}]$ denote the dominating overlap region between two adjacent states s_k and s_{k+1}, we can determine the border B_l and B_r by solving

$$\frac{p^{(k)}\left(B_l^{(k)}\right)}{p^{(k+1)}\left(B_l^{(k)}\right)} = \frac{p^{(k+1)}\left(B_r^{(k)}\right)}{p^{(k)}\left(B_r^{(k)}\right)} = R. \qquad (23)$$

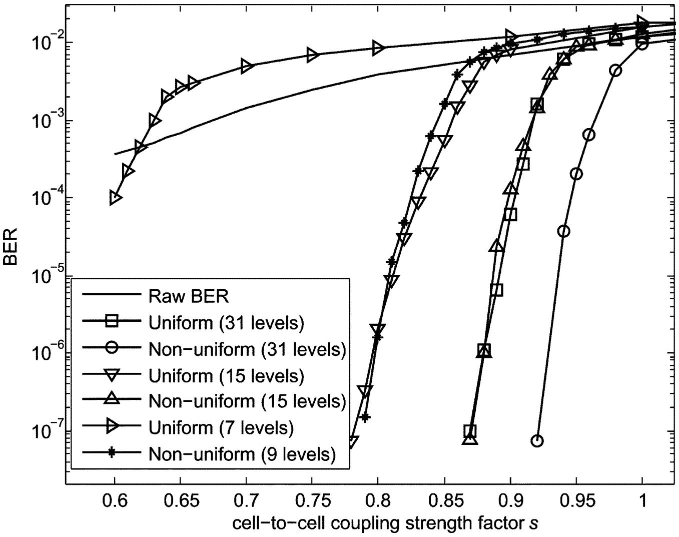


Fig. 7. Performance of LDPC code when using the nonuniform and uniform sensing schemes with various sensing level configurations.

TABLE I
COMPARISON BETWEEN REFERENCE VOLTAGES IN 15-LEVEL UNIFORM AND NONUNIFORM SENSING SCHEMES

Level index	Uniform sensing (V)	Non-uniform sensing (V)
1	1.2	2.5
2	1.537	2.525
3	1.875	2.549
4	2.212	2.665
5	2.549	2.781
6	2.701	2.950
7	2.854	3.055
8	3.007	3.159
9	3.159	3.273
10	3.271	3.386
11	3.382	3.400
12	3.494	3.503
13	3.605	3.605
14	3.751	3.72
15	3.898	3.835

We note that, since each dominating overlap region contains one hard-decision reference voltage and two borders, at least $3(K-1)$ sensing levels should be used in nonuniform sensing.

Example 3.2: Using the same memory parameters and same rate-19/20 (34 520, 32 794) QC-LDPC code as in Example 3.1, we carry out computer simulations to evaluate the previously presented nonuniform memory sensing approach in 2-bits/cell NAND Flash memory, where at least nine nonuniform sensing levels is required. We set the probability ratio R as 512 when determining the dominating overlap region. For the purpose of comparison, we also evaluate the use of conventional uniform sensing scheme. Fig. 7 shows the simulated BER performances of both sensing schemes under various memory sensing precisions. We can observe that 15-level nonuniform sensing provides almost the same performance as 31-level uniform sensing, corresponding to about 50% sensing latency reduction, and nine-level nonuniform sensing performs very closely to 15-level uniform sensing, corresponding to about 40% sensing latency reduction. To further show the difference between uniform and nonuniform sensing, Table I lists the normalized sensing levels for 15-level uniform sensing and nonuniform sensing used in this example.

IV. LLR Estimation With Known Interference Information

As pointed out in Section II-B, cell-to-cell interference is caused by threshold-voltage shift of interfering cells adjacent to the cells being read. In the mathematical formulation presented in Section III, the threshold-voltage shift of each interfering cell is completely unknown, and hence, we simply assume that each interfering cell has an equal probability to stay in any one of the total K states. Intuitively, if we can obtain certain information about the threshold-voltage shift of interfering cells, we should accordingly adjust the victim-cell threshold-voltage distribution model [27], which can lead to more accurate LLR estimation and hence further improve ECC decoding performance. This is conceptually similar to the signal equalization [31]–[33] being widely used in digital communication. In this section, we study the scenarios when memory sensing is also carried out on interfering cells, and hence, a certain degree of cell-to-cell interference information becomes known.

