Maximum Likelihood Decoding for Multi-Level Cell Memories with Scaling and Offset Mismatch

Bu, Renfei; Weber, Jos H.

DOI
10.1109/ICC.2019.8761978

Publication date
2019

Document Version
Final published version

Published in
2019 IEEE International Conference on Communications, ICC 2019 - Proceedings

Maximum Likelihood Decoding for Multi-Level Cell Memories with Scaling and Offset Mismatch

Renfei Bu
Applied Mathematics Dept., Optimization Group
Delft University of Technology
Delft, Netherlands
R.Bu@tudelft.nl

Jos H. Weber
Applied Mathematics Dept., Optimization Group
Delft University of Technology
Delft, Netherlands
J.H.Weber@tudelft.nl

***Abstract*—Reliability is a critical issue for modern multi-level cell memories. We consider a multi-level cell channel model such that the retrieved data is not only corrupted by Gaussian noise, but hampered by scaling and offset mismatch as well. We assume that the intervals from which the scaling and offset values are taken are known, but no further assumptions on the distributions on these intervals are made. We derive maximum likelihood (ML) decoding methods for such channels, based on finding a codeword that has closest Euclidean distance to a specified set defined by the received vector and the scaling and offset parameters. We provide geometric interpretations of scaling and offset and also show that certain known criteria appear as special cases of our general setting.**

***Index Terms*—multi-level cell memories, maximum likelihood decoding, Euclidean distance, Pearson distance, scaling and offset mismatch**

I. Introduction

As the on-going data revolution demands storage systems that can store large quantities of data, multi-level cell memories are gaining attention. A multi-level cell is a memory element capable of storing more than a single bit of information, compared to a single-level cell which can store only one bit per memory element [1]. For example, in multi-level cell NAND flash technology, information is stored by introducing more voltage levels that are used to represent more than one bit [2].

It is obvious that, as the number of levels increases, the storage capacity of multi-level cell memories is enhanced. However, due to the increase in the per-cell storage density, the reliability of multi-level cell memories experiences a diverse set of short-term and long-term variations.

Unpredictable stochastic errors are exacerbated with the short-term variation. For example, random errors occur in the programming/reading process, and sometimes it is hard to initialize a cell with the exact voltage. As a result, error correcting techniques are usually considered and applied in multi-level cell memories, such as BCH codes [3], Reed-Solomon codes [4], LDPC codes [5], trellis coded modulation [6], and so on.

In the long term, the performance of multi-level cell memories degrades with age. As documented in [7], the number of electrons of a cell decreases and some cells even become defective over time. The amount of electron leakage depends on various physical parameters, e.g., the device's temperature, the magnitude of the charge, the quality of the gate oxide or dielectric, and the time elapsed between writing and reading the data. It is hard to precisely model these long-term effects on multi-level cell memories. In this paper, we focus on the mean change over time, while variance issues were discussed in [8].

Scaling and offset can weaken the cell's state strength by moving its level closer to the next reference voltage. Various techniques have been proposed to improve the detector's resilience to scaling and offset mismatch. Estimation of the unknown shifts may be achieved by using reference cells, but this is very expensive with respect to redundancy. Also, coding techniques can be applied to strengthen the detector's reliability in case of scaling and offset mismatch; these include rank modulation [9], balanced codes [10], and composition check codes [11]. However, these methods often suffer from large redundancy and high complexity.

Immink and Weber [12] advocate the use of Pearson distance decoding instead of traditional Euclidean distance decoding, in situations which require resistance towards scaling and/or offset mismatch. We use the same channel model as used in [12]: besides the noise, which varies from symbol to symbol, a multiplicative factor a and an additive term b specify the scaling and offset mismatch, respectively, which are assumed to be constant within one block of code symbols, but may be different for the next block. Even though this model neglects certain aspects of multi-level cell memories, such as inter-cell coupling or dependent noise, it still captures key properties of the data corruption process in multi-level cell memories.

The contribution of this work is two-fold. Firstly, in Section III, we derive a maximum likelihood (ML) decoding criterion for multi-level cell channels with Gaussian noise and also suffering from the scaling a and the offset b, which are known to be within certain ranges, specifically $0 < a_1 \leq a \leq a_2$ and $b_1 \leq b \leq b_2$. The ML decoding criterion will also be illustrated with geometric interpretations. Secondly, the proposed ML criterion provides a general framework, including the scaling-only case and the offset-only case. Some known criteria [13] [14] are shown to be special cases of this framework for particular a_1, a_2, b_1, and b_2 settings.

This paper aims to generalize ML decoding for multi-level cell channel with Gaussian noise and scaling and offset mismatch. We start by providing the multi-level cell channel model in Section II, starting with several definitions and ending with the Euclidean distance-based and Pearson distance-based decoding criteria. In Section III, we show how to achieve ML decoding for this channel. We continue in Section IV considering several special cases, which relate to known results in this area. We wrap up the paper with some comments and ideas for future work in Section V.

II. Preliminaries and Channel Model

We start by introducing some notations. For any vector $\mathbf{u} = (u_1, u_2, \ldots, u_n) \in \mathbb{R}^n$, let

$$\bar{\mathbf{u}} = \frac{1}{n}\sum_{i=1}^{n} u_i$$

denote the average symbol value, let

$$\sigma_{\mathbf{u}} = \left(\sum_{i=1}^{n} (u_i - \bar{\mathbf{u}})^2\right)^{1/2}$$

denote the unnormalized symbol standard deviation, and let

$$\|\mathbf{u}\| = \left(\sum_{i=1}^{n} |u_i|^2\right)^{1/2}$$

denote the (Euclidean) norm. We write $\langle \mathbf{u}, \mathbf{v} \rangle$ for the standard inner product (the dot product) of two vectors $\mathbf{u}$ and $\mathbf{v}$, i.e.,

$$\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^{n} u_i v_i = \|\mathbf{u}\| \, \|\mathbf{v}\| \cos\theta,$$

where θ is the angle between $\mathbf{u}$ and $\mathbf{v}$. Note that $\langle \mathbf{u}, \mathbf{u} \rangle = \|\mathbf{u}\|^2$.

