TOWARDS MINIMIZING READ TIME FOR NAND FLASH

Borja Peleato, Rajiv Agarwal, John Cioffi

Stanford University
Stanford CA 94305-9510

Minghai Qin, Paul H. Siegel

University of California, San Diego
La Jolla, CA 92093

ABSTRACT

On NAND flash, a primary source of increased read time comes from the fact that in the presence of noise, the flash medium must be read several times using different read threshold voltages to find the optimal read location, which minimizes bit-error-rate. This paper proposes an algorithm to estimate the optimal read threshold in a fast manner using a limited number of re-reads. Then it derives an expression for the resulting BER in terms of the minimum possible BER. It is also shown that minimizing BER and minimizing codeword-error-rate are competing objectives in the presence of a limited number of allowed re-reads, and a trade-off between the two is proposed.

1. INTRODUCTION

Recently, flash memories have emerged as a faster and more efficient alternative to hard disk drives. The introduction of Solid State Drives (SSD) based on flash NAND memories has revolutionized mobile storage, laptop storage architecture, and enterprise storage, among others. Flash memories offer random access to the information with dramatically higher read throughput because of reduced read time. The reduced read time also increases power-efficiency in SSD's.

However, SSD's are considerably more expensive as compared to hard disk drives. To reduce cost, NAND flash manufacturers race to reduce cell-size and to pack more data in the same silicon, thus reducing cost in $/GB and making flash more attractive to consumers. This cell-size shrinkage comes at the cost of reduced performance. As cell-size shrinks to sub 19nm limits, there is significant noise in the voltage levels stored on the cells, resulting in very noisy read-back. Even in current state-of-the-art 21nm NAND, noise is significant towards the end of life of the drive. One way to recover host data in the presence of noise is to use advanced signal processing algorithms [?][?][?]. This threatens the fast read advantage of flash due to excessive post-read signal processing that is both time- and power-intensive.

During read-back of host data in the presence of noise, the voltage residing on the cell could be significantly different from the voltage that was intended to be stored at the time of write. Therefore, at the time of read, the default read thresholds, which are good for intended voltage levels during write, are no longer valid. Furthermore, the shift of voltages is random and depends on several factors such as time, data, and temperature; so a fixed set of read thresholds will not be optimal throughout the entire life of the drive. Thus, finding optimal read thresholds in a dynamic manner to minimize bit error rate (BER) is essential. Typically, all post-read signal processing algorithms require re-reads using different values of read thresholds. These re-reads increase total read time.

This paper proposes algorithms to estimate the optimal read thresholds in a fast manner, thus reducing total read time. It also presents an analytical expression that relates the BER found using the proposed methods to the minimum possible BER. Though BER is a useful metric for error-correction codes, the distribution of the number of errors is also important. Specifically, in NAND flash with layered coding approaches, a weaker decoder is used when the number of errors is small and a stronger decoder is used otherwise, both for the same code (e.g. hard-LDPC and soft-LDPC for decoding an LDPC code). The total power consumption depends on how frequently each decoder is used. Therefore the distribution of the number of errors is a useful tool to find NAND power consumption, and is derived here. Finally, the decoding time is also an important component of total read time. Specifically for LDPC codes, the quality of the input LLR values dictates the number of iterations required. Therefore, the proposed algorithm to minimize BER by finding good read thresholds is modified to improve the quality of input LLR values. Simulation results are provided to evaluate the performance of the proposed algorithms.

2. SYSTEM MODEL

In a b-bit MLC flash, each cell stores one of 2^b distinct predefined voltage levels. For the sake of simplicity, the rest of the paper will assume a SLC architecture for the flash, where $b = 1$. However, all the methods and results can be readily extended to the MLC case.

