Histogram-Based Flash Channel Estimation

Haobo Wang, Tsung-Yi Chen, and Richard D. Wesel
whb12@ucla.edu, tsungyi.chen@northwestern.edu, wesel@ee.ucla.edu

***Abstract*—Current generation Flash devices experience significant read-channel degradation from damage to the oxide layer during program and erase operations. Information about the read-channel degradation drives advanced signal processing methods in Flash to mitigate its effect. In this context, channel estimation must be ongoing since channel degradation evolves over time and as a function of the number of program/erase (P/E) cycles. This paper proposes a framework for ongoing model-based channel estimation using limited channel measurements (reads). This paper uses a channel model characterizing degradation resulting from retention time and the amount of charge programmed and erased. For channel histogram measurements, bin selection to achieve approximately equal-probability bins yields a good approximation to the original distribution using only ten bins (i.e. nine reads). With the channel model and binning strategy in place, this paper explores candidate numerical least squares algorithms and ultimately demonstrates the effectiveness of the Levenberg-Marquardt algorithm which provides both speed and accuracy.**

***Index Terms*—Flash, Channel Estimation, Least Square, Binning Strategy**

I. Introduction

With widespread use in computers, phones and even satellites, Flash memory has become one of the key components directly contributing to the fast-paced evolution of electronic systems. However, the reliability of Flash memory degrades with respect to usage and retention time. Both the extent of usage and the retention time during which data can be reliably recovered are limited. This causes significant drawbacks in practical applications. Furthermore, physical cell density and modulation constellation density are increasing rapidly to satisfy the demand for increased storage capacity under strict physical size constraints. This amplifies the degradation problem.

Modern Flash storage solutions employ channel codes [1], [2] to increase reliability, but this approach alone cannot effectively counteract the channel capacity decrease caused by degradation due to recursively program and erase the cells. Signal processing methods such as Dynamic Voltage Allocation (DVA) [3] and Dynamic Threshold Assignment (DTA) [4] actively mitigate channel degradation. Effective decoding of channel codes, possible selection of channel code rate, and adaptive signal processing such as DVA and DTA require channel state information. Thus, it is necessary to have a robust on-line estimation framework for this evolving channel.

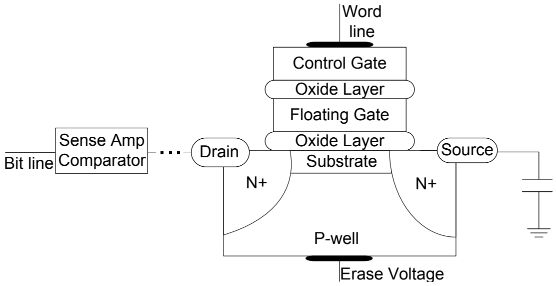


Fig. 1. Common structure of a P-well Flash memory cell.

Figure 1 shows basic physical structure for a P-well Flash memory cell. The threshold voltage of a Flash cell depends on the amount of charge in the floating gate and the device's intrinsic threshold voltage when the floating gate is empty. Electrical charge can be programmed to and erased from the floating gate through the oxide layer. As a result, the threshold voltage can be controlled by program and erase (P/E) operations [5]. Information about the threshold voltage can be obtained by applying a word line voltage and reading the sense amplifier output to determine if the threshold voltage has been surpassed. Due to channel degradation, the measured value of the threshold voltage often differs from the originally stored threshold voltage.

This paper explores histogram-based channel estimation (HBCE) for Flash. For a given block, multiple sense amplifier reads at distinct wordline voltages yield a histogram of the operational threshold voltages. This histogram can provide an estimate for the actual read channel distribution.

In [6], the authors demonstrate that a multimodal Gaussian distribution representing four-level MLC Flash can be accurately estimated by least squares (LS) with 12-bin histograms using the Levenberg-Marquardt (LM) algorithm. In [6] eight parameters are estimated, four means and four variances.

Our work is largely inspired by [6]. We apply HBCE to a read channel distribution that models the device physics of Flash degradation, namely wear-out and retention-loss. By modeling the underlying physical properties, our channel model uses fewer parameters (five) to characterize the channel. The combination of fewer parameters and the use of an improved binning paradigm allows LM to accurately estimate a more detailed channel distribution using only 10-bins.