Consider transmitting a codeword $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ from a codebook $\mathcal{S}$ over the q-ary alphabet $\mathcal{Q} = \{0, 1, \ldots, q-1\}$, $q \geq 2$, where n is a positive integer. This is based on the fact that each cell is initialized with one of a finite discrete set of voltages. The transmitted symbols x_i are distorted by additive noise v_i, by a factor $a > 0$, called scaling/gain, and by an additive term b, called offset, i.e., the received symbols r_i read

$$r_i = a(x_i + v_i) + b,$$

for $i = 1, \ldots, n$. The parameters $v_i \in \mathbb{R}$ are zero-mean i.i.d. Gaussian noise samples with variance of $\sigma^2 \in \mathbb{R}$, that is, the noise vector $\mathbf{v}$ has distribution

$$\phi(\mathbf{v}) = \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} e^{-v_i^2/(2\sigma^2)}. \tag{1}$$

The scaling and offset (unknown to both the sender and the receiver) may slowly vary in time due to various factors in multi-level cells. So we assume they may differ from codeword to codeword, but do not vary within a codeword. The received vector when a codeword $\mathbf{x}$ is transmitted is

$$\mathbf{r} = a(\mathbf{x} + \mathbf{v}) + b\mathbf{1}, \tag{2}$$

where $\mathbf{1} = (1, 1, \ldots, 1)$ is the real all-one vector of length n.

A. Euclidean Distance-Based Decoding

A well-known decoding criterion upon receipt of the vector $\mathbf{r}$ is to choose a codeword $\hat{\mathbf{x}} \in \mathcal{S}$ which minimizes the (squared) Euclidean distance between the received vector $\mathbf{r}$ and codeword $\hat{\mathbf{x}}$, i.e.,

$$L_e(\mathbf{r}, \hat{\mathbf{x}}) = \|\mathbf{r} - \hat{\mathbf{x}}\|^2 = \sum_{i=1}^{n} (r_i - \hat{x}_i)^2. \tag{3}$$

It is known to be ML with regard to handling Gaussian noise, but not optimal in situations which require resistance towards scaling and/or offset mismatch.

B. Pearson Distance-Based Decoding

The Pearson distance measure [12] naturally lends itself to immunity to scaling and/or offset mismatch. The Pearson distance between the received vector $\mathbf{r}$ and a codeword $\hat{\mathbf{x}} \in \mathcal{S}$ is defined as

$$L_p(\mathbf{r}, \hat{\mathbf{x}}) = 1 - \rho_{\mathbf{r},\hat{\mathbf{x}}}, \tag{4}$$

where $\rho_{\mathbf{r},\hat{\mathbf{x}}}$ is the Pearson correlation coefficient

$$\rho_{\mathbf{r},\hat{\mathbf{x}}} = \frac{\langle \mathbf{r} - \bar{\mathbf{r}}\mathbf{1}, \hat{\mathbf{x}} - \bar{\hat{\mathbf{x}}}\mathbf{1} \rangle}{\sigma_{\mathbf{r}}\sigma_{\hat{\mathbf{x}}}}. \tag{5}$$

A Pearson decoder chooses a codeword which minimizes this distance. As shown in [12], a modified Pearson distance-based criterion leading to the same result in the minimization process reads

$$L'_p(\mathbf{r}, \hat{\mathbf{x}}) = \sum_{i=1}^{n} (r_i - \hat{x}_i + \bar{\hat{\mathbf{x}}})^2, \tag{6}$$

if there is no scaling mismatch, i.e., $a = 1$. Use of the Pearson distance requires that the set of codewords satisfies certain special properties [12].

A geometric meaning for Pearson distance is provided in [14]. Since the offset b changes the mean of a vector, it seems reasonable to consider normalized vectors $\hat{\mathbf{x}} - \bar{\hat{\mathbf{x}}}\mathbf{1}$ and $\mathbf{r} - \bar{\mathbf{r}}\mathbf{1}$ rather than $\hat{\mathbf{x}}$ and $\mathbf{r}$. On the other hand, scaling a vector of mean 0 by a only changes its standard deviation by a factor of a. So it seems reasonable to scale the normalized vectors so that they have standard deviation 1. It is not difficult to show that this is $\rho_{\mathbf{r},\hat{\mathbf{x}}}$.

III. Maximum Likelihood Decoding

If a vector $\mathbf{r}$ is received, optimum decoding must determine a codeword $\hat{\mathbf{x}} \in \mathcal{S}$ maximizing $\mathrm{P}(\hat{\mathbf{x}} \,|\, \mathbf{r})$. If all codewords are equally likely to be sent, then, by Bayes Theorem, this scheme is equivalent to maximizing $\mathrm{P}(\mathbf{r} \,|\, \hat{\mathbf{x}})$, that is, the probability that $\mathbf{r}$ is received, given $\hat{\mathbf{x}}$ is sent.