Cells in a NAND flash are organized in terms of pages, which are the smallest units for write and read operations. The perturbance in cell voltages during page programming (writing) is commonly known as write noise. Additionally, there exists inter-cell interference (ICI). As the technology scales and more cells are packed in the same area, the interaction between them increases. Parasitic capacitances can therefore cause some cells to increase their voltage level. Both write noise and ICI increase the charge stored in the cells and tend to be larger for cells programmed to the lower levels. When a significant amount of time passes between the writing and reading of a page, cells suffer voltage leakage. This effect is more pronounced for the higher levels.

Reading the cells in a page is done by comparing their stored voltage with a threshold voltage t. The read operation returns a binary vector with one bit for each cell. Bits corresponding to cells with voltage lower than t are 1 and those corresponding to cells with voltage higher than t are 0. However, the aforementioned sources of voltage disturbance can cause the cells to be misclassified, making the reads noisy. The choice of a read threshold therefore becomes important to minimize the BER in the reads. Some attempts have been made to model these sources of noise as a function of time, voltage levels, amplitude of the programming pulses, etc. Unfortunately, the noise is temperature- and page-dependent [**?**] as well as time- and data-dependent. Since the controller cannot measure those factors, it cannot accurately estimate the noise without performing additional reads. This paper assumes that the overall noise follows a Gaussian distribution for each level, but assumes no prior knowledge about their means or variances.

Fig. 1 (top) shows two overlapping Gaussian probability density functions (pdfs), corresponding to the two voltage levels to which cells can be programmed. Since data is generally compressed before being written onto flash, approximately the same number of cells is programmed to each of the levels. Fig. 1 also shows three possible read thresholds. Denoting by (μ_1, σ_1) and (μ_2, σ_2) the means and standard deviations of the two Gaussian distributions, the thresholds are: $t_{\text{mean}} = \frac{\mu_1+\mu_2}{2}$, $t_{\text{median}} = \frac{\mu_1\sigma_2+\mu_2\sigma_1}{\sigma_1+\sigma_2}$ and $t^\star$, which minimizes BER. If the noise variance was the same for both levels, all three thresholds would be equal, but this is not the case in practice. The legend presents the BER resulting from each of the three thresholds. It can be observed that the threshold that returns an equal number of zeros and ones, t_{median}, is suboptimal.

There exist several ways in which the optimal threshold, $t^\star$, can be found[1]. The approach taken in this paper consists of estimating (μ_1, σ_1) and (μ_2, σ_2) and deriving $t^\star$ analytically. It will be shown how this can be done with only four reads, thereby reducing read time. Furthermore, the mean and variance estimates can also be used for other tasks, such as generating soft information for LDPC decoding.

A read operation with a threshold voltage t returns a binary vector with a 1 for each cell whose voltage level is lower

[1]A common approach is to perform several reads by shifting the thresholds in one direction until the decoding succeeds. Once the data has been recovered, it can be compared with the read outputs to find the best threshold. However, this method requires a large number of reads, which reduces read throughput.

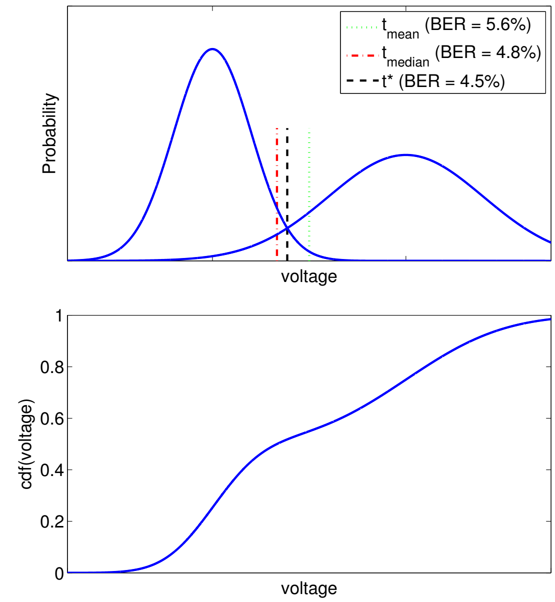

Fig. 1. Top: BER for different thresholds separating two levels. $t_{\text{mean}} = \frac{\mu_0+\mu_1}{2}$ is the average of the cell voltages, t_{median} returns the same number of 1s and 0s and $t^\star$ minimizes the BER and is located at the intersection of the two pdfs. Bottom: cdf corresponding to top pdf.

than t. The probability of a randomly chosen cell having a voltage level below t is then equal to the fraction of ones in the read output. A read with a threshold voltage t can therefore be used to obtain a sample from the cumulative distribution function (cdf) of cell voltages at t.