A. Refined Mathematical Formulation for LLR Calculation

As discussed in Section III, since diagonal coupling ratio γ_{xy} is much less than vertical coupling ratio γ_y, we only consider the interference from one cell that is vertically adjacent to the victim cell and programmed after the victim cell. If memory sensing shows that the interfering cell is still in the erased state, cell-to-cell interference does not occur and we can imply use the original cell threshold-voltage distributions shown in (1) and (2) to calculate the LLRs. If memory sensing shows that the interfering cell is programmed and its threshold voltage falls into the range $(R_l', R_r']$, we simply assume that the threshold voltage of the interfering cell uniformly distributes over $(R_l', R_r']$, and its threshold-voltage shift distribution becomes

$$p'_{\Delta}(x) = C_c\left(erf\left(\frac{R_r' - x - \mu_e}{\sqrt{2}\sigma_e}\right) - erf\left(\frac{R_l' - x - \mu}{\sqrt{2}\sigma_e}\right)\right) \tag{24}$$

where $C_c = 1/(2(R_r' - R_l'))$. Hence, the probability density function of the corresponding cell-to-cell interference experienced by the victim cell can be obtained as

$$\begin{aligned} p_c'(x) &= \frac{p'_{\Delta}\left(x/\gamma_y\right)}{\gamma_y} \\ &= C'\left(erf\left(\frac{R_r' - x/\gamma_y - \mu_e}{\sqrt{2}\sigma_e}\right)\right. \\ &\qquad \left. - erf\left(\frac{R_l' - x/\gamma_y - \mu_e}{\sqrt{2}\sigma_e}\right)\right) \end{aligned} \tag{25}$$

where $C' = 1/(2\gamma_y(R_r' - R_l'))$. If the victim cell should stay in the erased state, its threshold-voltage distribution in the presence of cell-to-cell interference can be obtained as

$$\begin{aligned} p_g^{(0)}(x) &= \int_t p_e(t) p_c'\left(\frac{x-t}{\gamma_y}\right) dt \\ &= \int_t \frac{1}{2\sqrt{2\pi}\sigma_e\gamma_y\left(R_r' - R_l'\right)} \exp\left(\frac{-(t-\mu_e)^2}{2\sigma^2}\right) \\ &\quad \times \left(erf\left(\frac{R_r' - \frac{x-t}{\gamma_y} - \mu_e}{\sqrt{2}\sigma_e}\right)\right. \\ &\qquad \left. - erf\left(\frac{R_l' - \frac{x-t}{\gamma_y} - \mu_e}{\sqrt{2}\sigma_e}\right)\right) dx_e. \end{aligned} \tag{26}$$

If the victim cell is in the kth programmed state, its threshold-voltage distribution after the occurrence of cell-to-cell interference can be modeled as

$$\begin{aligned} p_g^{(k)}(x) &= \int_t p_p(t) p_c'\left(\frac{x-t}{\gamma_y}\right) dt \\ &= \int_{V_l^{(i)}}^{V_r^{(i)}} \frac{1}{2\gamma_y \Delta V_{\text{pp}}\left(R_r' - R_l'\right)} \\ &\quad \times \left(erf\left(\frac{R_r' - \frac{x-t}{\gamma_y} - \mu_e}{\sqrt{2}\sigma_e}\right)\right. \\ &\qquad \left. - erf\left(\frac{R_l' - \frac{x-t}{\gamma_y} - \mu_e}{\sqrt{2}\sigma_e}\right)\right) dt. \end{aligned} \tag{27}$$

