Lattice-Based WOM Codes for Multilevel Flash Memories

Aman Bhatia, Minghai Qin, Aravind R. Iyengar, Brian M. Kurkoski, and Paul H. Siegel

***Abstract*—We consider t-write codes for write-once memories with n cells that can store multiple levels. Assuming an underlying lattice-based construction and using the continuous approximation, we derive upper bounds on the worst-case sum-rate optimal and fixed-rate optimal n-cell t-write write-regions for the asymptotic case of continuous levels. These are achieved using hyperbolic shaping regions that have a gain of 1 bit/cell over cubic shaping regions. Motivated by these hyperbolic write-regions, we discuss construction and encoding of codebooks for cells with discrete support. We present a polynomial-time algorithm to assign messages to the codebooks and show that it achieves the optimal sum-rate for any given codebook when $n = 2$. Using this approach, we construct codes that achieve high sum-rate. We describe an alternative formulation of the message assignment problem for $n \geq 3$, a problem which remains open.**

***Index Terms*—Write-Once Memories, WOM Codes, Rewriting Codes, Flash Memories, Lattices**

I. Introduction

THE STUDY of write-once memory (WOM) codes was motivated by various applications in the field of data storage [1], [2]. More, recently, they have been proposed as a lifetime-enhancing method for flash memories. Flash memory stores information in the form of charge in a floating gate transistor, referred to as a *cell*. While increasing the charge level of an individual cell is a simple *program* operation with low latency, decreasing the level of a cell requires a complex *erase* operation on a large block of cells that also decreases the lifetime of the device. WOM codes can be used to write information multiple times before an erase operation is required, thereby enhancing the lifetime of the device [3], [4], [5].

Considerable progress has been made in the last few years in the study of binary WOM code constructions for single-level cell (SLC) flash memory devices where each cell supports $q = 2$ levels [3], [5], [6]. However, to increase storage densities, future flash memory devices are expected to support a large number of cell levels, continuing the trend seen with the use of MLC and TLC devices that support 4 and 8 levels, respectively. This has motivated a body of work on the construction of WOM codes for multilevel cells that support $q \geq 3$ levels [7], [8], [9], [10], [11]. The capacity of rewrite codes for t writes on q-ary cells is known to be $\log_2 \binom{q+t-1}{q-1}$ [12], although explicit characterization of the capacity region remains an open problem.

In this paper, we consider the construction of lattice-based WOM codes for t writes on q-level cells. Lattice-based 2-write WOM codes over n cells in the asymptotic setting of continuous cell levels were derived in [11] for the fixed-rate scenario, where the cardinality of the message set is the same on each write. Allowing the cardinality of the message set on each write to be different can increase the sum-rate. Using a continuous approximation approach, it was hypothesized in [11] that the hyperbolic shaping regions were optimal for maximizing sum-rates of two writes over lattices in arbitrary dimensions. Optimality of hyperbolic shaping regions was proven in [13] for lattices in $n = 2$ dimensions when the number of writes, t, is arbitrary. A proof of optimality was provided in [14] for the case of an arbitrary number of cells, n, and $t = 2$ writes.

Here, we consider the most general case of an arbitrary number of writes on an arbitrary number of cells where each cell supports a large number of levels. Using the continuous approximation approach we prove that hyperbolic shaping regions are optimal for maximizing the sum-rate. The results are then extended to the fixed-rate case, a scenario of practical importance. We also discuss the problem of encoding for these codes. Encoding requires a consistent assignment of messages to cell levels that does not depend on the state of the memory after the previous write. A consistent message assignment algorithm is optimal if, for any given codebook, the message set cardinality is the largest possible. We show that a consistent assignment scheme for a given codebook and message set cardinality may not always exist when $n > 2$. The problem of determining the existence of a consistent assignment for arbitrary n is shown to be an instance of an NP-complete problem, but with additional structure introduced by the geometry of the shaping regions. We exploit this structure to find an optimal linear-time algorithm for the case where $n = 2$. For $n \geq 3$, we use a sub-optimal algorithm for finding consistent assignments. Using these results, we construct codes that achieve high sum-rates for memories with multilevel cells.

Manuscript received May 15, 2013; revised October 1, 2013 and December 10, 2013. B. M. Kurkoski's work was supported by the Ministry of Education, Science, Sports and Culture; Grant-in-Aid for Scientific Research (C) number 23560439. The work of other authors was supported by National Science Foundation Grant CCF-1116739, UC Lab Fees Research Program Award No. 09-LR-06-118620-SIEP and the Center for Magnetic Recording Research. This paper was presented in part at the 34th Symposium on Information Theory and Applications, Ousyuku, Iwate, Japan, Nov. 29 – Dec. 2 2011; the Non-Volatile Memories Workshop, San Diego, CA, USA, Mar. 4–6, 2012; the 2012 IEEE Information Theory Workshop, Lausanne, Switzerland, Sept. 3–7, 2012 and the IEEE International Symposium on Information Theory and its Applications, Honolulu, HI, USA, Oct. 28–31, 2012.

A. Bhatia, M. Qin, and P. H. Siegel are with the University of California, San Diego, La Jolla, CA 92093-0401, USA (e-mail: {amanbh, mqin, psiegel}@ucsd.edu).

A. R. Iyengar was with the University of California, San Diego. He is now with Qualcomm Technologies Inc., Santa Clara, CA 95051, USA (e-mail: ariyengar@qti.qualcomm.com).

B. M. Kurkoski is with the Japan Advanced Institute of Science and Technology, Nomi, Ishikawa 923-1292, Japan (e-mail: kurkoski@jaist.ac.jp).

Digital Object Identifier 10.1109/JSAC.2014.140513.

The rest of the paper is organized as follows. Section II formulates the code design problem. In Section III, we extend ideas presented in [11] and invoke the continuous approximation to obtain an upper bound on the worst-case sum-rate optimal t-write regions for n cells and consider its asymptotic behavior as the number of cells grows large. In Section IV, we derive the worst-case fixed-rate optimal t-write regions for n cells. Section V shows that the contribution of the lattice and shaping regions to the upper bound are separable, and the shaping gain of hyperbolic shaping regions is given. In Section VI, we discuss the case where the cells support a finite number of levels and propose methods for assignment of messages to cell levels. Finally, we present our conclusions in Section VII. Proofs of the lemmas and propositions appearing in Sections III, IV and VI are given in the appendices.

II. Lattice-Based WOM Codes

A. Lattices, Lattice Codes and Flash Memories

An n-dimensional lattice Λ is defined by an n-by-n generator matrix G. The lattice consists of the discrete set of points $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ for which

$$\mathbf{x} = \mathbf{b} \cdot G, \tag{1}$$

where $\mathbf{b} = (b_1, b_2, \ldots, b_n) \in \mathbb{Z}^n$ is any n-dimensional integer-valued vector.

The Voronoi region of a point $\mathbf{x} \in \Lambda$ is the set of points of $\mathbb{R}^n$ which are closer to $\mathbf{x}$ than to any other point $\mathbf{x}' \in \Lambda$. The volume of a Voronoi region $\mathrm{Vol}(\Lambda)$ is

$$\mathrm{Vol}(\Lambda) \triangleq |\det \mathrm{G}|. \tag{2}$$

A lattice code $\mathcal{L}$ is a finite subset of a lattice Λ, described by a shaping region $\mathbb{A} \subset \mathbb{R}^n$,

$$\mathcal{L} = \Lambda \cap \mathbb{A}. \tag{3}$$

For a thorough treatment of lattices, refer to [15].

The coordinate values of an n-dimensional lattice code may be stored in n cells of a flash memory. In the most general case, cell j stores a continuous value between 0 and ℓ_j, so that the stored values in n cells are represented by $\mathbf{x} \in [0, \ell_1] \times [0, \ell_2] \times \ldots [0, \ell_n] \triangleq \mathbb{A}$ where $x_j \in \mathbb{R}, \forall j = 1, 2, \ldots, n$. The volume of region $\mathbb{A}$ is $\|\mathbb{A}\| = \prod_{j=1}^{n} \ell_j$. Allowing arbitrary ℓ_j proves to be helpful in the sequel. However, in the typical case of a q-ary flash memory, we have $\ell_j = q - 1$ for all j, so $\mathbb{A} = [0, q-1]^n$ and the codebook is $\mathcal{L} = \mathbb{Z}_q^n$.

B. WOM Codebooks

Consider a flash memory device with n cells storing values $(x_1, x_2, \ldots, x_n) \in \mathcal{L}$ such that the level of a cell can only be increased during a write operation. We consider writing information in these cells t times before there is a need for a block erasure.

A *t-write WOM code* stores M_i messages in n cells in the worst case at write i, $i = 1, \ldots, t$. The instantaneous rate for write i and the *worst-case sum-rate* for the t-write code are

$$R_{i,t} = \frac{1}{n} \log_2 M_i \text{ bits per cell per write, and} \tag{4}$$

$$R_t = \sum_{i=1}^{t} R_{i,t} \text{ bits per cell per erase, respectively.} \tag{5}$$

A lattice-based t-write WOM codebook is defined by a partition of a lattice code $\mathcal{L}$ into t subsets, denoted as $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_t$. The subset $\mathcal{L}_i$ is the codebook for write i and has cardinality $|\mathcal{L}_i|$. Note that since these codes have disjoint codebooks for each write, they are a special case of WOM codes referred to as *synchronized WOM codes* in [1].