The remainder of this paper is organized as follows: Section II presents the Flash channel model used in the paper. Section III compares three binning strategies from the perspectives of squared Euclidean distance and effective resolution. Section IV compares three least squares channel parameter estimation algorithms in terms of their speed and accuracy in estimat-

This work was supported by a grant from Western Digital Corporation and by the Center on Development of Emerging Storage Systems (CoDESS) http://www.uclacodess.org.

ing channel parameters that minimize the squared Euclidean distance between the measured histogram and the histogram produced by the estimated channel parameters in the context of our channel model. Section V presents simulation results demonstrating that ten bins (nine reads) using an equal-probability binning strategy and the Levenberg-Marquardt least-squares algorithm provides excellent channel parameter estimation. Section VI concludes the paper..

II. Flash Memory Channel Model

Recent research in Flash provides many good channel models, we use the model presented in [3] as the basis of our analysis.

A. Degradation Mechanism

Based on the literature [7]–[9], two important forms of degradation are investigated in our analysis. The first mechanism is called wear-out, which is the P/E cycling-related degradation. The second mechanism, which is called retention loss, is also caused by the P/E cycling. The key distinction between wear-out and retention loss is that variations in threshold due to wear-out occur immediately after writing and variations in threshold due to retention loss occur over the retention time becoming more severe with longer retention times.

Because Flash cells are densely packed in a two dimensional array in the chip, the coupling effect among the cells and the share of electrical connections cause distortion of the channel known as cell-to-cell interference. This interference depends on the specific structure of individual Flash chip and the operation sequence of the controller [10], [11]. As a result, generalization of our channel model to include such disturbance is highly implementation dependent. Thus, in this paper, cell-to-cell interference is not modeled. Many methods counteracting this problem have been proposed in the semiconductor community, such as P/E sequence optimization and write voltage pre-distortion [11], [12].

Similarly, read disturb is implementation dependent and not included in our model. However, given the implementation details, we believe that both read disturb and cell-to-cell interference could be incorporated in a general channel model.

B. Channel Model

Based on the theoretical analysis and experimental results from the literature [7]–[9], [12]–[14], our Flash memory channel model is formulated with three additive noise components:

$$y = x + n_p + n_w + n_r \,, \tag{1}$$

where x is the intended threshold voltage written to a cell, and y is the measured threshold voltage. The noise n_p is the programing noise component related only to the P/E process, n_w denotes the wear-out noise caused by wear-out effect, and n_r represents the retention noise caused by retention loss.

Figure 2 shows an example channel distribution, which demonstrates the impact of the noise components to the channel.

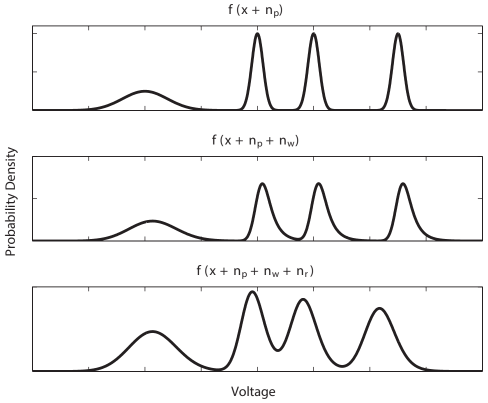


Fig. 2. Flash read channel probability density functions illustrating degradation mechanisms.

1) Programming Noise: The uncertainty of programmed and erased state threshold voltages immediately after P/E operations of a new cell in current generation Flash memory can be modeled by Gaussian distribution. The variance of the distribution depends on cell's stored state, [12]–[14]. Let l denote the level of intended threshold with $l = 0$ representing the erased state and $l > 0$ representing programmed states. Programming Noise can then be modeled as

$$f_{N_P}(n_p|l) = \begin{cases} \mathcal{N}(0, \sigma_p^2) & \text{if } l = 0, \\ \mathcal{N}(0, \sigma_e^2) & \text{if } l > 0. \end{cases} \quad \text{where } \sigma_e > \sigma_p \,. \tag{2}$$

The noise variance of the programmed states is significantly smaller that the variance for the erased state because of a tight programming feedback loop [14]. This is modeled by having $\sigma_e > \sigma_p$.

2) Wear-out Noise: Wear-out noise is caused by recurring P/E operations damaging the oxide layer through generation of oxide traps and interface traps. In [7], the authors point out that interface traps have the most significant impact on wear-out in deeply scaled devices; therefore, the impact of oxide traps is not considered in this component.