From (2), we know $\mathbf{v} = (\mathbf{r} - b\mathbf{1})/a - \hat{\mathbf{x}}$ when a and b are fixed, and since a is nonzero, the likelihood $\mathrm{P}(\mathbf{r} \,|\, \hat{\mathbf{x}})$ in this case is

$$\phi((\mathbf{r} - b\mathbf{1})/a - \hat{\mathbf{x}}).$$

Here, we consider the situation that the scaling and the offset take their values within certain ranges, specifically $0 < a_1 \leq a \leq a_2$ and $b_1 \leq b \leq b_2$, but do not make any further

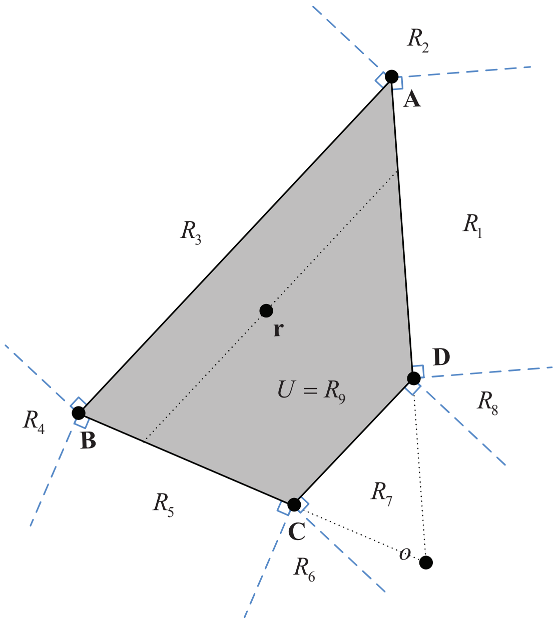

Fig. 1. Subdivision of $U' = \{c\mathbf{r} + d\mathbf{1} | c, d \in \mathbb{R}\}$.

assumptions on the distributions on these intervals. Thus, in order to achieve ML decoding, the criterion to maximize among all candidate codewords $\hat{\mathbf{x}}$ is

$$\max_{0<a_1\le a\le a_2, b_1\le b\le b_2} \phi((\mathbf{r} - b\mathbf{1})/a - \hat{\mathbf{x}}). \tag{7}$$

Since the logarithm function is strictly increasing on the positive real numbers and ϕ is a positive function, an equivalent formulation of the problem is to find $\hat{\mathbf{x}} \in \mathcal{S}$ that maximizes

$$\max_{0<a_1\le a\le a_2, b_1\le b\le b_2} \log \phi((\mathbf{r} - b\mathbf{1})/a - \hat{\mathbf{x}}).$$

Since

$$\begin{aligned}\log \phi((\mathbf{r} - b\mathbf{1})/a - \hat{\mathbf{x}}) &= -n \log(\sigma\sqrt{2\pi}) \\ &\quad - \frac{1}{2\sigma^2}\sum_{i=1}^{n}((r_i - b)/a - \hat{x}_i)^2\end{aligned} \tag{8}$$

has a component $-n\log(\sigma\sqrt{2\pi})$ that is independent of $\hat{\mathbf{x}}$ and $\mathbf{r}$, and since $\frac{1}{2\sigma^2}$ is a positive constant, a maximum likelihood decoder finds a codeword $\hat{\mathbf{x}}$ that minimizes

$$\min_{0<a_1\le a\le a_2, b_1\le b\le b_2} \sum_{i=1}^{n}((r_i - b)/a - \hat{x}_i)^2,$$

i.e., it minimizes the squared Euclidean distance between the candidate codeword $\hat{\mathbf{x}}$ and the points in

$$U = \{(\mathbf{r} - b\mathbf{1})/a | 0 < a_1 \le a \le a_2, b_1 \le b \le b_2\},$$

which is a subset of the subspace

$$U' = \{c\mathbf{r} + d\mathbf{1} | c, d \in \mathbb{R}\}$$

in $\mathbb{R}^n$.

The squared Euclidean distance between a vector $\hat{\mathbf{x}}$ and the set U is defined as

$$L_e(U, \hat{\mathbf{x}}) = \sum_{i=1}^{n}(p_i - \hat{x}_i)^2,$$

where $\mathbf{p} = (p_1, p_2, \ldots, p_n)$ is the point in U that is closest to $\hat{\mathbf{x}}$. The most likely candidate codeword $\mathbf{x}_o$ for a received vector has the smallest $L_e(U, \hat{\mathbf{x}})$, that is

$$\mathbf{x}_o = \arg\min_{\hat{\mathbf{x}}\in\mathcal{S}} L_e(U, \hat{\mathbf{x}}). \tag{9}$$

Hence, $\hat{\mathbf{x}} \in \mathcal{S}$ closest to U is chosen as the ML decoder output.

In order to calculate $L_e(U, \hat{\mathbf{x}})$ for a codeword $\hat{\mathbf{x}}$, we first find the point in U' that is closest to $\hat{\mathbf{x}}$ and then check if this point is in U. Applying the first derivative test gives that the closest point in U' to $\hat{\mathbf{x}}$ is $\mathbf{p}_0 = c_0\mathbf{r} + d_0\mathbf{1}$ with

$$c_0 = \frac{\langle \mathbf{r}, \hat{\mathbf{x}}\rangle - n\bar{\mathbf{r}}\bar{\hat{\mathbf{x}}}}{\langle \mathbf{r}, \mathbf{r}\rangle - n\bar{\mathbf{r}}^2}$$

and

$$d_0 = \frac{\langle \mathbf{r}, \mathbf{r}\rangle \bar{\hat{\mathbf{x}}} - \langle \mathbf{r}, \hat{\mathbf{x}}\rangle \bar{\mathbf{r}}}{\langle \mathbf{r}, \mathbf{r}\rangle - n\bar{\mathbf{r}}^2}.$$

In Fig. 1, we depict the subset U in gray when $a_1 < 1 < a_2$ and $b_1 < 0 < b_2$. Four vertices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ are also shown in the picture:

$$\mathbf{A} = (\mathbf{r} - b_1\mathbf{1})/a_1,$$
$$\mathbf{B} = (\mathbf{r} - b_2\mathbf{1})/a_1,$$
$$\mathbf{C} = (\mathbf{r} - b_2\mathbf{1})/a_2,$$
$$\mathbf{D} = (\mathbf{r} - b_1\mathbf{1})/a_2.$$

Perpendicular lines (blue dash) in U' to sides of U through vertices are pictured in Fig. 1. These perpendicular lines and the sides of U separate U' into 9 subsets, namely, $R_1, R_2, \ldots, R_9$. For instance, the perpendicular lines to side $\mathbf{BC}$ and $\mathbf{BC}$ itself form the boundaries of R_5. We use the notation R_9 in Fig. 1 for the subset U for clerical convenience.