The problem is then reduced to estimating the means and variances of a mixture of Gaussians using samples from their joint cdf. These samples will be corrupted by model, read and quantization noise. The model noise is caused by the deviation of the actual distribution of cell voltages from a Gaussian distribution. The read noise is caused by the intrinsic reading mechanism of the flash, which can read some cells as having higher or lower voltage than they actually have. The quantization noise is caused by limited computational accuracy and rounding of the Gaussian cdf[2]. All these sources of noise are collectively referred to as read noise in this paper. It is assumed to be zero mean. No other restriction is imposed on the read noise in our derivations.

It is desirable to devote as few reads as possible to the estimation of (μ_1, σ_1) and (μ_2, σ_2). The accuracy of the estimates would increase with the number of reads, but read time would also increase. Since there are four parameters to be estimated, at least four reads will be necessary.

[2]Since the Gaussian cdf has no analytical expression, it is generally quantized and stored as a lookup table

3. MINIMIZING BER

Let t_i, $i = 1, \ldots, 4$ be four voltage thresholds used for reading a page and let y_i, $i = 1, \ldots, 4$ be the fraction of ones in the output vector for each of the reads, respectively. If (μ_1, σ_1) and (μ_2, σ_2) denote the voltage mean and variance for the cells programmed to the lower and upper level, respectively, then

$$y_i = \frac{1}{2}Q\left(\frac{\mu_1 - t_i}{\sigma_1}\right) + \frac{1}{2}Q\left(\frac{\mu_2 - t_i}{\sigma_2}\right) + n_{y_i}, \quad (1)$$

$i = 1, \ldots, 4$, where

$$Q(x) = \int_x^\infty (2\pi)^{-\frac{1}{2}} e^{-\frac{t^2}{2}} dt \quad (2)$$

and n_{y_i} denotes the zero-mean read noise associated to y_i. In theory, it is possible to estimate (μ_1, σ_1) and (μ_2, σ_2) from (t_i, y_i), $i = 1, \ldots, 4$ by solving the system of non-linear equations in Eq. (1). However, the computational complexity is too large and would affect the read speed. This section proposes and evaluates a progressive read algorithm to find the estimators $(\hat{\mu_1}, \hat{\sigma_1})$ and $(\hat{\mu_2}, \hat{\sigma_2})$.

Progressive Read Algorithm: The key idea is to perform two reads at locations where one of the Q functions is known to be either close to 0 or close to 1. The problem with Eq. (1) was that a sum of Q functions cannot be easily inverted. However, once one of the two Q functions in Eq. (1) is fixed at 0 or 1, the other one can be inverted using a standard normal table to put the equation in linear form. The system of linear equations can then be solved to estimate the first mean and variance. These estimates can then be used to eliminate a Q function from each of the remaining two equations, which can then be solved using the same procedure as used for the first two. For example, if t_1 and t_2 are significantly smaller than μ_2, then $Q((\mu_2 - t_1)/\sigma_2) \simeq 0 \simeq Q((\mu_2 - t_2)/\sigma_2)$ and Eq. (1) can be solved for $\hat{\mu_1}$ and $\hat{\sigma_1}$ to get

$$\hat{\sigma_1} = \frac{t_2 - t_1}{Q^{-1}(2y_1) - Q^{-1}(2y_2)} \quad (3)$$

$$\hat{\mu_1} = t_2 + \hat{\sigma_1} Q^{-1}(2y_2). \quad (4)$$

Substituting these in the equations for the third and fourth reads and solving gives

$$\hat{\sigma_2} = \frac{t_4 - t_3}{Q^{-1}(2y_3 - q_3) - Q^{-1}(2y_4 - q_4)} \quad (5)$$

$$\hat{\mu_2} = t_4 + \hat{\sigma_2} Q^{-1}(2y_4 - q_4), \quad (6)$$

where $q_3 = Q((\hat{\mu_1} - t_3)/\hat{\sigma_1})$ and $q_4 = Q((\hat{\mu_1} - t_4)/\hat{\sigma_1})$.