The wear-out effect of traps on measured thresholds can be modeled as Random Telegraph Noise (RTN). RTN widens the distribution of read thresholds with exponential tails on both sides of the actual threshold voltage [8]. However, based on the data from real devices, the distribution of read thresholds features significant single sided exponential tails in the positive direction, thus we use the following exponential distribution as the model for wear-out noise:

$$f_{N_W}(n_w) = \begin{cases} \frac{1}{\lambda} e^{-\frac{n_w}{\lambda}} & \text{if } n_w \geq 0 \\ 0 & \text{if } n_w < 0 \end{cases} , \tag{3}$$

where λ is the channel parameter which functions as a metric for interface trap density.

3) Retention Noise: Retention loss is caused by electron detrapping. Thus, the characteristic of this noise component is determined by the interface trap density, oxide trap density and retention time [9]. Following [9], we model retention loss as Gaussian noise with distribution

$$f_{N_R}(n_r) = \frac{1}{\sigma_r\sqrt{2\pi}} e^{-\frac{(n_r-\mu_r)^2}{2\sigma_r^2}}\,, \tag{4}$$

where

$$\mu_r = \gamma_{\mu_r}(x - x_0)\,, \tag{5}$$

$$\sigma_r = \gamma_{\sigma_r}\sqrt{x - x_0}\,. \tag{6}$$

The parameters γ_{μ_r} and γ_{σ_r} represent trap density and retention time. Variable x is the intended threshold, and x_0 is the intended erased-state threshold. Note that in this model, there is no retention loss if the intended threshold is that of the erased state.

4) Overall Conditional Distribution for Flash Channel: From the discussion above, the conditional distribution for Flash read channel can be summarized as

$$f_{Y|X}(y|x) = \frac{e^{\frac{\mu_r+x-y}{\lambda}+\frac{\sigma^2}{2\lambda^2}}}{\lambda} \cdot Q\left(\frac{\mu_r+x-y}{\sigma} + \frac{\sigma}{\lambda}\right)\,, \tag{7}$$

where x is the intended threshold voltage and y is the measured threshold voltage. The value of σ depends on the intended voltage as follows:

$$\sigma = \begin{cases} \sqrt{\sigma_e^2 + \sigma_r^2} & \text{if } x = x_0 \\ \sqrt{\sigma_p^2 + \sigma_r^2} & \text{otherwise} \end{cases}\,. \tag{8}$$

$Q(m)$ is the tail probability of the standard normal distribution: $Q(m) = \int_m^\infty \frac{e^{-x^2/2}}{\sqrt{2\pi}}\,dx$. The parameters σ_r, μ_r and λ evolve over both time and P/E cycling process, as the channel degrades, while σ_e and σ_p remain constant. Given a specific retention time, P/E cycling condition, and the physics related parameters, the exact value of the current channel parameters can be determined using channel parameter degradation model proposed in [3].

III. Binning Strategy for Histogram Measurement

A good binning strategy, i.e. selecting the proper number and placement of word line voltages for the reads that will create the histogram bins, is critical for the efficiency and accuracy of practical histogram-dependent signal processing methods in Flash.

A. Number of Bins

A fairly accurate channel estimation can, of course, be derived from a complete voltage scan which uses the smallest possible bin sizes by reading at every available voltage level using the so-called debug mode. However, the large number of reads required by this process significantly increases device read latency and stalls normal operations. Such a large number of reads is likely not necessary. From the soft decoding literature [6], [15], [16], a relatively small number of word-line voltages sufficiently gives good performance in terms of both decoding and channel estimation.

Furthermore, too many bins in the histogram increase the computational complexity in each iteration of the least square algorithms described in Section IV below, and also require more storage space in the controller. Thus a relatively small number of bins can reduce both complexity and latency. The choice for the number of bins required also depends on the channel estimation algorithm employed. Basic algorithms usually require more detailed channel measurements than advanced algorithms. As shown below in Sec. V, a 10-bin histogram can provide accurate estimation of the channel parameters in our model. Thus, in exploring the performance of the three bin-placement paradigms, we focus on the 10-bin case.