Theorem 1. *If $\mathbf{p}_0$ is in the subset R_i, $i = 1, \ldots, 9$, then the closest point in U to $\hat{\mathbf{x}}$ is*

$$\mathbf{p} = \begin{cases} \dfrac{\langle \mathbf{r} - b_1\mathbf{1}, \hat{\mathbf{x}}\rangle}{\|\mathbf{r} - b_1\mathbf{1}\|^2}(\mathbf{r} - b_1\mathbf{1}) & \text{if } i = 1, \\ \dfrac{\langle \mathbf{r} - b_2\mathbf{1}, \hat{\mathbf{x}}\rangle}{\|\mathbf{r} - b_2\mathbf{1}\|^2}(\mathbf{r} - b_2\mathbf{1}) & \text{if } i = 5, \\ (\mathbf{r} - (\bar{\mathbf{r}} - a_1\bar{\hat{\mathbf{x}}})\mathbf{1})/a_1 & \text{if } i = 3, \\ (\mathbf{r} - (\bar{\mathbf{r}} - a_2\bar{\hat{\mathbf{x}}})\mathbf{1})/a_2 & \text{if } i = 7, \\ \mathbf{A} & \text{if } i = 2, \\ \mathbf{B} & \text{if } i = 4, \\ \mathbf{C} & \text{if } i = 6, \\ \mathbf{D} & \text{if } i = 8, \\ \mathbf{p}_0 & \text{if } i = 9. \end{cases} \tag{10}$$

The ML decoding criterion is minimizing $L_e(\mathbf{p}, \hat{\mathbf{x}})$ among all candidate codewords.

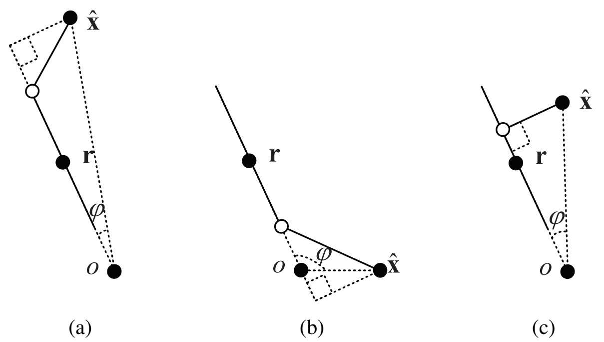

Fig. 2. The distance of a candidate codeword $\hat{\mathbf{x}}$ to the subset $\{\mathbf{r}/a\,|0 < a_1 \leq a \leq a_2\}$: three cases in (11), (a) $\langle \mathbf{r}, \hat{\mathbf{x}} \rangle > \langle \mathbf{r}, \mathbf{r} \rangle / a_1$, (b) $\langle \mathbf{r}, \hat{\mathbf{x}} \rangle < \langle \mathbf{r}, \mathbf{r} \rangle / a_2$ and (c) otherwise, assuming $a_1 < 1 < a_2$.

Proof. If $\mathbf{p}_0$ is in the subset R_1, maximizing (7) is equivalent to minimizing the smallest squared Euclidean distance from the codeword $\hat{\mathbf{x}}$ to the line segment

$$\mathbf{AD} = \{(\mathbf{r} - b_1\mathbf{1})/a | 0 < a_1 \leq a \leq a_2\},$$

which is shown in Fig. 1. Let θ be the angle between $\hat{\mathbf{x}}$ and $\mathbf{r} - b_1\mathbf{1}$. The point on $\mathbf{AD}$ closest to $\hat{\mathbf{x}}$ is $\mathbf{p} = \alpha(\mathbf{r} - b_1\mathbf{1})$ with

$$\alpha = (\|\hat{\mathbf{x}}\| \cos\theta)/\|\mathbf{r} - b_1\mathbf{1}\| = \langle \mathbf{r} - b_1\mathbf{1}, \hat{\mathbf{x}} \rangle / \|\mathbf{r} - b_1\mathbf{1}\|^2.$$

Similarly, when $\mathbf{p}_0$ is in the subset R_5, the point on $\mathbf{BC} = \{(\mathbf{r} - b_2\mathbf{1})/a | 0 < a_1 \leq a \leq a_2\}$ closest to $\hat{\mathbf{x}}$ is $\mathbf{p} = \alpha(\mathbf{r} - b_2\mathbf{1})$ with

$$\alpha = \langle \mathbf{r} - b_2\mathbf{1}, \hat{\mathbf{x}} \rangle / \|\mathbf{r} - b_2\mathbf{1}\|^2.$$

If $\mathbf{p}_0$ is in the subset R_3, the point $\mathbf{p} \in U$ that is closest to $\hat{\mathbf{x}}$ must be on the line segment

$$\mathbf{AB} = \{(\mathbf{r} - b\mathbf{1})/a_1 | b_1 \leq b \leq b_2\},$$

which is shown in Fig. 1. The point on $\mathbf{AB}$ that is closest to $\hat{\mathbf{x}}$ is $\mathbf{p} = (\mathbf{r} - \beta\mathbf{1})/a_1$, with $\beta = \bar{\mathbf{r}} - a_1\bar{\hat{\mathbf{x}}}$, which follows from the first derivative test. The proof is similar when $\mathbf{p}_0$ is in the subset R_7, with the line segment $\mathbf{CD}$ taking the role of $\mathbf{AB}$.