A point $\mathbf{x} \in \mathcal{L}$ is said to be *accessible* from another point $\mathbf{s}$, denoted as $\mathbf{x} \succ \mathbf{s}$, if $x_j \geq s_j$ for all $j = 1, 2, \ldots, n$ and $\mathbf{s} \neq \mathbf{x}$. Suppose the point stored at write $i-1$ is $\mathbf{s}$; then the set of points that may be stored at write i is

$$\mathcal{L}_i(\mathbf{s}) \triangleq \{\mathbf{x} \in \mathcal{L}_i : \mathbf{x} \succ \mathbf{s}\}. \tag{6}$$

Here, $\mathcal{L}_i(\mathbf{s})$, the subset of $\mathcal{L}_i$ accessible from $\mathbf{s}$, may be a proper subset of the codebook $\mathcal{L}_i$. Since the worst-case rate is of interest, define the *codebook cardinality*, denoted by C_i, as the minimum number of points in $\mathcal{L}_i$ that are accessible from any point in $\mathcal{L}_{i-1}$,

$$C_i \triangleq \min_{\mathbf{s} \in \mathcal{L}_{i-1}} |\mathcal{L}_i(\mathbf{s})|. \tag{7}$$

Also define the *total codebook cardinality*, Π_t, as

$$\Pi_t \triangleq \prod_{i=1}^{t} C_i. \tag{8}$$

The state of the memory before the first write is $\mathbf{s} = \mathbf{0}$ and all points in the codebook $\mathcal{L}_1$ are accessible. Thus, $M_1 = C_1$. However, the set of points that may be stored at any other write $i > 1$ depends on the point stored on the previous write. As a result, there may not exist a scheme which can consistently map C_i messages to points in the codebook $\mathcal{L}_i$. In some cases, then, M_i may be smaller than C_i,

$$M_i \leq C_i, \tag{9}$$

and accordingly each rate $R_{i,t}$ is upper bounded as

$$R_{i,t} \leq \frac{1}{n} \log_2 C_i, \tag{10}$$

and the worst-case sum-rate is upper bounded as

$$R_t \leq \frac{1}{n} \log_2 \Pi_t. \tag{11}$$

Parts of this paper concentrate on maximizing C_i and Π_t because they provide upper bounds on $R_{i,t}$ and R_t, respectively. The matter of consistent encoding-decoding is introduced in the next subsection and is discussed again in Section VI.

C. WOM Encoding and Decoding

The set of messages that can be stored on write i is $\mathcal{M}_i \triangleq \{1, 2, \ldots, M_i\}$. For given codebooks $\mathcal{L}_1, \mathcal{L}_2, \ldots, \mathcal{L}_t$, the encoder-decoder pair (ϕ, ψ) is

$$\phi : \bigcup_{i=1}^{t} \left(\mathcal{M}_i \times \mathcal{L}_{i-1}\right) \mapsto \mathcal{L}, \quad \psi : \mathcal{L} \mapsto [t] \times \bigcup_{i=1}^{t} \mathcal{M}_i \tag{12}$$

where $[t] \triangleq \{1, 2, \ldots, t\}$ which satisfy

$$\phi(m, \mathbf{x}) \in \mathcal{L}_i(\mathbf{x}) \quad \forall\, m \in \mathcal{M}_i, \mathbf{x} \in \mathcal{L}_{i-1}, \tag{13}$$

$$\psi(\mathbf{x}) \in \{i\} \times \mathcal{M}_i \quad \forall\, \mathbf{x} \in \mathcal{L}_i \tag{14}$$

for $i = 1, \ldots, t$. The condition in (13) implies that on write i, the t-write code encodes a message by only increasing the cell levels and the condition in (14) implies that the decoder maps every point in $\mathcal{L}_i$ to a message in $\mathcal{M}_i$. For a consistent encoder-decoder pair, we will further require

$$\psi\left(\phi(m, \mathbf{x})\right) = (i, m)\ \forall\ m \in \mathcal{M}_i, \mathbf{x} \in \mathcal{L}_{i-1} \tag{15}$$

for $i = 1, \ldots, t$. The condition in (15) requires that the cell levels after write i are decoded to the correct message without the knowledge of any of the previous $i-1$ writes. Thus, an n-cell t-write code $(\{\mathcal{L}_i\}, \phi, \psi)$ that satisfies conditions (13), (14) and (15) achieves worst-case sum-rate $R_t = \frac{1}{n}\sum_{i=1}^{t} \log M_i$.

Sections III and IV only consider the problem of partitioning the code $\mathcal{L}$ into t codebooks $\{\mathcal{L}_i\}$. Since the value of M_i cannot be determined without giving a consistent encoder-decoder scheme, the codebook cardinality, C_i, will be used as the figure of merit to define sum-rates in Sections III and IV.

D. Continuous Approximation

According to the *continuous approximation* principle for dense lattices [16], [17], the number of points in a codebook $\mathcal{L}$ formed using (3) can be approximated as

$$|\mathcal{L}| \approx \frac{\|\mathbb{A}\|}{\mathrm{Vol}\,(\Lambda)}, \tag{16}$$

where $|\mathcal{L}|$ denotes the cardinality of the discrete set $\mathcal{L}$ and $\|\mathbb{A}\|$ denotes the volume of the shaping region $\mathbb{A}$. This approximation becomes increasingly accurate as the density of the lattice increases. The use of the continuous approximation principle for WOM codes was introduced in [11].

WOM codebooks $\mathcal{L}_1, \ldots, \mathcal{L}_t$ may be constructed by partitioning $\mathbb{A}$ into t write-regions, $\mathbb{A}_1, \mathbb{A}_2, \ldots, \mathbb{A}_t$. To construct codebooks for cells that support discrete levels, let

$$\mathcal{L}_i = \mathcal{L} \cap \mathbb{A}_i. \tag{17}$$

Applying the continuous approximation to the individual write-regions, the codebook cardinality for the first write is approximated by

$$C_1 = |\mathcal{L}_1| \approx \frac{\|\mathbb{A}_1\|}{\mathrm{Vol}\,(\Lambda)} \triangleq V_1. \tag{18}$$

If the state of the memory after write $i-1$ is $\mathbf{s} \in \mathbb{A}_{i-1}$, then the set of possible levels that can be written on write i is

$$\mathbb{A}_i(\mathbf{s}) \triangleq \{\mathbf{x} \in \mathbb{A}_i : \mathbf{x} \succ \mathbf{s}\}. \tag{19}$$

Applying the continuous approximation for writes $2, 3, \ldots, t$, the codebook cardinality is approximated by

$$C_i \approx \frac{1}{\mathrm{Vol}(\Lambda)} \inf_{\mathbf{s} \in \mathbb{A}_{i-1}} \|\mathbb{A}_i(\mathbf{s})\| \triangleq V_i, \tag{20}$$

and the total codebook cardinality in t writes is approximated by

$$\Pi_t \approx \prod_{i=1}^{t} V_i \triangleq S_t. \tag{21}$$

In the following sections, the quantities V_i and S_t are also referred to as the codebook cardinality and the total codebook cardinality, respectively. Since both the lattice and the maximum cell values ℓ_i may be scaled arbitrarily, in Sections III and IV we assume that $\mathrm{Vol}\,(\Lambda) = 1$.

III. Optimal Codebook Cardinality

In this section and the next section, we do not design WOM codebooks directly. Rather, we select shaping regions $\mathbb{A}_1, \mathbb{A}_2, \ldots, \mathbb{A}_t$ and maximize the total codebook cardinality S_t. Under the continuous approximation principle, this corresponds to maximizing the upper bound on the worst-case sum rate (11).

Subsection III-A shows that for $t = 2$ writes, the optimal shaping region $\mathbb{A}_1$ has a hyperbolic shape. Subsection III-B extends this result to an arbitrary number of writes t. Subsection III-C considers the parameters defining the shaping regions which provide the optimal total codebook cardinality. Subsection III-D finds the value of S_t asymptotic in n, and gives conditions under which S_t approaches capacity.

A. Optimal 2-Write Regions

This subsection describes the case where n cells are used to store information twice before being erased, that is, $t = 2$. We will assume that the levels of the cell can only be increased from a previously written level.

Let $\mathbb{B}_1$ be the manifold that forms the boundary between $\mathbb{A}_1$ and $\mathbb{A}_2$. Alternatively, for a given boundary $\mathbb{B}_1$, $\mathbb{A}_1(\mathbb{B}_1)$ is defined as the closed region bounded by $\mathbb{B}_1$ and hyperplanes $x_j \geq 0$. The explicit dependence of the region $\mathbb{A}_1$ on the given boundary $\mathbb{B}_1$ will be suppressed when the meaning is clear from the context. With this notation, the codebook cardinality in the first write is

$$V_1(\mathbb{B}_1) \triangleq \|\mathbb{A}_1\|\,. \tag{22}$$

Suppose the state of the memory after the first write is $\mathbf{s}$. Then, the region that remains available for writing is

$$\mathbb{A}_2(\mathbf{s}) = \{\mathbf{x} \in \mathbb{A}_2 : \mathbf{x} \succ \mathbf{s}\}\,. \tag{23}$$

This is a rectangular region with volume $\prod_{j=1}^{n}(\ell_j - s_j)$, that is

$$\|\mathbb{A}_2(\mathbf{s})\| = \prod_{j=1}^{n}(\ell_j - s_j)\,. \tag{24}$$

In the worst case, the total codebook cardinality with two writes using n cells is

$$\begin{aligned} S_2\left(\mathbb{B}_1\right) &= V_1\left(\mathbb{B}_1\right) \cdot V_2\left(\mathbb{B}_1\right) && (25)\\ &= \|\mathbb{A}_1\| \cdot \inf_{\mathbf{s} \in \mathbb{A}_1} \|\mathbb{A}_2(\mathbf{s})\|\,. && (26) \end{aligned}$$

This quantity is a function of the boundary $\mathbb{B}_1$ alone. We refer to

$$\mathbb{B}_1^* = \arg\max_{\mathbb{B}_1 \subset \mathbb{A}} S_2\left(\mathbb{B}_1\right), \tag{27}$$

as the *optimal* boundary for the first write, assuming that the cells are to be used twice before each erase. The optimal boundary between the two write regions is known to be a rectangular hyperbola [14].