B. Selecting a Bin-Placement Paradigm

We will consider three bin-placement paradigms: equal-width, equal-probability, and maximum mutual information (MMI). Equal-width histograms have bins covering intervals of equal length except for the semi-infinite bins at the edges. Equal-probability histograms allocate bins having the same probability (same number of counts), to the extent that this can be achieved without a-posteriori knowledge. MMI histograms proposed in [15], [16] optimize decoding performance by maximizing the mutual information between the intended threshold voltages and the bin location identified by the reads.

As presented in Section IV, channel parameters are estimated by minimizing the squared Euclidean distance between the measured histogram acting as the reference and the histogram constructed with the estimated channel parameters. To achieve good estimation accuracy, the measured histogram should be as close to the original channel distribution as possible. To compare bin-placement paradigms, the squared Euclidean distance between the channel distribution $f(y)$ and the histogram induced by $f(y)$ is used as the metric to evaluate the amount of discretization error of each bin-placement paradigm. This metric D_{E^2} is defined as follows:

$$D_{E^2} = \sum_{i=0}^{M-1} \int_{q_i}^{q_{i+1}} \left(f(y) - \frac{H_i}{q_{i+1}-q_i}\right)^2 dy\,, \tag{9}$$

where $f(y)$ is the true read channel distribution, M is the number of bins, and q_i, q_{i+1} represent the left and right boundary of the the ith interval. H_i is the probability of ith bin induced by $f(y)$, $H_i = \int_{q_i}^{q_{i+1}} f(y)dy$, and $\frac{H_i}{q_{i+1}-q_i}$ denotes the probability density of the ith bin.

Fig. 3 compares the metric D_{E^2} generated from nine reads (ten bins) and the original channel distribution for the three bin-placement paradigms. The parameters provided in [3] are used to generate evolving channel distributions as a function of the number of P/E cycles. The equal-probability bin placement strategy provides a lower D_{E^2} metric, and hence a better approximation to the original channel, than the other two strategies over a large span of P/E cycling conditions. The performance difference is especially significant when the device condition is new. This behavior for all three paradigms is also seen when using histograms with 7 bins. As the number of bins grows, the performance difference becomes smaller, but we seek good performance with the fewest possible bins.

In addition to the D_{E^2} metric, effective resolution is used as a metric for the effectiveness of a bin placement strategy. Because the value of each bin is always greater than or equal to zero, two adjacent zero-height bins can be combined as one bin. Thus, a word-line voltage at the boundary of two zero-count bins is a wasted read. Although histogram bin probabilities derived from the integration will be nonzero in every interval, real measurements of a finite number of cells will produce zero-count bins.

To compute the effective resolution, combine adjacent zero-count bins into one effective bin and count the number of resulting bins. Fig. 4 shows the effective resolution as a function of the number of P/E cycles for the three bin-placement paradigms. Adjacent bins with induced probability less than 10^{-4} are combined. The equal-probability bin-placement paradigm has full resolution throughout the entire P/E cycling process, while the other paradigms lose resolution in some P/E cycling conditions. This suggests that the equal-probability bin-placement paradigm has a good tracking capability over the whole Flash lifetime.

Because it has superior performance both in terms of D_{E^2} and effective resolution, the equal-probability bin-placement paradigm is used in the channel parameter estimation discussed in the remainder of this paper.

IV. Channel Parameter Estimation

A. Cost Function

From the discussion in Section II, the channel parameter vector should be $[\lambda, \sigma_p, \sigma_e, \sigma_r, \mu_r]$. However, both σ_r and μ_r are intended-threshold-level dependent. As a result, $\boldsymbol{\alpha} = [\lambda, \sigma_p, \sigma_e, \gamma_{\sigma_r}, \gamma_{\mu_r}]$ is used in the following discussion as the level independent channel parameter vector to be estimated. Level-independent channel parameters are preferred for supporting the DVA algorithm, which is a target application for HBCE.

Define $[q_0, q_1, \ldots, q_M]$ as the boundaries of bins where $q_0 = -\infty$, $q_M = \infty$, and M is the number of bins. The number of cells in each bin can be estimated as

$$\hat{N}_{bin,i} = \sum_{k=1}^{L} N_k P(q_i < y < q_{i+1} | x_k) \,, \qquad (10)$$

where $P(q_i < y < q_{i+1}|x_k) = \int_{q_i}^{q_{i+1}} f_{Y|X}(y|x_k)\, dy$ denotes the probability of a measured threshold falling in the ith bin when the intended threshold is x_k. L is the number of intended threshold levels, and N_k is the number of cells in each level determined by the stored data.