If $\mathbf{p}_0$ is in the subset R_2, then the closest point in U to $\hat{\mathbf{x}}$ is the vertex $\mathbf{A} = (\mathbf{r} - b_1\mathbf{1})/a_1$, as can be observed from Fig. 1. Similar results are found for the situations that $\mathbf{p}_0$ is in the subset R_4, R_6, and R_8, where the closest point in U to $\hat{\mathbf{x}}$ is $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{D}$, respectively.

Obviously, the closest point in U to $\hat{\mathbf{x}}$ is $\mathbf{p}_0$ itself when $\mathbf{p}_0$ is in the subset $R_9 = U$. □

IV. Special Cases

Several special values of a_1, a_2, b_1 and b_2 are considered, leading to typical cases for maximizing (7); these include the scaling-only and offset-only cases. Not only ML decoding criteria are discussed, but also conventional decoding criteria as introduced in Section II.

A. Scaling-Only Case

In the scaling-only case, i.e., $b = 0$, we simply have

$$\mathbf{r} = a(\mathbf{x} + \mathbf{v}),$$

where the scaling, a, is unknown to both sender and receiver.

In Theorem 2 of [13], the following ML criterion was presented for the case that there is bounded scaling ($0 < a_1 \leq a \leq a_2$) and no offset mismatch ($b = 0$):

$$L_{a_1,a_2}(\mathbf{r}, \hat{\mathbf{x}}) = \begin{cases} L_e(\mathbf{r}/a_1, \hat{\mathbf{x}}) & \text{if } \langle \mathbf{r}, \hat{\mathbf{x}} \rangle > \langle \mathbf{r}, \mathbf{r} \rangle / a_1, \\ L_e(\mathbf{r}/a_2, \hat{\mathbf{x}}) & \text{if } \langle \mathbf{r}, \hat{\mathbf{x}} \rangle < \langle \mathbf{r}, \mathbf{r} \rangle / a_2, \\ \|\hat{\mathbf{x}}\|^2 - \left(\frac{\langle \mathbf{r}, \hat{\mathbf{x}} \rangle}{\|\mathbf{r}\|}\right)^2 & \text{otherwise.} \end{cases} \tag{11}$$

This result can also be simply found from the general framework presented in the previous section, by setting $b_1 = b_2 = 0$ in Theorem 1. Note that this gives indeed that $\mathbf{p} = \mathbf{r}/a_1$ if $\mathbf{p}_0 \in R_2 \cup R_3 \cup R_4$, which corresponds to the situation that

$$\frac{\|\hat{\mathbf{x}}\| \cos\varphi}{\|\mathbf{r}\|} = \frac{\langle \mathbf{r}, \hat{\mathbf{x}} \rangle}{\langle \mathbf{r}, \mathbf{r} \rangle} > 1/a_1,$$

where φ is the angle between $\hat{\mathbf{x}}$ and $\mathbf{r}$. Similarly, note that $\mathbf{p} = \mathbf{r}/a_2$ if $\mathbf{p}_0 \in R_6 \cup R_7 \cup R_8$, which corresponds to the situation that

$$\frac{\|\hat{\mathbf{x}}\| \cos\varphi}{\|\mathbf{r}\|} = \frac{\langle \mathbf{r}, \hat{\mathbf{x}} \rangle}{\langle \mathbf{r}, \mathbf{r} \rangle} < 1/a_2.$$

Finally, note that $\mathbf{p} = \frac{\langle \mathbf{r}, \hat{\mathbf{x}} \rangle}{\|\mathbf{r}\|^2}\mathbf{r}$ if $\mathbf{p}_0 \in R_1 \cup R_5 \cup R_9$, which corresponds to the 'otherwise' case in (11), and that

$$L_e(\mathbf{p}, \hat{\mathbf{x}}) = L_e\left(\frac{\langle \mathbf{r}, \hat{\mathbf{x}} \rangle}{\|\mathbf{r}\|^2}\mathbf{r}, \hat{\mathbf{x}}\right) = \|\hat{\mathbf{x}}\|^2 - \left(\frac{\langle \mathbf{r}, \hat{\mathbf{x}} \rangle}{\|\mathbf{r}\|}\right)^2.$$

In Fig. 2, we draw the three cases in (11), where the subset $\{\mathbf{r}/a\,|0 < a_1 \leq a \leq a_2\}$ is a line segment in the direction of $\mathbf{r}$. The circle points are the closest points on this line segment to $\hat{\mathbf{x}}$.

Next, we consider the situation that $a_1 \to 0$ and $a_2 \to \infty$, i.e., the only knowledge on the gain a is that it is a positive number, without further limitations. The subset $\{\mathbf{r}/a\,|a \in \mathbb{R}, a > 0\}$ is a ray from the origin in the direction of $\mathbf{r}$. In this case, it follows from the above that ML decoding can be achieved by minimizing

$$L_a(\mathbf{r}, \hat{\mathbf{x}}) = \begin{cases} \|\hat{\mathbf{x}}\|^2 - \left(\frac{\langle \mathbf{r}, \hat{\mathbf{x}} \rangle}{\|\mathbf{r}\|}\right)^2 & \text{if } \frac{\langle \mathbf{r}, \hat{\mathbf{x}} \rangle}{\langle \mathbf{r}, \mathbf{r} \rangle} > 0, \\ \|\hat{\mathbf{x}}\|^2 & \text{otherwise.} \end{cases} \tag{12}$$

One reason for this choice is that it behaves well with respect to an affine scaling function ($a > 0$), since

$$L_a(\mathbf{r}, \hat{\mathbf{x}}) = L_a(\mathbf{r}/a, \hat{\mathbf{x}}).$$

That is, scaling a vector $\mathbf{r}$ by a does not change the angle φ between $\hat{\mathbf{x}}$ and $\mathbf{r}$.