Definition III.1: Define $\mathbb{H}(u)$ as the region in $\mathbb{A}$ enclosed by an n-dimensional rectangular hyperbola with parameter u, i.e.,

$$\mathbb{H}(u) \triangleq \left\{\mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n}(\ell_j - x_j) \geq u \cdot \|\mathbb{A}\|\right\}, \tag{28}$$

with $0 \leq u \leq 1$.

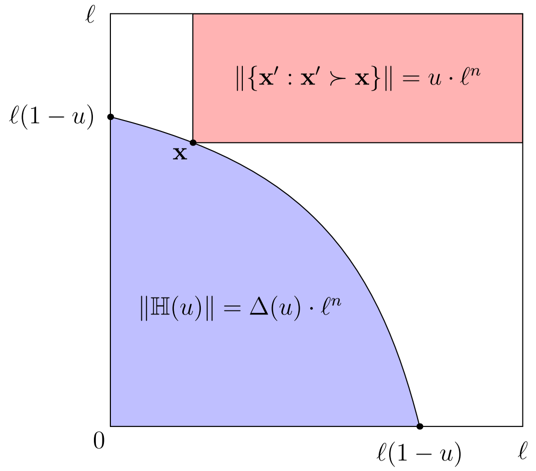

Fig. 1. A 2-dimensional rectangular hyperbola $\mathbb{H}(u)$ in region $\mathbb{A} = [0, \ell] \times [0, \ell]$. The region under the hyperbola and the region accessible from a given point $\mathbf{x}$ on the hyperbola are shaded in blue and red, respectively, and their volumes are equal to $\Delta(u) \cdot \ell^n$ and $u \cdot \ell^n$, respectively.

Definition III.2: The *normalized volume* of the region $\mathbb{H}(u)$ is denoted as

$$\Delta(u) \triangleq \|\mathbb{A}\|^{-1} \cdot \|\mathbb{H}(u)\| . \tag{29}$$

Geometrically, the parameter u, given in Definition III.1, characterizes the point where the hyperbola touches the axes, and is also equal to the normalized volume of the region accessible from any point on the boundary of the hyperbola. $\Delta(u)$ is equal to the normalized volume of the region under the hyperbola. As will be shown in Lemma III.5, $\Delta(u)$ can be expressed in closed form. Figure 1 shows a 2-dimensional rectangular hyperbola $\mathbb{H}(u)$.

The next lemma, first stated in [14], shows the optimality of the hyperbolic boundary.

Lemma III.3 (Optimal boundary for 2 writes): The optimal boundary for the first write-region for a 2-write WOM code is an n-dimensional rectangular hyperbola, i.e.,

$$\mathbb{B}_1^* = \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} (\ell_j - x_j) = u_2^* \cdot \|\mathbb{A}\| \right\} \tag{30}$$

where

$$u_2^* = \arg\max_{u \in [0,1]} u \cdot \Delta(u). \tag{31}$$

Proof: Let $\mathbb{B}_1^*$ denote the optimal boundary for the first write. Then, the codebook cardinality for the first write is

$$V_1 \left(\mathbb{B}_1^*\right) = \left\|\mathbb{A}_1 \left(\mathbb{B}_1^*\right)\right\| \tag{32}$$

and for the second write is

$$V_2 \left(\mathbb{B}_1^*\right) = \inf_{\mathbf{x} \in \mathbb{A}_1 \left(\mathbb{B}_1^*\right)} \left\|\mathbb{A}_2 \left(\mathbf{x}\right)\right\| = \inf_{\mathbf{x} \in \mathbb{B}_1^*} \prod_{j=1}^{n} (\ell_j - x_j) . \tag{33}$$

Define another boundary $\mathbb{B}_1' \subset \mathbb{A}$ such that

$$\mathbb{B}_1' = \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} (\ell_j - x_j) = V_2 \left(\mathbb{B}_1^*\right) \right\} . \tag{34}$$

Then, for any $\mathbf{x} \in \mathbb{B}_1'$, $\|\mathbb{A}_2(\mathbf{x})\| = V_2 \left(\mathbb{B}_1^*\right)$, and therefore, the boundaries $\mathbb{B}_1'$ and $\mathbb{B}_1^*$ achieve the same codebook cardinality on the second write, i.e.,

$$V_2 \left(\mathbb{B}_1'\right) = V_2 \left(\mathbb{B}_1^*\right) . \tag{35}$$

We claim that $\mathbb{A}_1 \left(\mathbb{B}_1'\right) \supseteq \mathbb{A}_1 \left(\mathbb{B}_1^*\right)$. Suppose the opposite is true, i.e., there exists some $\mathbf{x}' \in \mathbb{A}_1 \left(\mathbb{B}_1^*\right)$ such that $\mathbf{x}' \notin \mathbb{A}_1 \left(\mathbb{B}_1'\right)$ which implies that $\mathbf{x}' \notin \mathbb{H}\left(V_2 \left(\mathbb{B}_1^*\right) \cdot \|\mathbb{A}\|^{-1}\right)$. Then,

$$\prod_{j=1}^{n} \left(\ell_j - x_j'\right) < V_2 \left(\mathbb{B}_1^*\right) , \tag{36}$$

which contradicts the assumption that the codebook cardinality from any point in $\mathbb{A}_1 \left(\mathbb{B}_1^*\right)$ is

$$V_2 \left(\mathbb{B}_1^*\right) = \inf_{\mathbf{x} \in \mathbb{B}_1^*} \prod_{j=1}^{n} (\ell_j - x_j) . \tag{37}$$

Hence, $\mathbb{A}_1 \left(\mathbb{B}_1'\right) \supseteq \mathbb{A}_1 \left(\mathbb{B}_1^*\right)$ and $\left\|\mathbb{A}_1 \left(\mathbb{B}_1'\right)\right\| = V_1 \left(\mathbb{B}_1'\right) \geq V_1 \left(\mathbb{B}_1^*\right) = \left\|\mathbb{A}_1 \left(\mathbb{B}_1^*\right)\right\|$; that is, the codebook cardinality on the first write when using the boundary $\mathbb{B}_1'$ is at least as large as the codebook cardinality when using the optimal boundary.

Suppose $V_1 \left(\mathbb{B}_1'\right) > V_1 \left(\mathbb{B}_1^*\right)$. Thus, using $\mathbb{B}_1'$ as the boundary for the first write increases the codebook cardinality for the first write, while ensuring that the worst-case codebook cardinality on the second write is still the same. This contradicts the assumption that $\mathbb{B}_1^*$ is optimal, and therefore, $\left\|\mathbb{A}_1 \left(\mathbb{B}_1'\right)\right\| = V_1 \left(\mathbb{B}_1'\right) = V_1 \left(\mathbb{B}_1^*\right) = \left\|\mathbb{A}_1 \left(\mathbb{B}_1^*\right)\right\|$. We have shown that the first write-regions corresponding to boundaries $\mathbb{B}_1^*$ and $\mathbb{B}_1'$ have the same volume, and one is a subset of the other. It follows from Lemma A.1 in Appendix A that $\mathbb{A}_1(\mathbb{B}_1^*) = \mathbb{A}_1(\mathbb{B}_1')$ and $\mathbb{B}_1^* = \mathbb{B}_1'$. The optimal total codebook cardinality in two writes is given by

$$\begin{aligned} S_2(\mathbb{B}_1^*) &= V_1 \left(\mathbb{B}_1^*\right) \cdot V_2 \left(\mathbb{B}_1^*\right) && (38) \\ &= V_1 \left(\mathbb{B}_1'\right) \cdot V_2 \left(\mathbb{B}_1'\right) && (39) \\ &= \left(\Delta \left(u_2^*\right) \cdot \|\mathbb{A}\|\right) \cdot \left(u_2^* \cdot \|\mathbb{A}\|\right) . && (40) \end{aligned}$$

Since $\mathbb{B}_1^*$ is optimal, it follows that $u_2^* = \arg\max_{u \in [0,1]} u \cdot \Delta(u)$. This completes the proof. ■

In the next subsection, we extend Lemma III.3 to the case of $t > 2$ writes.

B. Optimal t-Write Regions

Consider n cells that can store levels $[0, \ell_1] \times [0, \ell_2] \times \ldots \times [0, \ell_n] = \mathbb{A}$. Let the subset $\mathbb{A}_i \subset \mathbb{A}$ denote the set of points that may be used on write i, for $i = 1, \ldots, t$. If $\mathbf{s}_i \in \mathbb{A}_i$ is written on write i, the set of points, $\mathbb{A}_{i+1}(\mathbf{s}_i)$, that can be written in write $i+1$ is

$$\mathbb{A}_{i+1}(\mathbf{s}_i) = \{\mathbf{x} \in \mathbb{A}_{i+1} : \mathbf{x} \succ \mathbf{s}_i\}. \tag{41}$$

Let $\mathbb{B}_i$ denote the boundary for write i, V_i denote the codebook cardinality for write i, and $S_t(\ell_1, \ell_2, \ldots, \ell_n)$ denote the total codebook cardinality.

The following theorem gives the sum-rate optimal write-regions for t writes on n cells.