The cost function is defined as the normalized square Euclidean distance between the estimated histogram and the reference histogram

$$C_M = \sum_{i=0}^{M-1} \left(\frac{N_{bin,i} - \hat{N}_{bin,i}}{N} \right)^2 \,, \qquad (11)$$

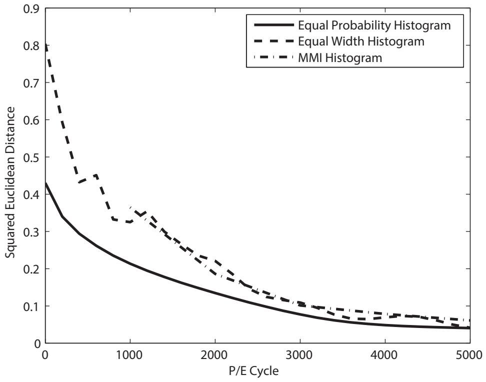


Fig. 3. Squared Euclidean distance between the channel distributions and corresponding histograms (10 bins).

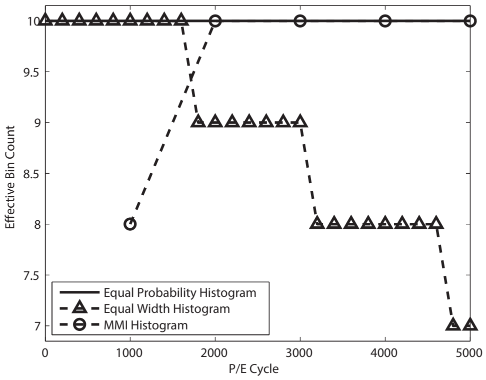


Fig. 4. Effective resolution of different histograms (10 bins).

where N is the total number of cells measured, and $N_{bin,i}$ is the ith bin's cell count in the reference histogram. The gradient of the cost function is defined as

$$\nabla \boldsymbol{C}_M(\boldsymbol{\alpha}) = 2 \cdot (\boldsymbol{J}_{\boldsymbol{G}_M}(\boldsymbol{\alpha}))^T \cdot \boldsymbol{G}_M(\boldsymbol{\alpha}) \,, \qquad (12)$$

where $\boldsymbol{J}_{\boldsymbol{G}_M}(\boldsymbol{\alpha})$ is the Jacobian matrix of the difference vector between the estimated histogram and the reference histogram.

B. Least Squares Algorithms

Least squares algorithms have been widely used to fit a parameterized model to a data set. Three algorithms are examined in the following discussion.

1) Gradient Descent (GD): GD minimizes the cost function by iteratively refining initial estimation of the parameters iteratively based on a linear model. In each iteration, the estimation is renewed by a step vector following the gradient of the cost function.

Algorithm 1 Gradient Descent Algorithm

```
Initialize step size $\beta$ and $\boldsymbol{\alpha} = \boldsymbol{\alpha}^{(0)}$
while $\|\boldsymbol{\alpha}^{(k+1)} - \boldsymbol{\alpha}^{(k)}\| > \eta$ and $k < MaxIteration$ do
    Compute $\boldsymbol{J}_{\boldsymbol{G}_M}(\boldsymbol{\alpha}^{(k)})$ and $\boldsymbol{G}_M(\boldsymbol{\alpha}^{(k)})$
    Compute $\nabla C_M(\boldsymbol{\alpha}^{(k)}) = 2 \cdot (\boldsymbol{J}_{\boldsymbol{G}_M}(\boldsymbol{\alpha}^{(k)})^T \cdot \boldsymbol{G}_M(\boldsymbol{\alpha}^{(k)})$
    $\boldsymbol{\alpha}^{(k+1)} = \boldsymbol{\alpha}^{(k)} - \beta \cdot \nabla \boldsymbol{C}_M(\boldsymbol{\alpha}^{(k)})$
    $k = k + 1$
end while
```