Suppose memory sensing shows that the threshold voltage of the victim cell falls into the range of $(R_l, R_r]$, the LLR of the ith bit in the victim cell can be estimated as

$$L(b_i) = \log \frac{\int_{R_l}^{R_r} \sum_{k\in\mathcal{S}_i} p_g^{(k)}(v) dv}{\int_{R_l}^{R_r} \sum_{k} p_g^{(k)}(v) dv - \int_{R_l}^{R_r} \sum_{k\in\mathcal{S}_i} p_g^{(k)}(v) dv}. \tag{28}$$

Similar to the discussion in Section III, we can precalculate all the possible LLRs and use table lookup in the run time. Let K_g and K_s denote the number of reference voltages being used for sensing each interfering cell and victim cell, respectively. The LLR lookup table should have $N_b(K_g+1)(K_s+1)$ entries for N_b-bit/cell NAND Flash memory.

To support the previously refined LLR estimation with known interference information, interfering pages should also be read, which nevertheless increases the memory sensing latency, incurs energy consumption overhead in both Flash memory chips and controllers, and increases the load of the chip-to-chip link between Flash memory chips and controllers [27]. Therefore, we should always try to minimize the overall sensing levels. In addition, it is very intuitive that we should employ a multistep progressive memory sensing strategy, i.e., in the first step, we only read the target page, based on which we carry out soft-decision ECC decoding, and only when the decoding fails do we further read the interfering page. In addition, we may progressively

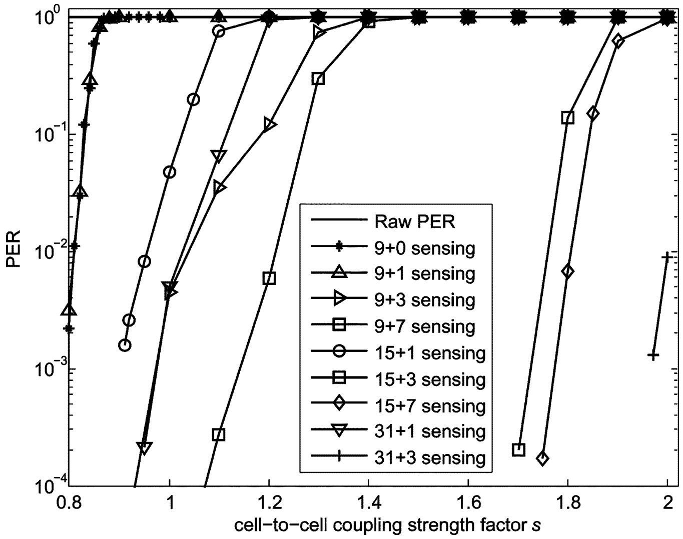


Fig. 8. PER performance of LDPC code under various sensing schemes, where we use nonuniform sensing on the selected page and uniform sensing on the interfering page.

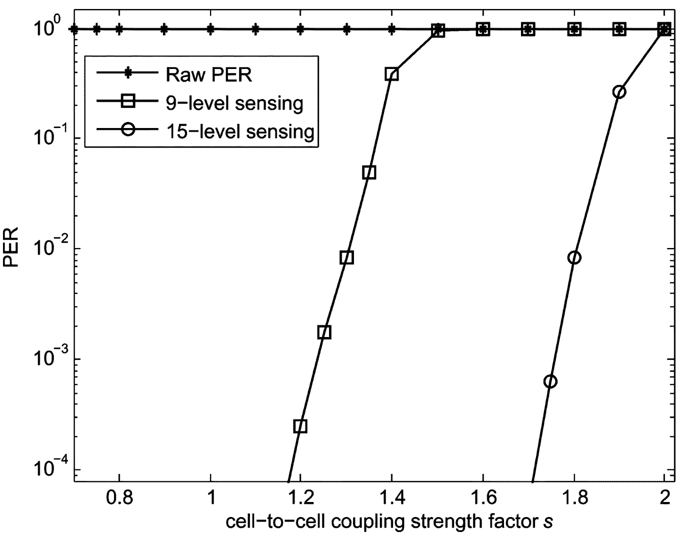


Fig. 9. Simulated PER performance when all the pages are read with nonuniform sensing in the case of consecutive page read.

increase the sensing precision for the target page and/or interfering page in order to further reduce the average-case sensing precision.