Theorem III.4 (Sum-rate optimal t-writes): The boundary for the i^{th} write when storing information t times on n

cells such that the total codebook cardinality is maximized is given by

$$\mathbb{B}_i^* = \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} (\ell_j - x_j) = \prod_{m=t-i+1}^{t} u_m^* \cdot \|\mathbb{A}\| \right\} \quad (42)$$

for all $i = 1, \ldots, t-1$ where $u_1^* = 0$ and

$$u_k^* \triangleq \arg\max_{u \in [0,1]} \Delta(u) \cdot u^{k-1} \quad (43)$$

for all $k = 2, \ldots, t$. The codebook cardinality on write i, for $i = 1, \ldots, t$, is

$$V_i^* = \left(\prod_{m=t-i+2}^{t} u_m^* \right) \cdot \Delta\left(u_{t-i+1}^*\right) \cdot \|\mathbb{A}\|, \quad (44)$$

and the total codebook cardinality in t writes is

$$S_t^*(\ell_1, \ldots, \ell_n) = \left(\prod_{m=2}^{t} (u_m^*)^{m-1} \Delta(u_m^*) \right) \cdot (\|\mathbb{A}\|)^t. \quad (45)$$

Proof: The theorem is proved by induction. From Lemma III.3, the claim is true for $t = 2$. Suppose the claim is true for k writes. Consider the case for $k+1$ writes. Let the optimal boundary for the first write be $\mathbb{B}_{1,k+1}^*$. Here the second subscript denotes that the total number of writes is $k+1$. The codebook cardinality on the first write is

$$V_{1,k+1}(\mathbb{B}_{1,k+1}^*) = \left\| \mathbb{A}_{1,k+1}(\mathbb{B}_{1,k+1}^*) \right\|. \quad (46)$$

Suppose that a point $\mathbf{x} \in \mathbb{A}_{1,k+1}(\mathbb{B}_{1,k+1}^*)$ is written on the first write; then an upper bound on the total codebook cardinality in k subsequent writes is $S_k^*(\ell_1 - x_1, \ell_2 - x_2, \ldots, \ell_n - x_n)$. Let $\Omega_2^{k+1}(\mathbb{B}_{1,k+1}^*)$ denote the product of the codebook cardinalities for writes $i = 2, 3, \ldots, k+1$ after an arbitrary first write with $\mathbb{B}_{1,k+1}^*$ as the boundary of the first write-region. Thus, we can upper bound this product of codebook cardinalities as

$$\begin{aligned} &\Omega_2^{k+1}(\mathbb{B}_{1,k+1}^*) \\ &\le \inf_{\mathbf{x} \in \mathbb{A}_{1,k+1}(\mathbb{B}_{1,k+1}^*)} S_k^*(\ell_1 - x_1, \ldots, \ell_n - x_n) \quad (47) \\ &= c_k^* \left[\inf_{\mathbf{x} \in \mathbb{A}_{1,k+1}(\mathbb{B}_{1,k+1}^*)} \prod_{j=1}^{n} (\ell_j - x_j) \right]^k, \quad (48) \end{aligned}$$

where $c_k^* = \prod_{m=2}^{k} (u_m^*)^{m-1} \Delta(u_m^*)$ is a constant independent of $\mathbf{x}$ or $\mathbb{B}_{1,k+1}^*$. Here the equality labeled (48) follows from the induction hypothesis. Let $p \in [0,1]$ satisfy

$$p \triangleq \|\mathbb{A}\|^{-1} \cdot \inf_{\mathbf{x} \in \mathbb{A}_{1,k+1}(\mathbb{B}_{1,k+1}^*)} \prod_{j=1}^{n} (\ell_j - x_j). \quad (49)$$

Define $\mathbb{B}'$ to be an n-dimensional hyperbola such that

$$\prod_{j=1}^{n} (\ell_j - y_j) = p \cdot \|\mathbb{A}\| \quad (50)$$

for all $\mathbf{y} \in \mathbb{B}'$. Then, using the definition of p, one can easily show that

$$\mathbb{A}_{1,k+1}\left(\mathbb{B}_{1,k+1}^*\right) \subseteq \mathbb{A}_{1,k+1}(\mathbb{B}') = \mathbb{H}(p) \quad (51)$$

and therefore,

$$V_{1,k+1}\left(\mathbb{B}_{1,k+1}^*\right) \le V_{1,k+1}(\mathbb{B}') = \Delta(p) \cdot \|\mathbb{A}\|. \quad (52)$$

By the induction hypothesis, for a point $\mathbf{x}' \in \mathbb{B}'$, the optimal boundary for the i^{th} subsequent write in $[x_1', \ell_1] \times [x_2', \ell_2] \times \ldots \times [x_n', \ell_n]$, is

$$\begin{aligned} &\mathbf{x}' + \mathbb{B}_{i,k}^*(\ell_1 - x_1', \ldots, \ell_n - x_n') \quad (53) \\ &= \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} \left((\ell_j - x_j') - (x_j - x_j') \right) \right. \\ &\qquad \left. = \prod_{m=k-i+1}^{k} u_m^* \cdot \prod_{j=1}^{n} (\ell_j - x_j') \right\} \quad (54) \\ &= \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} (\ell_j - x_j) = \prod_{m=k-i+1}^{k} u_m^* \cdot p \cdot \|\mathbb{A}\| \right\} \quad (55) \end{aligned}$$

where $\mathbb{B}_{i,k}^*(a_1, \ldots, a_n)$ is the optimal boundary for cells that can store levels $[0, a_1] \times \ldots \times [0, a_n]$ and $\mathbf{x} + \mathbb{B}$ denotes the set $\{\mathbf{x} + \mathbf{y} : \mathbf{y} \in \mathbb{B}\}$. Note that the boundary in (53) is independent of the point $\mathbf{x}'$ written on the first write, for all $i = 1, \ldots, k-1$. This implies that

$$\begin{aligned} \Omega_2^{k+1}(\mathbb{B}') &= c_k^* \cdot p^k \|\mathbb{A}\|^k \quad (56) \\ &\ge \Omega_2^{k+1}(\mathbb{B}_{1,k+1}^*). \quad (57) \end{aligned}$$

Therefore the optimal boundary $\mathbb{B}_{1,k+1}^*$ is the same as the hyperbolic boundary $\mathbb{B}'$, and

$$V_{1,k+1}\left(\mathbb{B}_{1,k+1}^*\right) = \|\mathbb{A}_{1,k+1}(\mathbb{B}')\| = \Delta(p) \cdot \|\mathbb{A}\| \quad (58)$$

for some $p \in [0,1]$. Then the total codebook cardinality in $k+1$ writes is

$$\begin{aligned} &S_{k+1}(\mathbb{B}_{1,k+1}^*) \\ &= V_{1,k+1}(\mathbb{B}') \cdot \Omega_2^{k+1}(\mathbb{B}') \quad (59) \\ &= c_k^* \|\mathbb{A}\|^{k+1} \cdot \Delta(p) \cdot p^k \quad (60) \end{aligned}$$

By (43), the value of p that maximizes $p^k \cdot \Delta(p)$ is u_{k+1}^*, which implies that

$$\mathbb{B}_{i,k+1}^* = \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} (\ell_j - x_j) = \prod_{m=k-i+2}^{k+1} u_m^* \cdot \|\mathbb{A}\| \right\} \quad (61)$$

for $i = 1, \ldots, k$. Hence the claim is true for $k+1$ writes. This completes the induction step, proving the theorem. ∎

C. Computing the optimal hyperbola parameters, u_k^*

The expression defining u_k^* in (43) can be interpreted as follows. When the first write region is $\mathbb{H}(u_k^*)$, the total codebook cardinality for k writes, S_k, is given by the product of the codebook cardinality in the first write, V_1, and the total codebook cardinality in the last $k-1$ writes, S_{k-1}. V_1 is proportional to $\Delta(u_k^*)$, the normalized volume enclosed by the hyperbola. In the worst case, the volume accessible for the last $k-1$ writes is equal to the volume accessible from a point on the boundary of $\mathbb{H}(u_k^*)$ and, thus, proportional to u_k^*. Then, replacing t by $k-1$ and $\|\mathbb{A}\|$ by a constant times u_k^* in

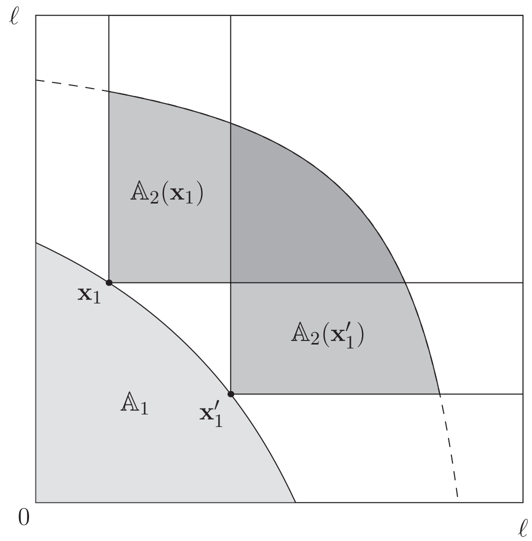

Fig. 2. Optimal boundaries for 3 writes over 2 cells. We can see that the boundaries for the second write are independent of the points written on the first write.

(45), the total codebook cardinality in the last $k-1$ writes is proportional to $(u_k^*)^{k-1}$. Therefore, the total codebook cardinality in k writes is proportional to $(u_k^*)^{k-1} \cdot \Delta(u_k^*)$ and the optimal hyperbola parameter is clearly given by the expression in (43). We now show how to compute this optimal parameter. We make use of the following lemma, which gives an expression for the normalized volume of the region $\mathbb{H}(u)$.