2) Gauss-Newton (GN): A quadratic model is employed to provide more accurate approximation of the cost function. The iterative relation can be represented as

$$\boldsymbol{\alpha}^{(k+1)} = \boldsymbol{\alpha}^{(k)} - (\boldsymbol{J}^T\boldsymbol{J})^{-1}\boldsymbol{J}^T\boldsymbol{G} = \boldsymbol{J}^{+}\boldsymbol{G}\,, \tag{13}$$

where $\boldsymbol{J}^{+}$ is the pseudo-inverse of $\boldsymbol{J}$. Gauss-Newton Algorithm can then be formulated as follows:

Algorithm 2 Gauss-Newton Algorithm

```
Initialize $\boldsymbol{\alpha} = \boldsymbol{\alpha}^{(0)}$
while $\|\boldsymbol{\alpha}^{(k+1)} - \boldsymbol{\alpha}^{(k)}\| > \eta$ and $k < MaxIteration$ do
    Compute $\boldsymbol{J}_{\boldsymbol{G}_M}(\boldsymbol{\alpha}^{(k)})$ and $\boldsymbol{G}_M(\boldsymbol{\alpha}^{(k)})$
    $\boldsymbol{\alpha}^{(k+1)} = \boldsymbol{\alpha}^{(k)} - (\boldsymbol{J}_{\boldsymbol{G}_M}(\boldsymbol{\alpha}^{(k)}))^{+} \cdot \boldsymbol{G}_M(\boldsymbol{\alpha}^{(k)})$
    $k = k + 1$
end while
```

3) Levenberg-Marquardt (LM) [17]: By combining GD and GN, LM possesses the advantages of both algorithms. The update vector $\boldsymbol{\delta_\alpha}$ is calculated by solving $(\boldsymbol{J}^T\boldsymbol{J} + \beta \cdot diag((\boldsymbol{J}_{\boldsymbol{G}_M})^T\boldsymbol{J}_{\boldsymbol{G}_M}))\boldsymbol{\delta_\alpha} = \boldsymbol{J}^T\boldsymbol{G}$ where β acts as a weight to combine the two algorithms.

Algorithm 3 Levenberg-Marquardt Algorithm

```
Initialize $\beta, v, \boldsymbol{\alpha} = \boldsymbol{\alpha}^{(0)}$ and $UpdateFlag = 1$
while $\|\boldsymbol{\alpha}^{(k+1)} - \boldsymbol{\alpha}^{(k)}\| > \eta$ and $k < MaxIteration$ do
    if $UpdateFlag = 1$ then
        Compute $\boldsymbol{J}_{\boldsymbol{G}_M}(\boldsymbol{\alpha}^{(k)})$ and $\boldsymbol{G}_M(\boldsymbol{\alpha}^{(k)})$
    end if
    Solve $((\boldsymbol{J}_{\boldsymbol{G}_M})^T\boldsymbol{J}_{\boldsymbol{G}_M} + \beta \cdot diag((\boldsymbol{J}_{\boldsymbol{G}_M})^T\boldsymbol{J}_{\boldsymbol{G}_M}))\boldsymbol{\delta_\alpha} = (\boldsymbol{J}_{\boldsymbol{G}_M})^T\boldsymbol{G}_M$
    Compute $\boldsymbol{J}_{\boldsymbol{G}_M}(\boldsymbol{\alpha}^{(k)})$ and $G_M(\boldsymbol{\alpha}^{(k)})$
    $\boldsymbol{\alpha}_{temporary} = \boldsymbol{\alpha} - \boldsymbol{\delta_\alpha}$
    if $\sum(err(\boldsymbol{\alpha}))^2 > \sum(err(\boldsymbol{\alpha}_{temporary}))^2$ then
        $UpdateFlag = 1$
        $\beta = \beta \cdot v$
        $\boldsymbol{\alpha} = \boldsymbol{\alpha}_{temporary}$
    else
        $UpdateFlag = 0$
        $\beta = \beta / v$
    end if
    $k = k + 1$
end while
```

V. Simulation Results

To verify the effectiveness of the algorithms described in Section IV and determine the number of reads needed for equal-probability bin-placement paradigm, channel distributions generated with parameters from [3] are used in our simulations. The retention time is set to be one year. P/E cycling conditions from 0 to 4000 P/E are sampled every 300 P/E. The initial conditions for the three algorithms are the same [0.007,0.1,0.4,0.04,-0.4]. Figure 5 illustrates one of the estimation results of the 14 P/E cycling conditions.