Let "$m + n$" sensing represent the memory sensing scheme in which we carry out m-level nonuniform memory sensing for the target page and n-level uniform sensing for its interfering page. Note that we use uniform sensing for the interfering page since it can better reveal the threshold-voltage-shift range of the interfering cell. We use the following example to demonstrate the previous discussions.

Example 4.1: We use the same memory parameters and same rate-19/20 (34 520, 32 794) QC-LDPC code as in Example 3.1. We set the probability ratio R as 512 when determining the dominating overlap region in nonuniform memory sensing. We carry out computer simulations over various sensing configurations, as shown in Fig. 8, where the "9+0" sensing means that we only carry out nine-level nonuniform sensing for the target page without sensing its interfering page at all. First, we note that, the "9+1" sensing, compared with the "9+0" sensing, does not achieve any noticeable decoding performance gain and that "15+1" and "31+1" sensing, although with more sensing levels and, thus, more sensing latency than "9+3" sensing, perform worse than "9+3" sensing.

This suggests that the sensing precision of the interfering page should be appropriately chosen according to the target page sensing precision, in order to achieve a noticeable performance gain. The simulation results shown in Fig. 8 can be used to determine the corresponding multistep progressive sensing configuration. For example, when the coupling strength factor s is 1.2, we may set "9+3" sensing as the first step, which leads to the LDPC code decoding failure rate of 0.12. Once LDPC decoding fails, we can carry out a "9+7" sensing (i.e., we only need to increase the interfering page sensing precision), which can bring the decoding failure rate down to 0.0059, and the last step can be set as "15+7" sensing. This can reduce the average-case sensing latency by 43%, compared with only using the "15+7" sensing configuration.

B. A Special Case: Read of Consecutive Pages

In this paper, we further consider the scenario when a certain number of consecutive pages are being read. In fact, due to the use of garbage collection and wear-leveling at the Flash transaction layer, it is common that NAND Flash memory carries out consecutive page write and read in practice. In addition, the storage of multimedia files such as videos and images clearly tends to incur consecutive page read. Clearly, when consecutive pages are being read, information on the interfering pages become inherently available; hence, we can capture the cell-to-cell interference on the fly during the read operations.

In the previous discussions in Section IV-A, the interfering page is read with uniform sensing and is only used to help the decoding of the victim page. In the case of consecutive page read, all the pages are read with nonuniform sensing and each page is the interfering page to the previous page and, meanwhile, is the victim page of the next page. Clearly, we can still employ a multistep progressive sensing strategy, i.e., we employ nonuniform sensing with less precision on all the pages in the first step, and finer grained sensing will be carried out only for those pages (and possibly their interfering pages) when ECC decoding fails.

Example 4.2: Using the same system configurations as in aforementioned examples, we carry out simulations of multistep progressive sensing strategy where pages are read and decoded using nine-level nonuniform sensing scheme first and then 15-level nonuniform sensing scheme if the first LDPC decoding fails. Fig. 9 shows the simulated PER performance under a range of cell-to-cell coupling strength factor s. These simulation results can be used to determine the corresponding multistep progressive sensing configuration. For example, nine-level sensing should be accordingly set as the first step if $s \leq 1.5$ and 15-level sensing as the first step if $1.5 < s \leq 2.0$. When s is 1.2, nine-level sensing could reduce the LDPC decoding failure rate to 2.5×10^{-4}, compared with the failure rate of 0.12 from "9+3" sensing as the first step in the scenario considered in Section IV-A, and as a result, the sensing latency can be further reduced by more than 27%.