Lemma III.5: For $n \geq 2$,

$$\Delta(u) = 1 - u \sum_{i=0}^{n-1} \frac{1}{i!} \left[\ln\left(\frac{1}{u}\right) \right]^i. \tag{62}$$

Proof: See Appendix B. ■

First consider the case of $n = 2$ cells. The following proposition from [13] gives the value of the optimal hyperbola parameter, u_k^*.

Proposition III.6: For $n = 2$ cells and for any $k \geq 2$, let

$$f_k(u) \triangleq \Delta(u) \cdot u^{k-1} \tag{63}$$
$$= (1 - u + u \ln u) \cdot u^{k-1} \tag{64}$$

for all $u \in [0, 1]$ and let $\tau_k \triangleq -\frac{k-1}{k}$. Then, f_k has one local maximum in the interval $[0, 1]$ at

$$u_k^* = \arg\max_{u \in (0,1)} f_k(u) = \frac{\tau_k}{W_{-1}\left(\tau_k e^{\tau_k}\right)} \tag{65}$$

where W_{-1} is the real branch of the Lambert W function satisfying $W(x) < -1$ [18].

Proof: See Appendix C. ■

For $n = 2$, the proposition shows that the optimal hyperbola parameter, u_k^*, can be expressed in closed form using the Lambert W function. In contrast, for arbitrary n, the parameter u_k^* can only be found numerically. For the case of $t = 2$ writes on n cells, let $z_n^* \in [0, \infty)$ satisfy $u_2^* = e^{-z_n^*}$, where u_2^* is the optimal hyperbola parameter. We computed the value of z_n^* for different values of n and plot the results in Figure 3.

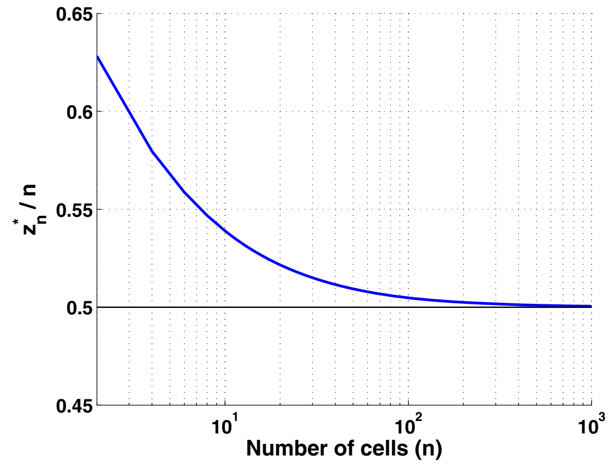

Fig. 3. Ratio of z_n^* and n, where $u_2^* = e^{-z_n^*}$ is the optimal hyperbola parameter for $t = 2$ writes on n cells. The ratio z_n^*/n converges monotonically to 0.5 as n increases.

D. Asymptotic Rates

In this subsection, we determine the rates that the continuous approximation achieves asymptotically as the number of cells, n, becomes large, that is, as $n \to \infty$, when cells can store fixed discrete levels in $\mathbb{Z}_q^n$. We consider $t = 2$ writes on n cells in the region $[0, q-1]^n$ in this subsection.

Let the first write-region be the hyperbolic region $\mathbb{H}(u)$. Then the total codebook cardinality in two writes is $S_2(u) = u \cdot \Delta(u) \cdot \|\mathbb{A}\|^2$. We consider $u = e^{-z_n}$ for some non-negative $z_n \in \mathbb{R}$ so that $0 \leq u \leq 1$. Then, by Lemma III.5,

$$S_2(e^{-z_n}) = e^{-z_n} \left(1 - e^{-z_n} \sum_{i=0}^{n-1} \frac{(z_n)^i}{i!} \right) \cdot (q-1)^{2n} \tag{66}$$

where $\|\mathbb{A}\| = \prod_{j=1}^n \ell_i = (q-1)^n$. Continuing, the sum-rate upper bound in (11) is

$$R_2 \leq \frac{1}{n} \log_2 \Pi_2 \approx \frac{1}{n} \log_2 S_2(e^{-z_n}) \tag{67}$$
$$= \log_2 (q-1)^2 - \log_2 e \cdot \frac{z_n}{n} + \log_2 \left[\mathrm{P}(n, z_n) \right]^{\frac{1}{n}} \tag{68}$$

where $\mathrm{P}(n, x)$ is the lower normalized incomplete Gamma function [19].

The value of z_n that maximizes the expression in (68), denoted by z_n^*, can be found numerically. Figure 3 shows that z_n^*/n approaches 0.5 monotonically as n increases. The following proposition confirms this numerical behavior.

Proposition III.7: Consider $t = 2$ writes on n cells with hyperbola parameter $u_2 = e^{-\alpha n}$, where α depends on n. For increasing n, the optimal value of α, in the sense of maximizing the total codebook cardinality, $S_2(e^{-\alpha n})$, approaches 0.5, that is

$$\lim_{n \to \infty} \arg\max_{\alpha} \frac{1}{n} \log_2 S_2\left(e^{-\alpha n}\right) = 0.5, \tag{69}$$

and

$$\lim_{n \to \infty} \frac{1}{n} \log_2 S_2\left(e^{-0.5n}\right) = \log_2\left(q^2 - 2q + 1\right) - 1. \tag{70}$$

Proof: See Appendix D. ■

For comparison, the capacity for t writes on cells with q discrete levels was shown in [12] to be $C_{q,t} \triangleq \log_2 \binom{q+t-1}{q-1}$. For $t = 2$ writes,

$$R_2 \le \lim_{n\to\infty} \frac{1}{n} \log_2 \Pi_2 \tag{71}$$
$$\approx \lim_{n\to\infty} \frac{1}{n} \log_2 S_2 \left(e^{-z_n^*}\right) \tag{72}$$
$$< C_{q,2} = \log_2 \left(q^2 + q\right) - 1. \tag{73}$$

That is, for any fixed q, even if the optimal value z_n^* is selected, the sum-rate under the assumptions of the continuous approximation is strictly less than the capacity. But, as q becomes large, their ratio goes to 1.

IV. Fixed-Rate Optimal t-Writes

In practice, it might be preferable to constrain successive writes to have the same rate. Continuing the developments of the previous section, this section considers shaping regions $\mathbb{A}_1, \ldots, \mathbb{A}_t$ such that the codebook cardinality is constant, that is $V_1 = V_2 = \cdots = V_t$. The following theorem, the proof of which is analogous to the proof of Theorem III.4, gives the optimal write-regions under this constraint.

Theorem IV.1 (Fixed-rate optimal t-writes): The unique optimal boundary for write i when storing information t times on n cells such that the total codebook cardinality is maximized and the codebook cardinality on each write is the same is given by

$$\mathbb{B}_i^* = \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} (\ell_j - x_j) = \prod_{m=t-i+1}^{t} v_m^* \cdot \|\mathbb{A}\| \right\} \tag{74}$$

for all $i = 1, \ldots, t-1$ where $v_1^* = 0$, and for $k \ge 2$, v_k^* satisfies

$$\Delta\left(v_k^*\right) = v_k^* \cdot \Delta\left(v_{k-1}^*\right). \tag{75}$$

The codebook cardinality on write i, $i = 1, \ldots, t$, is

$$V_{\text{fix}}^* = \Delta\left(v_t^*\right) \cdot \|\mathbb{A}\|, \tag{76}$$

and the total codebook cardinality in t writes is

$$S_{\text{fix}}^*(\ell_1, \ldots, \ell_n) = \left(\Delta\left(v_t^*\right) \cdot \|\mathbb{A}\|\right)^t. \tag{77}$$

Proof: We prove the theorem by induction. First, suppose $t = 2$. Let the boundary be

$$\mathbb{B}' = \left\{ \mathbf{x} \in \mathbb{A} \ : \ \prod_{j=1}^{n} (\ell_j - x_j) = v_2^* \cdot \|\mathbb{A}\| \right\} \tag{78}$$

so that the first write-region $\mathbb{A}_1(\mathbb{B}')$ is $\mathbb{H}(v_2^*)$. The second write-region $\mathbb{A}_2$ is $\mathbb{A} \setminus \mathbb{A}_1(\mathbb{B}')$ and $\mathbb{A}_2(\mathbf{x}_1) = \{\mathbf{x} \in \mathbb{A}_2 : \mathbf{x} \succ \mathbf{x}_1\}$ for all $\mathbf{x}_1 \in \mathbb{A}_1(\mathbb{B}')$. Then, the codebook cardinality in each of the two writes, under the constraint that the rates are equal, is

$$V_{\text{fix}}(\mathbb{B}') = \min\left\{ \|\mathbb{A}_1(\mathbb{B}')\| \ , \ \min_{\mathbf{x}_1 \in \mathbb{A}_1} \|\mathbb{A}_2(\mathbf{x}_1)\| \right\} \tag{79}$$
$$= \min\left\{ \|\mathbb{A}_1(\mathbb{B}')\| \ , \ \min_{\mathbf{x} \in \mathbb{B}'} \|\mathbb{A}_2(\mathbf{x})\| \right\} \tag{80}$$
$$= \min\left\{ \Delta(v_2^*) \cdot \|\mathbb{A}\| \ , \ v_2^* \cdot \|\mathbb{A}\| \right\} \tag{81}$$
$$= \Delta(v_2^*) \cdot \|\mathbb{A}\| \ = \ v_2^* \cdot \|\mathbb{A}\| \tag{82}$$