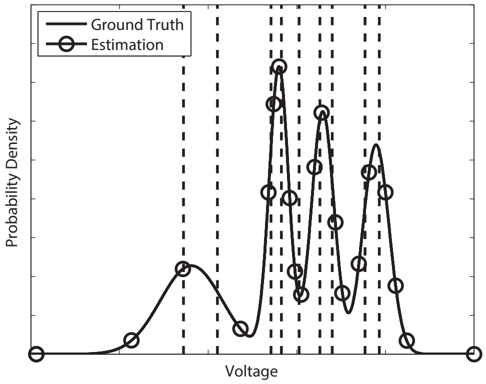


Fig. 5. Channel estimation result at 3000 P/E using Levenberg-Marquardt algorithm and 10-bin equal-probability histogram. (ground truth parameter vector is [0.0099,0.3500,0.0500,0.0617,-0.5882], estimation error vector is 10^{-4}×[0.0101 0.0214 -0.1774 0.0405 -0.0044])

Figure 6 compares the estimating results of γ_{μ_r} with the ground truth, where the estimation algorithms employ 10-bin equal-probability histograms as input. The LM algorithm performs significantly better than GD and GN in terms of both the estimation accuracy and the ability to adapt to different channel conditions.

This behavior is further demonstrated in Table I, which shows the convergence counts of the three algorithms over the 14 sample conditions. Every estimated parameter needs to be within ±1% of the ground truth parameter to qualify the estimated parameter vector as a converged result. GD fails in all simulation cases while reaching the maximum allowed number of iterations in every case. GN provides good results in certain cases with very few iterations. LM provides high estimation accuracy over different channel conditions, with some failures when the channel conditions are very good. Note that estimation accuracy of γ_{σ_r} is usually higher than the other parameters. An intuitive explanation for this is that channel distribution mean shifts can be easily identified by even the histogram itself.

Figure 7 depicts key statistical metrics about the number of iterations when employing LM with histograms that differ in resolution over the 14 cases. The horizontal line segments represent the ranges of iteration counts, the vertical line segments indicate the mean values, and the rectangles show the standard deviations. 10-bin (9 reads) histograms reduce the number of iterations needed with respect to the results using 7-bin (6 reads) histograms. 13-bin (12 reads) histograms do not provide significant reduction in iteration counts, but do increase computational cost in each iteration. Overall, the 10-bin histograms provide a balance between iteration counts and

TABLE I
Converge counts of least square algorithms (over 14 cases).

No. of Reads	GD	GN	LM
6	0	1	12
9	0	3	13
12	0	4	11

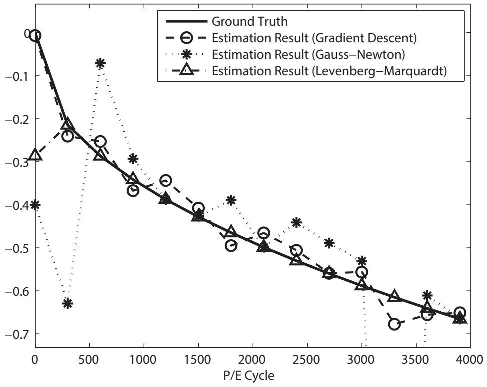
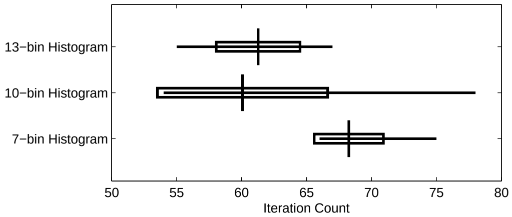


Fig. 6. Estimation result versus ground truth for γ_{μ_r} using 10-bin equal-probability histogram.



Horizontal Line Segment: Range of Iteration Count
Vertical Line Segment: Mean of Iteration Count
Rectangle: Standard Deviation of Iteration Count

Fig. 7. Iteration count statistics using Levenberg-Marquardt algorithm

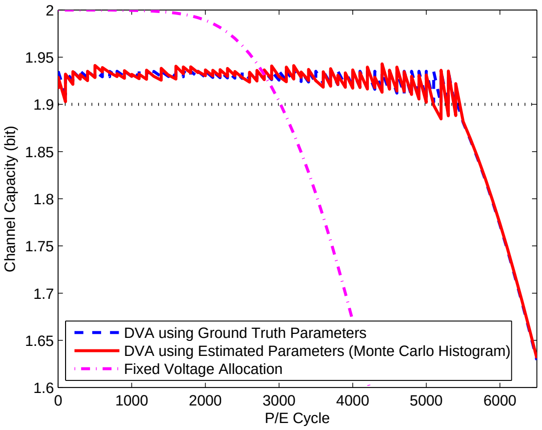


Fig. 8. Dynamic Voltage Allocation simulation result.

the cost of each iteration.