It could be argued that, since the pdfs are not known a priori, it is not possible to determine two read locations where one of the Q functions is close to 0 or close to 1. In practice, however, each read threshold can be chosen based on the result from the previous ones. For example, say the first randomly chosen read location returned $y_1 = 0.6$. This read, if used for estimating the higher level distribution, will be a bad choice because there will be significant overlap from the lower level. Hence, a smart choice would be to obtain two reads for the lower level that are clear of the higher level by reading to the far left of t_1. Once the lower level is canceled, the $y_1 = 0.6$ read can be used in combination with a fourth read to the right of t_1 to estimate the higher level distribution.

Once the mean and variance of both pdfs have been estimated, it is possible to provide an estimate of the read threshold minimizing the BER. The BER associated to a given read threshold t is given by

$$\text{BER}(t) = \frac{1}{2}\left(Q\left(\frac{\mu_2 - t}{\sigma_2}\right) + 1 - Q\left(\frac{\mu_1 - t}{\sigma_1}\right)\right). \quad (7)$$

Making its derivative equal to zero gives the following equation for the optimal threshold $t^\star$

$$\frac{1}{\sigma_2}G\left(\frac{\mu_2 - t^\star}{\sigma_2}\right) = \frac{1}{\sigma_1}G\left(\frac{\mu_1 - t^\star}{\sigma_1}\right), \quad (8)$$

where $G(x) = (2\pi)^{-(1/2)} e^{-x^2/2}$. The optimal threshold $t^\star$ is located at the point where both Gaussian pdfs intersect. An estimate $\hat{t^\star}$ for $t^\star$ can be found from the following quadratic equation

$$2\log\left(\frac{\hat{\sigma_2}}{\hat{\sigma_1}}\right) = \left(\frac{\hat{t^\star} - \hat{\mu_1}}{\hat{\sigma_1}}\right)^2 - \left(\frac{\hat{t^\star} - \hat{\mu_2}}{\hat{\sigma_2}}\right)^2, \quad (9)$$

which can be shown to be equivalent to solving Eq. (8) with (μ_1, σ_1) and (μ_2, σ_2) replaced by their estimated values.

4. ERROR PROPAGATION

This section first studies how the choice of read locations affects the accuracy of the estimators $(\hat{\mu_1}, \hat{\sigma_1})$, $(\hat{\mu_2}, \hat{\sigma_2})$, and correspondingly $\hat{t^\star}$. Then it analyzes how the accuracy of $\hat{t^\star}$ translates into BER$(\hat{t^\star})$. Using this it provides some guidelines as to how the read locations should be chosen. Without loss of generality, it will be assumed that (μ_1, σ_1) are estimated first using (t_1, y_1) and (t_2, y_2) according to the Progressive Read Algorithm described in Section 3. In this case, Eq. (1) reduces to $Q\left(\frac{\mu_1 - t_i}{\sigma_1}\right) = 2y_i - 2n_{y_i}$ for $i = 1, 2$.