where (80) follows from the fact that the minimum is achieved when $\mathbf{x}_1$ lies on the boundary $\mathbb{B}'$ and (82) follows from (75). Now, suppose the optimal write-regions are $\mathbb{A}_1^*$ and $\mathbb{A}_2^*$ with maximum codebook cardinality V_{fix}^*. Then $V_{\text{fix}}(\mathbb{B}') \le V_{\text{fix}}^*$. We will prove that $\mathbb{A}_1^* \subseteq \mathbb{A}_1(\mathbb{B}')$. Suppose the opposite is true; that is, suppose there exists $\mathbf{x}' \in \mathbb{A}_1^*$ such that $\prod_j (\ell_j - x_j') < v_2^* \cdot \|\mathbb{A}\|$. Since the codebook cardinality V_{fix}^* is upper bounded by the volume of the second region,

$$V_{\text{fix}}^* \le \|\mathbb{A}_2^*(\mathbf{x}')\| = \prod_j (\ell_j - x_j') < v_2^* \cdot \|\mathbb{A}\| = V_{\text{fix}}(\mathbb{B}'), \tag{83}$$

which contradicts the optimality of $\mathbb{A}_1^*$. Thus, the optimal first write-region $\mathbb{A}_1^*$ must be contained in $\mathbb{A}_1(\mathbb{B}')$. On the other hand, by the optimality of $\mathbb{A}_1^*$ and $\mathbb{A}_2^*$,

$$V_{\text{fix}}(\mathbb{B}') \le V_{\text{fix}}^* \le \|\mathbb{A}_1^*\| \tag{84}$$
$$\le \|\mathbb{A}_1(\mathbb{B}')\| = V_{\text{fix}}(\mathbb{B}'), \tag{85}$$

which implies that

$$V_{\text{fix}}^* = \|\mathbb{A}_1^*\| = V_{\text{fix}}(\mathbb{B}') = \Delta(v_2^*) \cdot \mathbb{A}. \tag{86}$$

To sum up, $\mathbb{A}_1^* \subseteq \mathbb{A}_1(\mathbb{B}')$ and $\|\mathbb{A}_1^*\| = \|\mathbb{A}_1(\mathbb{B}')\|$. It now follows from Lemma A.1 that $\mathbb{A}_1^* = \mathbb{A}_1(\mathbb{B}')$. It can similarly be shown that $\mathbb{A}_2^* = \mathbb{A} \setminus \mathbb{A}_1(\mathbb{B}')$. This proves that the theorem holds for $t = 2$.

Now suppose the theorem holds when the number of writes is k. Consider the case for $k+1$ writes. Suppose v_{k+1}^* satisfies $\Delta(v_{k+1}^*) = v_{k+1}^* \cdot \Delta(v_k^*)$. Define the boundary for the first write-region as

$$\mathbb{B}'_{1,k+1} = \left\{ \mathbf{x} \in \mathbb{A} \ : \ \prod_{j=1}^{n} (\ell_j - x_j) = v_{k+1}^* \cdot \|\mathbb{A}\| \right\} \tag{87}$$

so that the first write-region is $\mathbb{A}_{1,k+1} = \mathbb{H}\left(v_{k+1}^*\right)$. Let $V_{\text{fix},k}^*\left(\ell_1 - x_1, \ldots, \ell_n - x_n\right)$ denote the optimal codebook cardinality for the last k writes in the region $[x_1, \ell_1] \times \ldots \times [x_n, \ell_n]$. Then the codebook cardinality on each subsequent write is given by

$$V_{\text{fix},k+1} \le \min\left\{ \|\mathbb{A}_{1,k+1}\| \ , \ \min_{\mathbf{x} \in \mathbb{A}_{1,k+1}} V_{\text{fix},k}^*\left(\ell_1 - x_1, \ldots, \ell_n - x_n\right) \right\} \tag{88}$$
$$= \min\left\{ \left\|\mathbb{H}(v_{k+1}^*)\right\| \ , \ \min_{\mathbf{x} \in \mathbb{A}_{1,k+1}} \Delta(v_k^*) \cdot \prod_{j=1}^{n} (\ell_j - x_j) \right\} \tag{89}$$
$$= \min\left\{ \left\|\mathbb{H}(v_{k+1}^*)\right\| \ , \ \min_{\mathbf{x} \in \mathbb{B}'_{1,k+1}} \Delta(v_k^*) \cdot \prod_{j=1}^{n} (\ell_j - x_j) \right\} \tag{90}$$
$$= \min\left\{ \Delta(v_{k+1}^*) \cdot \|\mathbb{A}\| \ , \ \Delta(v_k^*) \cdot v_{k+1}^* \cdot \|\mathbb{A}\| \right\} \tag{91}$$
$$= \Delta(v_{k+1}^*) \cdot \|\mathbb{A}\| = \Delta(v_k^*) \cdot v_{k+1}^* \cdot \|\mathbb{A}\|. \tag{92}$$

Equality (89) follows from the induction hypothesis, (90) follows from the fact that the minimum occurs when $\mathbf{x}$ lies on $\mathbb{B}'_{1,k+1}$, and (92) follows from (75). In a manner similar to the proof of Theorem III.4, it can be shown that the optimal boundary for the i^{th} subsequent write after storing a point $\mathbf{x}' \in \mathbb{B}'_{1,k+1}$ on the first write is given by

$$\begin{aligned}
&\mathbb{B}^*_{i+1,k+1} \\
&= \mathbf{x}' + \mathbb{B}^*_{i,k}\left(\ell_1 - x'_1, \ldots, \ell_n - x'_n\right) && (93)\\
&= \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} (\ell_j - x_j) = \prod_{m=k-i+1}^{k} v^*_m \cdot v^*_{k+1} \cdot \|\mathbb{A}\| \right\} && (94)\\
&= \left\{ \mathbf{x} \in \mathbb{A} : \prod_{j=1}^{n} (\ell_j - x_j) = \prod_{m=k-i+1}^{k+1} v^*_m \cdot \|\mathbb{A}\| \right\}. && (95)
\end{aligned}$$

From (95), $\mathbb{B}^*_{i+1,k+1}$ is independent of $\mathbf{x}'$, the point stored on the first write, and the inequality in (88) is in fact an equality. Thus, the claim in the theorem is true for $k+1$ writes. This proves the theorem by induction. ■

A. Computing the optimal hyperbola parameters, v^*_t

As was the case for sum-rate optimal write-regions in Section III-C, the hyperbola parameters for optimal fixed-rate write-regions can be computed easily for the case of $n = 2$ cells. This is done in the following proposition which is similar to Proposition III.6.

Proposition IV.2: For $n = 2$ cells, the optimal hyperbola parameters, v^*_k, are given by the recurrence relation

$$v^*_k = \frac{-1}{W_{-1}\left(-\exp\left(-1 - \prod_{m=1}^{k-1} v^*_m\right)\right)} \quad (96)$$

Proof: See Appendix E. ■

Again, as was the case for sum-rate optimal write-regions, the hyperbola parameters for optimal fixed-rate write-regions for $n > 2$ cells can only be computed numerically. In the following proposition, we derive bounds on v^*_t.

Proposition IV.3: For $t \geq 2$

$$\exp\left(-(n!)^{\frac{1}{n}}\right) < v^*_t < 1. \quad (97)$$

Proof: See [14, Section IV.A]. ■

V. Separability of Shaping Gain

This section shows that the contributions of the lattice Λ and the shaping regions $\mathbb{A}_i$ to the total codebook cardinality may be separated under the continuous approximation. We allow the lattice Λ to be scaled arbitrarily, while fixing $\ell = 1$. Define $A_1 = \|\mathbb{A}_1\|$ and for $i = 2, \ldots, t$,

$$A_i = \inf_{\mathbf{x} \in \mathbb{A}_{i-1}} \|\mathbb{A}_i(\mathbf{x})\|. \quad (98)$$

Here, $\ell = 1$ implies that $A_i \leq 1$. Using this A_i along with equations (8) and (20), the sum-rate upper bound may be written as

$$\frac{1}{n}\log_2 S_n \approx \frac{t}{2}\log_2\left(\frac{1}{\mathrm{Vol}\,(\Lambda)^{2/\mathrm{n}}}\right) + \log_2\left(A_{\text{shape}}\right), \quad (99)$$

where

$$A_{\text{shape}} = \prod_{i=1}^{t} (A_i)^{1/n}. \quad (100)$$

This shows that the upper bound on the sum-rate can be separated into two parts — one part that depends only on the lattice Λ and one part that depends only on the shaping regions $\mathbb{A}_i$.

The term $\mathrm{Vol}\,(\Lambda)^{2/\mathrm{n}}$ is sometimes called the normalized volume of the lattice, and appears frequently in the study of lattices for communications [17]. The design of the lattice has a strong influence on the error-correction capability of the code. The continuous approximation is relevant when $\mathrm{Vol}\,(\Lambda) \ll 1$; the contribution of the lattice dominates (99).

A "shaping gain" γ_{shape} expresses the benefit of using non-cubic shaping regions $\mathbb{A}_1, \ldots, \mathbb{A}_t$. A naive, cubic scheme is a partition of $\mathbb{A}$ based on cubes with edge length i/t for $i = 1, \ldots, t$, which easily leads to $A_i = \frac{1}{t^n}$. Accordingly, in analogy to (100), define $A_{\text{cube}} = (\frac{1}{t^n})^t$. Then, the shaping gain of any scheme with A_{shape} is defined as

$$\gamma_{\text{shape}} = \log_2 \frac{A_{\text{shape}}}{A_{\text{cube}}} \text{ bits per cell.} \quad (101)$$

As shown in Section III, the optimal shape for $t = 2$ is a hyperbola with parameter $\frac{z^*_n}{n} = 0.5$, in the limit of $n \to \infty$. The optimal sum-rate upper bound in (70) separates into two parts, $\log_2(q^2 - 2q - 1)$ and -1. The former part is the contribution due to the lattice only, and the latter part is the contribution due to the shape only; refer to (D.6). That is, Proposition III.7 also shows that, for sum-rate optimal shaping, $\lim_{n\to\infty} \frac{1}{n}\log_2 A_{\text{shape}} = -1$. This corresponds to $A_{\text{shape}} = \frac{1}{2}$, giving shaping gain

$$\gamma_{\text{shape}} = \log_2 \frac{1}{2} - \log_2 \frac{1}{4} = 1 \text{ bit per cell.} \quad (102)$$

That is, the sum-rate optimal shaping region gains 1 bit/cell over the cubic shaping.