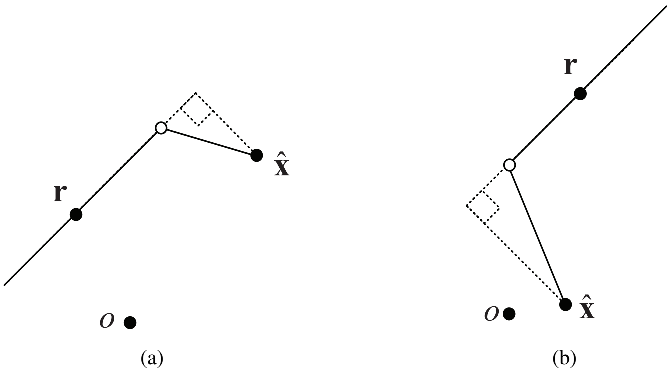

Fig. 3. The distance of a candidate codeword $\hat{\mathbf{x}}$ to the line segment $\{\mathbf{r} - b\mathbf{1} \,|b_1 \le b \le b_2\}$: two cases in (13), (a) $\bar{\mathbf{r}} - \bar{\hat{\mathbf{x}}} < b_1$ and (b) $\bar{\mathbf{r}} - \bar{\hat{\mathbf{x}}} > b_2$, assuming $b_1 < 0 < b_2$.

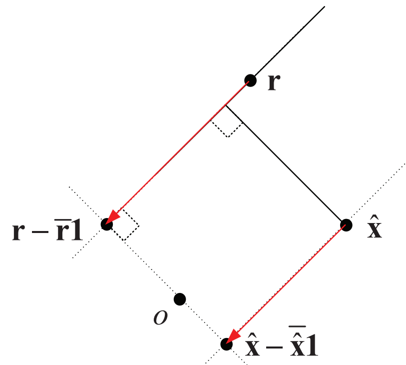

Fig. 4. The distance of a candidate codeword $\hat{\mathbf{x}}$ to the line segment $\{\mathbf{r} - b\mathbf{1} \,|b_1 \le b \le b_2\}$ for the 'otherwise' case in (13), assuming $b_1 < 0 < b_2$.

B. Offset-Only Case

In the offset-only case, i.e., $a = 1$, we simply have

$$\mathbf{r} = \mathbf{x} + \mathbf{v} + b\mathbf{1},$$

where the offset b is unknown to both sender and receiver.

In Theorem 1 of [13], the following ML criterion was presented for the case that $a = 1$ and $b_1 \le b \le b_2$:

$$L_{b_1,b_2}(\mathbf{r}, \hat{\mathbf{x}}) = \begin{cases} L_e(\mathbf{r} - b_1\mathbf{1}, \hat{\mathbf{x}}) & \text{if } \bar{\mathbf{r}} - \bar{\hat{\mathbf{x}}} < b_1, \\ L_e(\mathbf{r} - b_2\mathbf{1}, \hat{\mathbf{x}}) & \text{if } \bar{\mathbf{r}} - \bar{\hat{\mathbf{x}}} > b_2, \\ L_e(\mathbf{r} - (\bar{\mathbf{r}} - \bar{\hat{\mathbf{x}}})\mathbf{1}, \hat{\mathbf{x}}) & \text{otherwise.} \end{cases} \tag{13}$$

This result also follows from the general setting presented in the previous section, by substituting $a_1 = a_2 = 1$. Note that the first case in (13) corresponds to the situation that $\mathbf{p}_0 \in R_1 \cup R_2 \cup R_8$, the second case to $\mathbf{p}_0 \in R_4 \cup R_5 \cup R_6$, and the last case to $\mathbf{p}_0 \in R_3 \cup R_7 \cup R_9$.

We illustrate the first two situations of $L_{b_1,b_2}(\mathbf{r}, \hat{\mathbf{x}})$ in Fig. 3 and the last one in Fig. 4, where $\{\mathbf{r} - b\mathbf{1} \,|b_1 \le b \le b_2\}$ is shown by a line segment passing through $\mathbf{r}$ with direction $\mathbf{1}$. The point in $\{\mathbf{r}-b\mathbf{1} \,|b_1 \le b \le b_2\}$ that is closest to $\hat{\mathbf{x}}$ is $\mathbf{r}-b_1\mathbf{1}$ or $\mathbf{r}-b_2\mathbf{1}$ for the situations in Fig. 3. For the 'otherwise' case in (13), we consider in Fig. 4 the normalized vectors $\hat{\mathbf{x}} - \bar{\hat{\mathbf{x}}}\mathbf{1}$ and $\mathbf{r} - \bar{\mathbf{r}}\mathbf{1}$ rather than $\hat{\mathbf{x}}$ and $\mathbf{r}$.

By letting $b_1 \to -\infty$ and $b_2 \to \infty$, we obtain from (13) that the criterion

$$\begin{aligned} L_b(\mathbf{r}, \hat{\mathbf{x}}) &= L_e(\mathbf{r} - (\bar{\mathbf{r}} - \bar{\hat{\mathbf{x}}})\mathbf{1}, \hat{\mathbf{x}}) \\ &= \sum_{i=1}^{n} (r_i - \hat{x}_i + \bar{\hat{\mathbf{x}}})^2 - n\bar{\mathbf{r}}^2 \\ &= L'_p(\mathbf{r}, \hat{\mathbf{x}}) - n\bar{\mathbf{r}}^2, \end{aligned}$$

when there is no knowledge at all of the magnitude of the offset [13]. Noting that the last term $n\bar{\mathbf{r}}^2$ is irrelevant in the minimization process, we conclude that the modified Pearson criterion $L'_p(\mathbf{r}, \hat{\mathbf{x}})$ achieves ML decoding in this case.