Finally, we note that pages are generally programmed and read both in the same order, i.e., a page with lower index is programmed and read prior to a page with higher index. As a result, one victim page is read before its interfering page is read; hence, it will induce extra read latency if we need to estimate cell-to-cell interference strength in order to successfully decode the victim page. To eliminate this extra read latency, we can use a simple *reverse programming* scheme. The basic idea is very intuitive: We simply reverse the order of programming pages to the descending order, i.e., pages with lower index are programmed later; meanwhile, we still read pages in ascending order. Clearly, when we read those consecutive pages, after one page is read, it can naturally serve to compensate cell-to-cell interference for the page being read next.

V. Conclusion

In this paper, we are concerned with the practical use of soft-decision ECCs in future NAND Flash memory. The decoding of these ECCs requires soft-decision LLR information, which demands fine-grained soft-decision memory sensing and the availability of a memory-cell threshold-voltage distribution model. Since fine-grained memory-cell sensing tends to incur significant implementation overhead and access latency penalty, it is critical to minimize the fine-grained memory sensing precision. In addition, since cell-to-cell interference dominantly contributes to memory-cell threshold-voltage distribution distortion, it must be carefully incorporated into the memory-cell threshold-voltage distribution model. In this paper, we have derived mathematical formulations to approximately model the threshold-voltage distribution of memory cells in the presence of cell-to-cell interference and discussed the corresponding LLR calculation. Moreover, we propose a nonuniform memory sensing strategy that can effectively reduce the memory sensing precision and, in the meantime, still maintain good error-correction performance. Since cell-to-cell interference can be explicitly estimated by reading interfering memory cells, we further consider the scenario when interfering cells can also be read and accordingly investigate the LLR calculation and memory sensing issues.

References

[1] G. Marotta, A. Macerola, A. D'Alessandro, A. Torsi, C. Cerafogli, C. Lattaro, C. Musilli, D. Rivers, E. Sirizotti, F. Paolini, G. Imondi, G. Naso, G. Santin, L. Botticchio, L. De Santis, L. Pilolli, M. L. Gallese, M. Incarnati, M. Tiburzi, P. Conenna, S. Perugini, V. Moschiano, W. Di Francesco, M. Goldman, C. Haid, D. Di Cicco, D. Orlandi, F. Rori, M. Rossini, T. Vali, R. Ghodsi, and F. Roohparvar, "A 3 bit/cell 32 Gb NAND flash memory at 34 nm with 6 MB/s program throughput and with dynamic 2 b/cell blocks configuration mode for a program throughput increase up to 13 MB/s," in *Proc. IEEE Int. Solid-State Circuits Conf.*, 2010, pp. 444–445.

[2] T. Futatsuyama, N. Fujita, N. Tokiwa, Y. Shindo, T. Edahiro, T. Kamei, H. Nasu, M. Iwai, K. Kato, Y. Fukuda, N. Kanagawa, N. Abiko, M. Matsumoto, T. Himeno, T. Hashimoto, Y.-C. Liu, H. Chibvongodze, T. Hori, M. Sakai, H. Ding, Y. Takeuchi, H. Shiga, N. Kajimura, Y. Kajitani, K. Sakurai, K. Yanagidaira, T. Suzuki, Y. Namiki, T. Fujimura, M. Mui, H. Nguyen, S. Lee, A. Mak, J. Lutze, T. Maruyama, T. Watanabe, T. Hara, and S. Ohshima, "A 113 mm^2 32 Gb 3 b/cell NAND flash memory," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2009, pp. 242–243.

[3] S. Lee, Y. Fong, F. Pan, T.-C. Kuo, J. Park, T. Samaddar, H. T. Nguyen, L. Mui, K. Htoo, T. Kamei, M. Higashitani, E. Yero, G. Kwon, P. Kliza, J. Wan, T. Kaneko, H. Maejima, H. Shiga, M. Hamada, N. Fujita, K. Kanebako, E. Tam, A. Koh, I. Lu, C.-H. Kuo, T. Pham, J. Huynh, Q. Nguyen, H. Chibvongodze, M. Watanabe, K. Oowada, G. Shah, B. Woo, R. Gao, J. Chan, J. Lan, P. Hong, L. Peng, D. Das, D. Ghosh, V. Kalluru, S. Kulkarni, R.-A. Cernea, S. Huynh, D. Pantelakis, C.-M. Wang, and K. Quader, "A 16 Gb 3-bit per cell (X3) NAND flash memory on 56 nm technology with 8 MB/s write rate," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 195–207, Jan. 2009.