Fixed rates were studied in Section IV. In [14] it was hypothesized that for $t = 2$ writes, $\frac{z^*}{n} = \frac{1}{e}$ is optimal for fixed rates as $n \to \infty$. This corresponds to $A_1 = A_2 = e^{-\frac{n}{e}}$. In this case,

$$\gamma_{\text{shape}} = \log_2 e^{-\frac{2}{e}} - \log_2 \frac{1}{4} \approx 0.9385 \text{ bit/cell.} \quad (103)$$

Thus, the hyperbolic fixed-rate shaping achieves nearly the same shaping gain as sum-rate optimized shaping.

VI. Message Assignment for Codes with Discrete Support

In this section we consider cells that support discrete levels $\mathbf{x} \in \mathcal{L}$ and discuss the construction of an encoder-decoder pair (ϕ, ψ) for a given codebook $\{\mathcal{L}_i\}_{i=1}^t$ as described in Section II-B. We associate with the encoder-decoder pair a function

$$\Phi : \mathcal{L} \mapsto \cup_i \mathcal{M}_i,$$

such that $\Phi(\mathbf{x})$ denotes the message associated with any point $\mathbf{x} \in \mathcal{L}$. Then, the encoder and decoder are defined in Algorithm 1 using look-ups of the message assignment Φ. We will refer to the function Φ as the *message assignment*. The encoder-decoder pair (ϕ, ψ) that satisfy the conditions (13), (14) and (15) can be defined in this manner if and only if Step 7 in Algorithm 1 can be performed successfully, that is,

Algorithm 1 Encoder-decoder pair (ϕ, ψ) based on matrix Φ

// Inputs are message to be written and current cell levels
// Output is next cell levels
function $\phi\ (m, \mathbf{x})$
Require: $(m, \mathbf{x}) \in \cup_{j=1}^{t} (\mathcal{M}_j \times \mathcal{L}_{j-1})$
    // Determine the number of writes already completed
    $i \leftarrow \sum_{j=1}^{t} j \cdot \mathbf{1}\{\mathbf{x} \in \mathcal{L}_{j-1}\}$
    // Find an accessible point that encodes message m
    Find $\mathbf{x}' \in \mathcal{L}_i(\mathbf{x}) : \Phi(\mathbf{x}') = m$
    return $\mathbf{x}'$
end function

// Input is current cell levels
// Outputs are number of writes and message
function $\psi\ (\mathbf{x})$
    // Determine the number of writes done
    $i \leftarrow \sum_{j=1}^{t} j \cdot \mathbf{1}_{\{\mathbf{x} \in \mathcal{L}_j\}}$
    // Determine the message that point $\mathbf{x}$ encodes
    $m \leftarrow \Phi(\mathbf{x})$
    return (i, m)
end function

Φ satisfies the following property for all $j = 1, \ldots, t$ and for all $\mathbf{x} \in \mathcal{L}_{j-1}$:

$$\exists\, \mathbf{x}' \in \mathcal{L}_j(\mathbf{x}) : \Phi(\mathbf{x}') = m \quad \forall\, m \in \mathcal{M}_j. \tag{104}$$

The condition in (104) implies that from any point, $\mathbf{x}$, there exists at least one accessible point in the next write-region, $\mathbf{x}'$, that encodes the message m, for all messages m.

In the following subsections, we discuss the construction of a message assignment function Φ for the case of $n = 2$ cells, and then for an arbitrary number of cells. Note that [10] uses a lattice to define the message assignment function where codebooks are defined using tilings of the lattice. Here, we consider the message assignment problem for an arbitrary codebook $\{\mathcal{L}_i\}$. For numerical results in this section, the codebooks were constructed using (17) with the proposed hyperbolic shaping regions as $\mathbb{A}_i$.

A. *Message assignment for* 2 *cells*

We first consider the case of $n = 2$ cells with cell levels $(x_1, x_2) \in \mathcal{L} = \mathbb{Z}_q^2$. In this case, we propose an algorithm to find a function Φ that satisfies (104) for a given codebook $\{\mathcal{L}_i\}_{i=1}^{t}$ such that the number of messages on write i, M_i, is equal to the codebook cardinality, C_i. The implementation details are given in Algorithm 2. We now describe how this iterative algorithm works and prove its correctness.

For each write i, on iteration 0, we start at the point in the codebook $\mathcal{L}_{i-1}$ that can access the least number of points,

$$(\hat{x}_1, \hat{x}_2) = \underset{(x_1,x_2) \in \mathcal{L}_{i-1}}{\arg\min} \ |\mathcal{L}_i(x_1, x_2)|\,.$$

Let us call this point the *pivot point* at iteration 0. Note that $|\mathcal{L}_i(\hat{x}_1, \hat{x}_2)|$ is equal to C_i by the definition of codebook cardinality in (7). We assign messages $\mathcal{M}_i = \{1, 2, \ldots, C_i\}$ to the points in $\mathcal{L}_i$ accessible from this pivot point. On the first iteration, the algorithm considers as the pivot point a point $(x_1'', x_2'') \in \mathcal{L}_{i-1}$ such that $x_1'' = \hat{x}_1 - 1$ and x_2'' is the largest

Algorithm 2 Algorithm to construct Φ given $\{\mathcal{L}_i\}_{i=1}^{t}$

procedure DEFINEPHI $(\{\mathcal{L}_i\}_{i=1}^{t})$
    for all $i \in \{1, \cdots, t\}$ **do**
        // Initialize Φ to Unassigned (denoted by ϵ)
        for all $(x_1, x_2) \in \mathcal{L}_i$ **do**
            $\Phi(x_1, x_2) \leftarrow \epsilon$
        end for
        // Start with point that achieves the min. volume
        $(\hat{x}_1, \hat{x}_2) \leftarrow \underset{(x_1,x_2) \in \mathcal{L}_{i-1}}{\arg\min} \ |\mathcal{L}_i(x_1, x_2)|$
        $\mathcal{M}_i \leftarrow \{1, 2, \ldots, |\mathcal{L}_i(\hat{x}_1, \hat{x}_2)|\}$
        // Assign messages to points $(\hat{x}_1, \hat{x}_2)$ can access
        ASSIGNPHI $(\mathcal{L}_i(\hat{x}_1, \hat{x}_2), \mathcal{M}_i)$
        // Assign messages from other points in $\mathcal{L}_{i-1}$
        // Iterations with decreasing x_1
        $(x_1', x_2') \leftarrow (\hat{x}_1, \hat{x}_2)$
        while $x_1' > 1 \wedge x_2' < q$ **do**
            // Find new pivot point and lost messages
            $(x_1'', x_2'') \leftarrow (x_1' - 1, x_2) \in \mathcal{L}_{i-1} : x_2$ is largest
            $\mathcal{M}_{\text{lost}} \leftarrow \mathcal{M}_i \setminus \Phi(\mathcal{L}_i(x_1'', x_2''))$
            ASSIGNPHI $(\mathcal{L}_i(x_1'', x_2''), \mathcal{M}_{\text{lost}})$
            $(x_1', x_2') \leftarrow (x_1'', x_2'')$
        end while
        // Iterations with increasing x_1
        $(x_1', x_2') \leftarrow (\hat{x}_1, \hat{x}_2)$
        while $x_1' < q \wedge x_2' \geq 1$ **do**
            // Find new pivot point and lost messages
            $(x_1'', x_2'') \leftarrow (x_1' + 1, x_2) \in \mathcal{L}_{i-1} : x_2$ is largest
            $\mathcal{M}_{\text{lost}} \leftarrow \mathcal{M}_i \setminus \Phi(\mathcal{L}_i(x_1'', x_2''))$
            ASSIGNPHI $(\mathcal{L}_i(x_1'', x_2''), \mathcal{M}_{\text{lost}})$
            $(x', y') \leftarrow (x'', y'')$
        end while
        // Repeat for all writes
    end for
end procedure

possible. On subsequent iterations, the algorithm will consider pivot points with decreasing levels on the first cell. The pivot point in the previous iteration is denoted as (x_1', x_2').

At every iteration, the algorithm finds messages previously assigned to the points in $\mathcal{L}_i(x_1', x_2') \setminus \mathcal{L}_i(x_1'', x_2'') \triangleq \mathcal{L}_{\text{lost}}(x_1'', x_2'')$. Let $\mathcal{M}_{\text{lost}}$ be the set of these messages that were encodable in the previous iteration but are not encodable in the current iteration. The algorithm then assigns $\mathcal{M}_{\text{lost}}$ to the points $\mathcal{L}_i(x'', y'')$ that have not yet been assigned any messages, a set we denote as $\mathcal{L}_{\text{spare}}(x'', y'')$. The algorithm repeats these steps, with a different pivot point at every iteration. We show in Appendix F that $|\mathcal{L}_{\text{spare}}(x'', y'')| \geq |\mathcal{M}_{\text{lost}}|$ at every iteration, so that a message assignment function that satisfies the condition in (104) can be defined for any given codebook.