C. Unbounded Scaling and Offset Case

In this subsection, an ML decoding criterion derived by Blackburn [14] for the situation when both the scaling a and the offset b are unbounded ($a_1 \to 0$, $a_2 \to \infty$, $b_1 \to -\infty$, $b_2 \to \infty$) is reconsidered as a special case of the results presented in Section III. In [14], Blackburn shows that an ML decoder chooses a codeword $\hat{\mathbf{x}}$ minimizing

$$l_{\mathbf{r}}(\hat{\mathbf{x}}) = \begin{cases} \sigma_{\hat{\mathbf{x}}}^2(1 - \rho_{\mathbf{r},\hat{\mathbf{x}}}^2) & \text{when } \rho_{\mathbf{r},\hat{\mathbf{x}}} > 0, \\ \sigma_{\hat{\mathbf{x}}}^2 & \text{otherwise.} \end{cases} \tag{14}$$

His argument was that when the scaling factor a and the offset term b are fully unknown, except for the sign of a, then maximizing (7) is equivalent to minimizing the smallest squared Euclidean distance from the codeword $\hat{\mathbf{x}}$ to the subset

$$U^+ = \{(\mathbf{r} - b\mathbf{1})/a | a, b \in \mathbb{R}, a > 0\},$$

which is a half-subspace of $U' \subset \mathbb{R}^n$. Note that when $a_1 \to 0$, $a_2 \to \infty$, $b_1 \to -\infty$, $b_2 \to \infty$, our U is indeed equal to Blackburn's set U^+. Note that $\mathbf{p}_0 = c_0\mathbf{r} + d_0\mathbf{1}$ is either in $R_9 = U = U^+$ or in R_7. By (5), c_0 and d_0 can be rewritten as

$$c_0 = \frac{\rho_{\mathbf{r},\hat{\mathbf{x}}}\sigma_{\hat{\mathbf{x}}}}{\sigma_{\mathbf{r}}} \tag{15}$$

and

$$d_0 = \bar{\hat{\mathbf{x}}} - c_0\bar{\mathbf{r}}. \tag{16}$$

In case $\mathbf{p}_0 \in R_9$, which happens if and only if $\rho_{\mathbf{r},\hat{\mathbf{x}}} > 0$, then Theorem 1 says $\mathbf{p} = \mathbf{p}_0 = c_0\mathbf{r} + d_0\mathbf{1}$. Note that

$$\begin{aligned} &L_e(c_0\mathbf{r} + d_0\mathbf{1}, \hat{\mathbf{x}}) \\ &= \sum_{i=1}^{n} [c_0 r_i + d_0 - \hat{x}_i]^2 \\ &= \sum_{i=1}^{n} \left[c_0(r_i - \bar{\mathbf{r}}) - (\hat{x}_i - \bar{\hat{\mathbf{x}}})\right]^2 \\ &= \sum_{i=1}^{n} \left[c_0^2(r_i - \bar{\mathbf{r}})^2 - 2c_0(r_i - \bar{\mathbf{r}})(\hat{x}_i - \bar{\hat{\mathbf{x}}}) + (\hat{x}_i - \bar{\hat{\mathbf{x}}})^2\right] \\ &= c_0^2\sigma_{\mathbf{r}}^2 - 2c_0\rho_{\mathbf{r},\hat{\mathbf{x}}}\sigma_{\mathbf{r}}\sigma_{\hat{\mathbf{x}}} + \sigma_{\hat{\mathbf{x}}}^2 \\ &= \left(\frac{\rho_{\mathbf{r},\hat{\mathbf{x}}}\sigma_{\hat{\mathbf{x}}}}{\sigma_{\mathbf{r}}}\right)^2 \sigma_{\mathbf{r}}^2 - 2\left(\frac{\rho_{\mathbf{r},\hat{\mathbf{x}}}\sigma_{\hat{\mathbf{x}}}}{\sigma_{\mathbf{r}}}\right) \rho_{\mathbf{r},\hat{\mathbf{x}}}\sigma_{\hat{\mathbf{x}}}\sigma_{\mathbf{r}} + \sigma_{\hat{\mathbf{x}}}^2 \\ &= \sigma_{\hat{\mathbf{x}}}^2(1 - \rho_{\mathbf{r},\hat{\mathbf{x}}}^2), \end{aligned}$$

which is indeed the same as in (14) when $\rho_{\mathbf{r},\hat{\mathbf{x}}} > 0$.

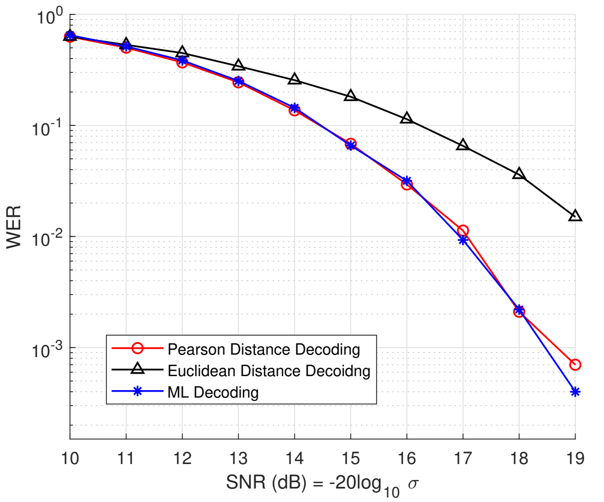

Fig. 5. Word error rate (WER) against signal-to-noise ratio (SNR) when $q = 4$, $n = 8$, $a = 1.07$, and $b = 0.07$.