If the read locations are on the tails of the distributions, a small perturbation in the cdf value y could cause a significant change in $Q^{-1}(y)$. This will in turn lead to a significant change in the estimates. Specifically, a first-order Taylor expansion of $Q^{-1}(y + n_y)$ at y can be written as

$$Q^{-1}(y + n_y) = x - \sqrt{2\pi} e^{\frac{x^2}{2}} n_y + O(n_y^2), \quad (10)$$

where $x = Q^{-1}(y)$. Since the exponent of e is always positive, the first-order error term is minimized when $x = 0$. The expressions for (μ_1, σ_1) and (μ_2, σ_2) as seen in Eqs (3)-(6) use inverse Q functions, so the estimation error due to read noise will be reduced when the read is done at the mean of the Gaussian distribution. Using the expansion in Eq. (10), Eq. (3) can be written as

$$\hat{\sigma_1} = \frac{t_2 - t_1}{\frac{t_2-t_1}{\sigma_1} + 2\sqrt{2\pi}(e^{x_2^2/2}n_{y_2} - e^{x_1^2/2}n_{y_1}) + O(n_{y_1}^2, n_{y_2}^2)}, \tag{11}$$

where $x_i = \frac{\mu_1 - t_i}{\sigma_1}$, $i = 1, 2$. The first-order Taylor expansion of Eq. (11) at σ_1 is given by

$$\begin{aligned} \hat{\sigma_1} &= \sigma_1 - \frac{\sigma_1^2}{t_2 - t_1}(n_2 - n_1) + O(n_1^2, n_2^2) \\ &= \sigma_1 + e_{\sigma_1}, \end{aligned} \tag{12}$$

where

$$n_1 = 2\sqrt{2\pi}e^{\frac{(t_1-\mu_1)^2}{2\sigma_1^2}} n_{y_1} + O(n_{y_1}^2) \tag{13}$$

$$n_2 = 2\sqrt{2\pi}e^{\frac{(t_2-\mu_1)^2}{2\sigma_1^2}} n_{y_2} + O(n_{y_2}^2). \tag{14}$$

A similar expansion can be performed for $\hat{\mu_1}$, obtaining

$$\begin{aligned} \hat{\mu_1} &= \mu_1 - \sigma_1\frac{(t_2 - \mu_1)n_1 - (t_1 - \mu_1)n_2}{t_2 - t_1} + O(n_1^2, n_2^2) \\ &= \mu_1 + e_{\mu_1}, \end{aligned} \tag{15}$$

where e_{σ_1} and e_{μ_1} are the estimation errors.

Two different tendencies can be observed in the above expressions. On one hand, Eq. (13) and (14) suggest that both t_1 and t_2 should be chosen close to μ_1 so as to reduce the magnitude of n_1 and n_2. On the other hand, if t_1 and t_2 are very close together, the denominators in Eq. (12) and (15) can become small, increasing the estimation error.

The error expansions for $\hat{\mu_2}$, $\hat{\sigma_2}$ and $\hat{t^\star}$, are omitted for simplicity, but it can be shown that the dominant terms are linear in n_{y_i}, $i = 1, \ldots, 4$ as long as all n_{y_i} are small enough.

The Taylor expansion for BER($\hat{t^\star}$) at $t^\star$ is

$$\begin{aligned} \text{BER}(\hat{t^\star}) &= \text{BER}(t^\star) + \left(\frac{1}{2\sigma_2}G\left(\frac{\mu_2 - t^\star}{\sigma_2}\right) - \right. \\ &\quad \left. \frac{1}{2\sigma_1}G\left(\frac{\mu_1 - t^\star}{\sigma_1}\right)\right) e_{t^\star} + O(e_{t^\star}^2) \\ &= \text{BER}(t^\star) + G\left(\frac{\mu_2 - t^\star}{\sigma_2}\right) \cdot \\ &\quad \left(\frac{\mu_2 - t^\star}{2\sigma_2^3} - \frac{\mu_1 - t^\star}{2\sigma_1^2\sigma_2}\right) e_{t^\star}^2 + O(e_{t^\star}^3), \end{aligned} \tag{16}$$

where $\hat{t^\star} = t^\star + e_{t^\star}$. The cancellation of the first-order term is justified by Eq. (8). Summarizing, the mean and variance estimation error increases linearly with the read noise, as does the deviation in the estimated optimal read threshold. The BER, on the other hand, increases quadratically. As a consequence, even if the estimation error in $\hat{t^\star}$ is large, the resulting BER($\hat{t^\star}$) is still close to the minimum possible BER for the proposed Progressive Read Algorithm. The numerical simulations in Fig. 2 confirm these results.