VI. Conclusion

By combining measured histograms with least squares algorithms, we introduce a framework to estimate evolving Flash memory channel. Our analysis and simulation results show that good estimation accuracy and speed can be achieved by using Levenberg-Marquardt algorithm with 10-bin equal-probability histograms. With this framework, Flash channel estimation can be implemented with limited measurements. This enables further utilization of channel characteristics in future Flash solutions. We have successfully used the LM algorithm with a 10-bin histogram for a Dynamic Voltage Allocation [3] algorithm. As shown in Figure 8, the performance using histogram generated by Monte Carlo histogram is not distinguishable from perfect knowledge of the channel.

References

[1] Y. Maeda and H. Kaneko, "Error control coding for multilevel cell flash memories using nonbinary low-density parity-check codes," in *Defect and Fault Tolerance in VLSI Systems, 2009. DFT '09. 24th IEEE International Symposium on*, Oct. 2009, pp. 367–375.

[2] T. Klove, B. Bose, and N. Elarief, "Systematic, single limited magnitude error correcting codes for flash memories," *Information Theory, IEEE Transactions on*, vol. 57, no. 7, pp. 4477–4487, Jul. 2011.

[3] T.-Y. Chen, A. R. Williamson, and R. D. Wesel, "Increasing flash memory lifetime by dynamic voltage allocation for constant mutual information," in *Information Theory and Applications Workshop (ITA), 2014*, Feb. 2014, pp. 1–5.

[4] F. Sala, R. Gabrys, and L. Dolecek, "Dynamic threshold schemes for multi-level non-volatile memories," *Communications, IEEE Transactions on*, vol. 61, no. 7, pp. 2624–2634, Jul. 2013.

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

[6] D.-h. Lee and W. Sung, "Estimation of NAND flash memory threshold voltage distribution for optimum soft-decision error correction," *Signal Processing, IEEE Transactions on*, vol. 61, no. 2, pp. 440–449, Jan. 2013.

[7] J.-D. Lee, J.-H. Choi, D. Park, and K. Kim, "Data retention characteristics of sub-100 nm NAND flash memory cells," *Electron Device Letters, IEEE*, vol. 24, no. 12, pp. 748–750, Dec. 2003.

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

[9] N. Mielke, H. Belgal, A. Fazio, Q. Meng, and N. Righos, "Recovery effects in the distributed cycling of flash memories," in *Reliability Physics Symposium Proceedings, 2006. 44th Annual., IEEE International*, Mar. 2006, pp. 29–35.

[10] R.-A. Cernea, 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," *Solid-State Circuits, IEEE Journal of*, vol. 44, no. 1, pp. 186–194, Jan. 2009.

[11] Y. Cai, O. Mutlu, E. Haratsch, and K. Mai, "Program interference in MLC NAND flash memory: Characterization, modeling, and mitigation," in *Computer Design (ICCD), 2013 IEEE 31st International Conference on*, Oct. 2013, pp. 123–130.

[12] G. Dong, Y. Pan, and T. Zhang, "Using lifetime-aware progressive programming to improve SLC NAND flash memory write endurance," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on*, vol. PP, no. 99, pp. 1–1, 2013.

[13] K. Takeuchi, T. Tanaka, and H. Nakamura, "A double-level-vth select gate array architecture for multilevel NAND flash memories," *Solid-State Circuits, IEEE Journal of*, vol. 31, no. 4, pp. 602–609, Apr. 1996.

[14] C. Compagnoni, A. Spinelli, R. Gusmeroli, A. Lacaita, S. Beltrami, A. Ghetti, and A. Visconti, "First evidence for injection statistics accuracy limitations in NAND flash constant-current fowler-nordheim programming," in *Electron Devices Meeting, 2007. IEDM 2007. IEEE International*, Dec. 2007, pp. 165–168.

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

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

[17] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. pp. 431–441, 1963. [Online]. Available: http://www.jstor.org/stable/2098941