[4] H. Maejima, K. Isobe, K. Iwasa, M. Nakagawa, M. Fujiu, T. Shimizu, M. Honma, S. Hoshi, T. Kawaai, K. Kanebako, S. Yoshikawa, H. Tabata, A. Inoue, T. Takahashi, T. Shano, Y. Komatsu, K. Nagaba, M. Kosakai, N. Motohashi, K. Kanazawa, K. Imamiya, H. Nakai, M. Lasser, M. Murin, A. Meir, A. Eyal, and M. Shlick, "A 70 nm 16 Gb 16-Level-Cell NAND flash memory," *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 929–937, Apr. 2008.

[5] N. Shibata, T. Nakano, M. Ogawa, J. Sato, Y. Takeyama, K. Isobe, B. Le, F. Mooga, N. Mokhlesi, K. Kozakai, P. Hong, T. Kamei, K. Iwasa, J. Nakai, T. Shimizu, M. Honma, S. Sakai, T. Kawaai, S. Hoshi, J. Yuh, C. Hsu, T. Tseng, J. Li, J. Hu, M. Liu, S. Khalid, J. Chen, M. Watanabe, H. Lin, J. Yang, K. McKay, K. Nguyen, T. Pham, Y. Matsuda, K. Nakamura, K. Kanebako, S. Yoshikawa, W. Igarashi, A. Inoue, T. Takahashi, Y. Komatsu, C. Suzuki, K. Kanazawa, M. Higashitani, S. Lee, T. Murai, K. Nguyen, J. Lan, S. Huynh, M. Murin, M. Shlick, M. Lasser, R. Cernea, M. Mofidi, K. Schuegraf, and K. Quader, "A 5.6 MB/s 64 Gb 4 b/Cell NAND flash memory in 43 nm CMOS," in *Proc. IEEE Int. Solid-State Circuits Conf.*, Feb. 2009, pp. 246–247.

[6] R. E. Blahut, *Algebraic Codes for Data Transmission*. Cambridge, U.K.: Cambridge Univ. Press, 2003.

[7] S. Li and T. Zhang, "Improving multi-level NAND flash memory storage reliability using concatenated BCH-TCM coding," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 18, no. 10, pp. 1412–1420, Oct. 2010.

[8] R. G. Gallager, "Low-density parity-check codes," *IRE Trans. Inf. Theory*, vol. IT-8, no. 1, pp. 21–28, Jan. 1962.

[9] D. J. C. MacKay, "Good error-correcting codes based on very sparse matrices," *IEEE Trans. Inf. Theory*, vol. 45, no. 2, pp. 399–431, Mar. 1999.

[10] V. Guruswami and M. Sudan, "Improved decoding of Reed–Solomon and algebraic-geometry codes," *IEEE Trans. Inf. Theory*, vol. 45, no. 6, pp. 1757–1767, Sep. 1999.

[11] R. Koetter and A. Vardy, "Algebraic soft-decision decoding of Reed–Solomon codes," *IEEE Trans. Inf. Theory*, vol. 49, no. 11, pp. 2809–2825, Nov. 2003.

[12] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo-codes," in *Proc. ICC*, Geneve, Switzerland, May 1993, pp. 1064–1070.

[13] LSI Delivers Industrys First 40 nm Read Channel to Hard Disk Drive Manufacturers [Online]. Available: http://www.lsi.com/news/product_news/2009/2009_06_23.html Jun. 2009

[14] K. Kim, "Future memory technology: Challenges and opportunities," in *Proc. Int. Symp. VLSI Technol., Syst., Appl.*, Apr. 2008, pp. 5–9.