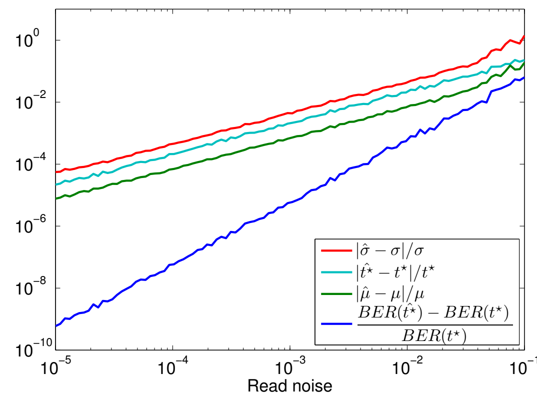

Fig. 2. The error in the mean, variance and threshold estimates increases linearly with the read noise, but the resulting increase in BER grows quadratically.

5. LDPC: TRADEOFF BER-LLR

This section considers a new scenario where a layered decoding approach is used for increased error-correction capability. Specifically, for an LDPC code, after reading a page, the controller may first attempt to correct any bit errors in the read-back codeword using a hard decoder alone, typically a bit-flipping hard-LDPC (H-LDPC) decoder [**?**]. The success of this decoding depends mostly on the number of such errors. Reading with the threshold $\hat{t^\star}$ found through Eq. (9) increases the probability of hard-decoding success. However, there are cases in which even BER($\hat{t^\star}$) is too high for the hard decoder. When this happens, the controller will attempt a soft decoding, typically using a min-sum or sum-product soft-LDPC (S-LDPC) decoder.

Soft decoders are more powerful, but also significantly slower and less power efficient than hard decoders. Consequently, the frequency with which S-LDPC decoding is used plays a significant role in the controller's average read time. In order to estimate this frequency, one must look at the distribution of the number of errors, and not at BER alone. Subsection 5.1 addresses this topic.

The error-correction capability of a soft decoder depends heavily on the quality of the input LLR values. It is always possible to increase such quality by performing additional reads, but this decreases read throughput. Subsection 5.2 shows how the Progressive Read Algorithm from the previous section can be modified to find high quality LLR values.

5.1. Distribution of the number of errors

Let N be the number of bits in a codeword. Assuming that all levels are equally likely, the probability of error for any given bit, denoted p, is given in Eq. (7). Errors can be considered to be independent, hence the number of errors in a codeword follows a binomial distribution with parameters N and p. If N is large, it becomes convenient to approximate the bino-

Failure rate	$p = 0.008$	$p = 0.01$	$p = 0.012$
$k = 23$	0.05	0.28	0.62
$k = 25$	0.016	0.15	0.46
$k = 27$	0.004	0.07	0.31

Table 1. Failure rate for a $N = 2048$ BCH code.

mial by a Gaussian distribution with mean Np and variance $Np(1-p)$, or by a Poisson distribution with parameter Np when Np is small.

Under the Gaussian approximation paradigm, a codeword fails to decode with probability $Q\left(\frac{k-Np}{\sqrt{Np(1-p)}}\right)$, where k denotes the number of bit errors that can be corrected. As seen in Table 1, a small change in the value of k may result in a significant increase in the amount of time a stronger code is needed. This has a direct impact on average power consumption of the controller. The distribution of bit errors can thus be used to judiciously obtain a value of k in order to meet a power constraint.