In case $\mathbf{p}_0 \in R_7$, then Theorem 1 says $\mathbf{p} = \bar{\hat{\mathbf{x}}}\mathbf{1}$ since $a_2 \to \infty$. Hence,

$$L_e(\mathbf{p}, \hat{\mathbf{x}}) = L_e(\bar{\hat{\mathbf{x}}}\mathbf{1}, \hat{\mathbf{x}}) = \sigma^2_{\hat{\mathbf{x}}}.$$

This shows that Blackburn's criterion (14) indeed appears as a special case of our general setting.

D. Simulation Results

Thus far, we have discussed ML decoding for Gaussian noise channels with scaling and offset mismatch, and have mentioned that Euclidean distance decoding is ML decoding for Gaussian noise channels in Section II, while the Pearson distance criterion (4) is optimal for channels with scaling and offset mismatch, due to its intrinsic immunity to both scaling and offset mismatch.

Figure 5 shows simulation results of Pearson distance decoding, Euclidean distance decoding, and ML decoding (14) when $q = 4$ and $n = 8$. The word error rate (WER) of 10,000 trials is shown as a function of the signal-to-noise ratio (SNR $= -20\log_{10}\sigma$). Results are given for 2-constrained codes [12], [15], while $a = 1.07$ and $b = 0.07$. The simulations indicate that for this case Pearson distance decoding has a comparable performance as ML decoding, while Euclidean distance decoding performs considerably worse.

V. Conclusion

We have derived a maximum likelihood decoding criterion for multi-level cell memories with Gaussian noise and scaling and/or offset mismatch. In our channel model, scaling and offset are restricted to certain ranges, $0 < a_1 \le a \le a_2$ and $b_1 \le b \le b_2$, which is a generalization of several prior art settings. For instance, by letting $a_1 \to 0$, $a_2 \to \infty$, $b_1 \to -\infty$, $b_2 \to \infty$, we obtain the same ML decoding criterion as proposed by Blackburn for the case of unbounded gain and offset. We also provided geometric interpretations illustrating the main ideas.

Scaling and offset mismatch are important issues in multi-level cell memories, but not the only ones. As future work, one could try to derive ML decoding criteria for multi-level cell memories for which the channel model includes dependent noise and/or inter-cell interference as well.

References

[1] S. Aritome, *NAND Flash Memory Technologies*. Hoboken, NJ, USA: John Wiley & Sons, Inc., 2015, ch. 4.

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

[3] W. Liu, J. Rho, and W. Sung, "Low-power high-throughput BCH error correction VLSI design for multi-level cell NAND flash memories," in *Proc. IEEE Workshop on Signal Processing Systems Design and Implementation (SIPS'06)*, Banff, Canada, 2006, pp. 303–308.

[4] B. Chen, X. Zhang, and Z. Wang, "Error correction for multi-level NAND flash memory using Reed-Solomon codes," in *Proc. IEEE Workshop on Signal Processing Systems*, Washington, USA, 2008, pp. 94–99.

[5] F. Zhang, H. D. Pfister, and A. Jiang, "LDPC codes for rank modulation in flash memories," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Austin, Texas, USA, 2010, pp. 859–863.

[6] S. Solda, D. Vogrig, A. Bevilacqua, A. Gerosa, and A. Neviani, "Analog decoding of trellis coded modulation for multi-level flash memories," in *Proc. IEEE Int. Symp. Circuits and Systems (ISCAS)*, Seattle, WA, USA, 2008, pp. 744–747.

[7] G. Atwood, A. Fazio, D. Mills, and B. Reaves, "Intel strataflash memory technology overview," *Intel Technology Journal*, pp. 1–8, 1997.

[8] X. Huang, A. Kavcic, G. D. X. Ma, and T. Zhang, "Multilevel flash memories: channel modeling, capacities and optimal coding rates," *Int. Journal on Advances in Systems and Measurements*, vol. 6, no. 3, pp. 364–373, 2013.

[9] A. Jiang, R. Mateescu, M. Schwartz, and J. Bruck, "Rank modulation for flash memories," *IEEE Trans. Inf. Theory*, vol. 55, no. 6, pp. 2659–2673, Jun. 2009.

[10] H. Zhou and J. Bruck, "Balanced modulation for nonvolatile memories," *arXiv preprint arXiv: 1209.0744*, Sep. 2012.

[11] K. A. S. Immink and K. Cai, "Composition check codes," *IEEE Trans. Inf. Theory*, vol. 64, no. 1, pp. 249–256, Jan. 2018.

[12] K. A. S. Immink and J. H. Weber, "Minimum Pearson distance detection for multilevel channels with gain and/or offset mismatch," *IEEE Trans. Inf. Theory*, vol. 60, no. 10, pp. 5966–5974, Oct. 2014.

[13] J. H. Weber and K. A. S. Immink, "Maximum likelihood decoding for Gaussian noise channels with gain or offset mismatch," *IEEE Commun. Lett.*, vol. 22, no. 6, pp. 1128–1131, Jun. 2018.

[14] S. R. Blackburn, "Maximum likelihood decoding for multilevel channels with gain and offset mismatch," *IEEE Trans. Inf. Theory*, vol. 62, no. 3, pp. 1144–1149, Mar. 2016.

[15] J. H. Weber, K. A. S. Immink, and S. R. Blackburn, "Pearson codes," *IEEE Trans. Inf. Theory*, vol. 62, no. 1, pp. 131–135, Jan. 2016.