[15] K. Prall, "Scaling non-volatile memory below 30 nm," in *Proc. IEEE 2nd Non-Volatile Semicond. Memory Workshop*, Aug. 2007, pp. 5–10.

[16] H. Liu, S. Groothuis, C. Mouli, J. Li, K. Parat, and T. Krishnamohan, "3D simulation study of cell-cell interference in advanced NAND flash memory," in *Proc. IEEE Workshop Microelectron. Electron Devices*, Apr. 2009, pp. 1–3.

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

[18] K. Takeuchi, T. Tanaka, and H. Nakamura, "A double-level- $\mathrm{V_{th}}$ select gate array architecture for multilevel NAND flash memories," *IEEE J. Solid-State Circuits*, vol. 31, no. 4, pp. 602–609, Apr. 1996.

[19] K.-D. Suh, B.-H. Suh, Y.-H. Lim, J.-K. Kim, Y.-J. Choi, Y.-N. Koh, S.-S. Lee, S.-C. Kwon, B.-S. Choi, J.-S. Yum, J.-H. Choi, J.-R. Kim, and H.-K. Lim, "A 3.3 V 32 Mb NAND flash memory with incremental step pulse programming scheme," *IEEE J. Solid-State Circuits*, vol. 30, no. 11, pp. 1149–1156, Nov. 1995.

[20] R. Bez, E. Camerlenghi, A. Modelli, and A. Visconti, "Introduction to flash memory," *Proc. IEEE*, vol. 91, no. 4, pp. 489–502, Apr. 2003.

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

[22] A. Ghetti, C. M. Compagnoni, F. Biancardi, A. L. Lacaita, S. Beltrami, L. Chiavarone, A. S. Spinelli, and A. Visconti, "Scaling trends for random telegraph noise in deca-nanometer flash memories," in *IEDM Tech. Dig.*, 2008, pp. 1–4.

[23] J.-D. Lee, S.-H. Hur, and J.-D. Choi, "Effects of floating-gate interference on NAND flash memory cell operation," *IEEE Electron Device Lett.*, vol. 23, no. 5, pp. 264–266, May 2002.

[24] Y. Kameda, S. Fujimura, H. Otake, K. Hosono, H. Shiga, Y. Watanabe, T. Futatsuyama, Y. Shindo, M. Kojima, M. Iwai, M. Shirakawa, M. Ichige, K. Hatakeyama, S. Tanaka, T. Kamei, J.-Y. Fu, A. Cernea, Y. Li, M. Higashitani, G. Hemink, S. Sato, K. Oowada, S.-C. Lee, N. Hayashida, J. Wan, J. Lutze, S. Tsao, M. Mofidi, K. Sakurai, N. Tokiwa, H. Waki, Y. Nozawa, K. Kanazawa, and S. Ohshima, "A 56-nm CMOS $99 - \mathrm{m}^2$ 8-Gb multi-level NAND flash memory with 10-MB/s program throughput," *IEEE J. Solid-State Circuits*, vol. 42, no. 1, pp. 219–232, Jan. 2007.

[25] S. Lee, Y. Fong, F. Pan, T.-C. Kuo, J. Park, T. Samaddar, H. Nguyen, M. Mui, K. Htoo, T. Kamei, M. Higashitani, E. Yero, G. Kwon, P. Kliza, J. Wan, T. Kaneko, H. Maejima, H. Shiga, M. Hamada, N. Fujita, K. Kanebako, A. Koh, I. Lu, C. Kuo, T. Pham, J. Huynh, Q. Nguyen, H. Chibvongodze, M. Watanabe, K. Oowada, G. Shah, B. Woo, R. Gao, J. Chan, J. Lan, P. Hong, L. Peng, D. Das, D. Ghosh, V. Kalluru, S. Kulkarni, R. Cernea, S. Huynh, D. Pantelakis, C.-M. Wang, and K. Quader, "A 16 Gb 3 b/cell NAND flash memory in 56 nm with 8 MB/s write rate," in *Proc. IEEE ISSCC*, Feb. 2008, pp. 506–632.