5.2. Obtaining soft inputs

Soft inputs to an S-LDPC decoder are given in the form of log-likelihood ratios (LLR). The LLR value associated to a bit is defined as $LLR(b) = \log(p_0/p_1)$, where p_0 and p_1 are the probability that the bit is 0 and 1, respectively. If a total of M reads have been performed on a page, each cell can be classified as falling into one of the $M+1$ intervals between the read thresholds. The cell's LLR value can then be found using the mean and variance estimates as

$$LLR = \log\left(\frac{Q((t_l - \hat{\mu_2})/\hat{\sigma_2}) - Q((t_u - \hat{\mu_2})/\hat{\sigma_2})}{Q((t_l - \hat{\mu_1})/\hat{\sigma_1}) - Q((t_u - \hat{\mu_1})/\hat{\sigma_1})}\right), \tag{17}$$

where t_l is the largest read threshold known to be below the cell's voltage value and t_u is the smallest read threshold known to be above it. If the cell's voltage is smaller than all the read thresholds then $t_l = -\infty$. Similarly, if the cell's voltage is larger than all the read thresholds then $t_u = \infty$.

When the mean and variance are known, it is possible to obtain good LLR values by reading at the locations that maximize the mutual information between the input and output of the multiple read channel [**?**]. Those locations tend to be on the so-called uncertainty region, where both pdfs are comparable. Unfortunately, the mean and variance are generally not known and need to be estimated. Section 3 provided some guidelines on how to choose read thresholds in order to obtain accurate estimates, but those reads tend to produce poor LLR values. Hence, there are two opposing trends: spreading out the reads over a wide range of voltage values yields more accurate mean and variance estimates but degrades the performance of the LDPC decoder, while concentrating the reads on the uncertainty region provides better LLR values but might yield inaccurate estimates which in turn undermine the LDPC decoding. The simulations in the next section show that a good trade-off can be obtained by taking two reads close to one of the levels' mean voltages and another two in the uncertainty region.

6. NUMERICAL RESULTS

This section presents simulation results evaluating the read schemes proposed in Sections 3 and 5. Two scenarios will be considered, corresponding to a fresh page with $\text{BER}(t^\star) = 0.002$ and a worn-out page with $\text{BER}(t^\star) = 0.02$. The mean voltage values for each level will be the same in both scenarios, but the standard deviations will differ. Specifically, $\mu_1 = 1$ and $\mu_2 = 2$ for both pages, but the fresh page will be modeled using $\sigma_1 = 0.13$ and $\sigma_2 = 0.22$, while the old page will be modeled using $\sigma_1 = 0.18$ and $\sigma_2 = 0.32$.

For each scenario, three different strategies for selecting the read thresholds were evaluated. The first strategy, S_1, tries to minimize the error in the estimates by reading close to the level means. The second strategy, S_2, performs two reads close to one of the level means and the other two on the uncertainty region. Finally, S_3 concentrates all the reads on the uncertainty region, attempting to attain highly informative LLR values. Specifically, as depicted in Fig. 3,

- S_1 : (t_1, t_2, t_3, t_4) such that $y_1 = 0.15$, $y_2 = 0.35$, $y_3 = 0.65$ and $y_4 = 0.85$
- S_2 : (t_1, t_2, t_3, t_4) such that $y_1 = 0.2$, $y_2 = 0.41$, $y_3 = 0.52$ and $y_4 = 0.63$
- S_3 : (t_1, t_2, t_3, t_4) such that $y_1 = 0.39$, $y_2 = 0.48$, $y_3 = 0.51$ and $y_4 = 0.6$

where Eq. (1) relates t_i to y_i, and n_{y_i}, $i = 1, \ldots, 4$, was assumed to be uniformly distributed between -0.02 and 0.02.

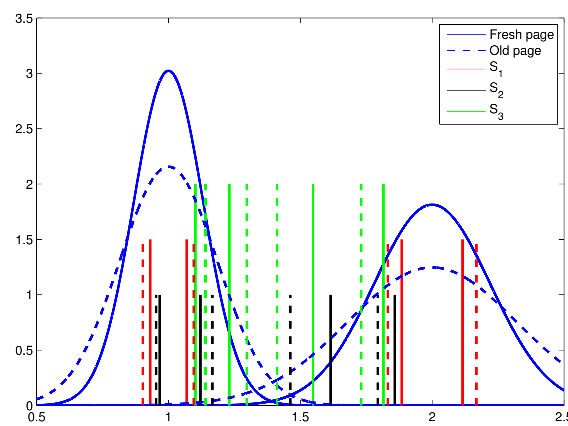

Fig. 3. Read thresholds for strategies S_1, S_2 and S_3 for a fresh and a worn-out page.