[26] L. Pham, F. Moogat, S. Chan, B. Le, Y. Li, S. Tsao, T.-Y. Tseng, K. Nguyen, J. Li, J. Hu, J.H. Yuh, C. Hsu, F. Zhang, T. Kamei, H. Nasu, P. Kliza, K. Htoo, J. Lutze, Y. Dong, M. Higashitani, J. Yang, H.-S. Lin, V. Sakhamuri, A. Li, F. Pan, S. Yadala, S. Taigor, K. Pradhan, J. Lan, J. Chan, T. Abe, Y. Fukuda, H. Mukai, K. Kawakami, C. Liang, T. Ip, S.-F. Chang, J. Lakshmipathi, S. Huynh, D. Pantelakis, M. Mofidi, and K. Quader, "A 34 MB/s MLC write throughput 16 Gb NAND with all bit line architecture on 56 nm technology," *IEEE J. Solid-State Circuits*, vol. 44, no. 1, pp. 186–194, Jan. 2009.

[27] G. Dong, S. Li, and T. Zhang, "Using data postcompensation and predistortion to tolerate cell-to-cell interference in MLC NAND flash memory," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 57, no. 10, pp. 2718–2728, Oct. 2010.

[28] S. M. Sadooghi-Alvandi, A. R. Nematollahi, and R. Habibi, "On the distribution of the sum of independent uniform random variables," *Stat. Papers*, vol. 50, no. 1, pp. 171–174, Jan. 2009.

[29] H. Zhong and T. Zhang, "Block-LDPC: A practical LDPC coding system design approach," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 52, no. 4, pp. 766–775, Apr. 2005.

[30] I. Alrod and M. Lasser, "Fast, low-power reading of data in a flash memory," U.S. Patent 2 009 031 987 2A1, Dec. 24, 2009.

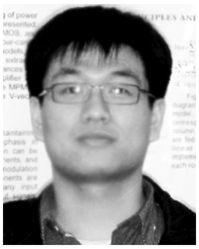
[31] E. A. Lee and D. G. Messerschmidt, *Digital Communication*. Norwell, MA: Kluwer, 1994.

[32] J. Chen, Y. Gu, and K. K. Parhi, "Novel FEXT cancellation and equalization for high speed ethernet transmission," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 56, no. 6, pp. 906–912, Jun. 2009.

[33] T. M. Hollis, D. J. Comer, and D. T. Comer, "Mitigating ISI through self-calibrating continuous-time equalization," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 53, no. 10, pp. 2234–2245, Oct. 2006.

Guiqiang Dong (S'09) received the B.S. and M.S. degrees from the University of Science and Technology of China, Hefei, China, in 2004 and 2008, respectively. He is currently working toward the Ph.D. degree in the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY.

His research interests include coding theory, signal processing for data storage systems, and system design for various digital memory.



Ningde Xie received the B.S. and M.S. degrees in radio engineering from Southeast University, Nanjing, China, in 2004 and 2006, respectively, and the Ph.D. degree in the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY, in 2010.

He has been with the Storage Technology Group, Intel Corporation, Hillsboro, OR, since 2010. His research interests include system and architecture design for high-performance low-power communication and storage systems. Currently, he is working on error control coding and signal processing in NAND-based solid-state drives as well as system design in phase-change memories.

Tong Zhang (M'02–SM'08) received the B.S. and M.S. degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1995 and 1998, respectively, and the Ph.D. degree in electrical engineering from the University of Minnesota, Minneapolis, in 2002.

He is currently an Associate Professor with the Department of Electrical, Computer, and Systems Engineering, Rensselaer Polytechnic Institute, Troy, NY. His research activities span over circuits and systems for various data storage and computing applications.

Dr. Zhang currently serves as an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II and the IEEE TRANSACTIONS ON SIGNAL PROCESSING.