The first three rows in Table 2 show the relative estimation error of the mean, variance and optimal threshold. It can be observed that S_1 provides the lowest estimation error, while

FRESH PAGE	S_1	S_2	S_3
$\lvert\hat{\mu}-\mu\rvert/\mu$	0.008	0.018	0.13
$\lvert\hat{\sigma}-\sigma\rvert/\sigma$	0.075	0.15	0.81
$\lvert\hat{t^\star}-t^\star\rvert/t^\star$	0.025	0.038	0.07
$\lvert\mathrm{BER}(\hat{t^\star})-\mathrm{BER}(t^\star)\rvert/\mathrm{BER}(t^\star)$	0.07	0.16	0.56
LDPC fail rate	1	0	0.05
Genie LDPC fail rate	1	0	0
OLD PAGE	S_1	S_2	S_3
$\lvert\hat{\mu}-\mu\rvert/\mu$	0.01	0.025	0.14
$\lvert\hat{\sigma}-\sigma\rvert/\sigma$	0.075	0.15	0.6
$\lvert\hat{t^\star}-t^\star\rvert/t^\star$	0.022	0.023	0.04
$\lvert\mathrm{BER}(\hat{t^\star})-\mathrm{BER}(t^\star)\rvert/\mathrm{BER}(t^\star)$	0.015	0.017	0.05
LDPC fail rate	1	0.059	0.26
Genie LDPC fail rate	1	0.006	0

Table 2. Trade-off between BER and LDPC failure rate.

S_3 produces clearly wrong estimates. The estimates provided by S_2 are noisier than those provided by S_1, but are still acceptable. The relative increase in BER when reading at $\hat{t^\star}$ instead of at $t^\star$ is shown in the fourth row of each table. It is worth noting that the BER($\hat{t^\star}$) does not increase significantly, even with inaccurate mean and variance estimates. This validates the derivation in Section 4.

Finally, the last two rows on each table show the failure rate after 20 iterations of a min-sum LDPC decoder for two different methods of obtaining soft information. The LDPC code had 18% redundancy and codeword length equal to 35072 bits. The fifth row corresponds to LLR values obtained using the mean and variance estimates from the Progressive Read Algorithm and the last row, labeled "Genie LDPC", corresponds to using the actual values instead of the estimated ones. It can be observed that strategy S_1, which provided very accurate estimates, always fails in the LDPC decoding. This is due to the wide range of cell voltages that fall between the middle two reads, being assigned an LLR value close to 0. The fact that the "Genie LDPC" performs better with S_3 than with S_2 shows that the read locations chosen by the former are better. However, S_2 provides lower failure rate in the more realistic case where the means and variances need to be estimated.

In summary, S_2 was found to be best from an LDPC code point of view and S_1 from a pure BER-minimizing perspective. S_3 as proposed in [**?**] is worse in either case unless the voltage distributions are known.

7. CONCLUSION

NAND flash controllers typically do several re-reads using different read thresholds to recover host data in the presence of noise. A read operation takes the same time regardless of the threshold being used. Therefore, reducing read time can only be done by reducing the number of reads. Another component of read time is the decoding time required to decode the ECC protecting the written data. This paper proposed algorithms to reduce total read time by using a limited number of re-reads with good read thresholds to minimize bit-error-rate. Minimizing raw bit-error-rate does not always lead to minimal codeword-error-rate, specifically for LDPC codes. Therefore, the paper proposed an algorithm to minimize codeword-error-rate or sector-failure-rate for LDPC codes. Simulation results were used to validate the performance of the proposed algorithms.

Acknowledgment

This research was supported in part by NSF Grant CCF-1116739 and by the Center for Magnetic Recording Research at UCSD.