A 4-write code for $n = 2$ cells with $q = 8$ levels each is shown in Figure 4. For this code, the codebook $\{\mathcal{L}_i\}_{i=1}^{4}$ is constructed using Equation (17) where the shaping regions $\{\mathbb{A}_i\}$ are as defined in Theorem III.4 with $\ell = q - 1 = 7$ and the message assignment function Φ is defined using Algorithm 2. The code allows $8, 8, 9$ and 8 messages on the sequence of

Algorithm 3 Procedure to assign messages $\mathcal{M}'$ to points in set $\mathcal{L}'$

Require: $|\mathcal{M}'| \leq |\{(x_1, x_2) \in \mathcal{L}' : \Phi(x_1, x_2) = \epsilon\}|$

procedure ASSIGNPHI $(\mathcal{L}', \mathcal{M}')$
  $\mathcal{L}_{\text{spare}} \leftarrow \{(x_1, x_2) \in \mathcal{L}' : \Phi(x_1, x_2) = \epsilon\}$
  for all $m \in \mathcal{M}'$ **do**
    $(x_{\text{spare}}, y_{\text{spare}}) \leftarrow (x, y) : (x, y) \in \mathcal{L}_{\text{spare}}$
    $\Phi(x_{\text{spare}}, y_{\text{spare}}) \leftarrow m$
    $\mathcal{L}_{\text{spare}} \leftarrow \mathcal{L}_{\text{spare}} \setminus \{(x_{\text{spare}}, y_{\text{spare}})\}$
  end for
end procedure

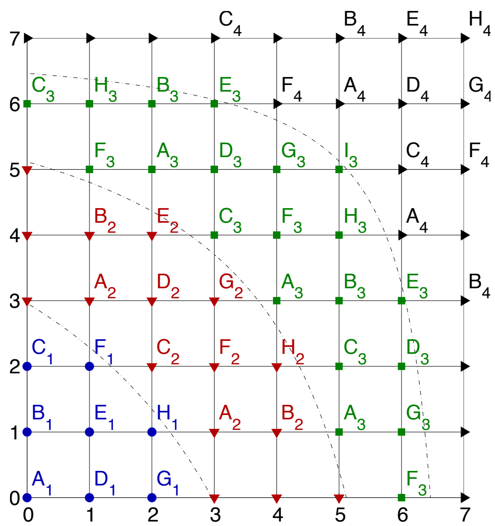

Fig. 4. 4-write code for 2 cells with 8 levels that achieves rate 6.085 bits/cell/erase. Points in the i^{th} codebook are assigned messages $\{\mathsf{A}_i, \mathsf{B}_i, \cdots\}$ according to message assignment function defined in Algorithm 2 such that after any $i-1$ writes, M_i messages may be stored on the next write.

four writes to give a worst-case sum-rate of

$$\begin{aligned} R_4 &= \frac{1}{2}\log_2(8 \cdot 8 \cdot 9 \cdot 8) \\ &= 6.085 \text{ bits per cell per erase.} \end{aligned} \tag{105}$$

In Table I, we list the worst-case sum-rate R_t achieved by the proposed codes over $n = 2$ cells that support levels $\{0, 1, \ldots q-1\}$, for various values of t and q. With no coding, these cells achieve a sum-rate equal to $\log_2 q$ bits/cell/erase. As is clear from Table I, the proposed WOM codes achieve a significant gain in sum-rate even when coding is done over only 2 cells.

TABLE I
WORST-CASE SUM-RATES R_t IN BITS PER CELL PER ERASE ACHIEVED BY t-WRITE CODES ON 2 CELLS WITH q LEVELS.

q	4	8	12	16	32
$\downarrow t \backslash \log_2 q \rightarrow$	2	3	3.59	4	5
2	2.70	4.55	5.63	6.44	8.40
3	2.95	5.48	7.11	8.25	11.13
4	2.59	**6.09**	8.17	9.71	13.46
5	2.09	6.55	9.07	10.90	15.40
6	1.79	6.61	9.63	11.80	17.21
7	–	6.70	10.10	12.54	18.72
8	–	6.42	10.26	13.15	20.22
9	–	6.38	10.55	13.73	21.43
10	–	5.88	10.78	14.19	22.57

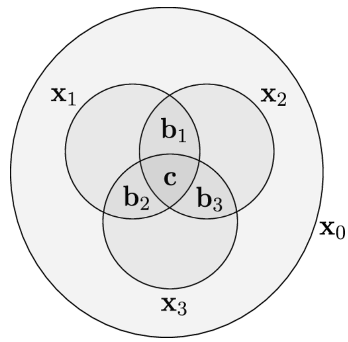

Fig. 5. Diagram for Example VI.1 showing points of second write-region accessible from points of first write-region.

B. Message assignment for arbitrary number of cells

In the case of $n = 2$ cells above, we could always define an encoder-decoder pair that, on any write, could store a set of messages with cardinality equal to the codebook cardinality, that is, $M_i = C_i$ for all i. However, when $n > 2$, a consistent encoder-decoder with this property may not exist for some codebooks over n cells, as we show in the following example.

Example VI.1: Consider a code over 3 cells that support 2 levels, $\{0, 1\}$, with codebooks

$$\mathcal{L}_1 = \{(0,0,0),(1,0,0),(0,1,0),(0,0,1)\} = \{\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3\},$$
$$\mathcal{L}_2 = \{(1,1,0),(1,0,1),(0,1,1),(1,1,1)\} = \{\mathbf{b}_1, \mathbf{b}_2, \mathbf{b}_3, \mathbf{c}\}.$$

Then, there are 3 points in the second write-region that are accessible from points $\mathbf{x}_i$ for $i = 1, 2, 3$. For example, points $\mathbf{b}_1, \mathbf{b}_2$ and $\mathbf{c}$ are accessible from $\mathbf{x}_1$, but $\mathbf{b}_3$ is not. This is depicted schematically in Figure 5 where points $\mathcal{L}_1$ are represented as circular regions and accessible points in $\mathcal{L}_2$ for each of the points in $\mathcal{L}_1$ are represented as subregions of the respective regions. Clearly $C_2 = 3$. However, there does not exist any assignment of 3 messages to points in $\mathcal{L}_2$ such that all points in $\mathcal{L}_1$ satisfy the condition in (104) required for a valid Φ function. Therefore, $M_2 < C_2$ for the given codebooks.

It is easy to determine that the maximum number of messages that can be stored in the second write for the case discussed in Example VI.1 is 2; that is, $M_2 = 2$. However, in general, given codebooks $\mathcal{L}_{i-1}$ and $\mathcal{L}_i$ and a positive integer $M \geq 2$, the problem to determine whether there exists an encoder-decoder pair that guarantees M messages on write i is an instance of a combinatorial problem, referred to as *Set M-Coloring* in [20], and defined as follows.

Definition VI.1 (Set M-Coloring): Given a collection of finite subsets $\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_m$ of a set $\mathcal{U}$, does there exist a coloring

$$I : \bigcup_{k=1}^{m} \mathcal{S}_k \mapsto \{1, 2, \ldots, M\}$$

such that, for each $k = 1, \ldots, m$, and for each color $j = 1, \ldots, M$, the set $\mathcal{S}_k$ has at least one element of color j?

One can see that the existence of a message assignment with M messages can be interpreted as an instance of the Set M-Coloring problem. Specifically suppose $\mathcal{L}_{i-1} = \{\mathbf{x}_1, \ldots, \mathbf{x}_m\}$. Then sets $\mathcal{U} = \mathcal{L}_i$ and $\mathcal{S}_k = \mathcal{L}_i(\mathbf{x}_k)$ for $k = 1, \ldots, m$. It was shown in [20, Theorem 5.5] that the Set M-Coloring problem is NP-complete [21] for any fixed $M \geq 2$. It is important to note that this does not prove hardness for the message assignment problem. Indeed, Algorithm 2 is a polynomial-time algorithm that solves the message assignment problem for $n = 2$.

For a given codebook, the problem of finding a message assignment has been interpreted as finding the disjoint coverings for a corresponding bipartite graph, as shown in [22]. This bipartite covering problem is an optimization version of the Set-M Coloring problem and therefore is NP-hard. Solving this covering problem sub-optimally [23, Algorithm 1], we have found message assignments for some codebooks that achieve higher sum-rates with 3 cells than achieved with 2 cells in Table I. For e.g., for $t = 2$ writes on $n = 3$ cells where $q = 8$, we could construct a message assignment function with $M_1 = 136$ and $M_2 = 101$, giving a sum-rate 4.582 bits/cell/erase; the case $n = 2$ achieves 4.554 bits/cell/erase for the same t and q.

VII. Conclusions

This paper described the design of lattice-based WOM codebooks using minimal assumptions of the continuous approximation and a worst-case rate. The optimal shaping regions were shown to bounded by rectangular hyperbolas, and we defined a total codebook cardinality Π_t which gave an upper bound on the sum-rate. For large q, corresponding to the condition of the continuous approximation, this upper bound approaches the capacity, for the case of $t = 2$ writes.

Note that the analysis only considered maximizing the size of codebooks. Indeed, an example showed that a consistent encoder-decoder pair that achieves rates equal to the codebook cardinality may not exist for all codebooks when $n \geq 3$. That is, it has not been shown that hyperbolic shaping regions have a consistent encoder-decoder that can achieve capacity. But for $n = 2$ cells, we proposed an algorithm to find a message assignment for codes and proved that the algorithm achieves a message set cardinality equal to the codebook cardinality and is therefore optimal.

Determining the existence of a message assignment with $M \geq 2$ messages for codes over multiple cells for a given codebook is a special case of the Set M-Coloring problem. An important open problem is finding low-complexity algorithms to determine message assignments that can achieve high rates for a given codebook.

Appendix A
Lemma for Section III

Lemma A.1: Let $\mathbb{C}$ and $\mathbb{D}$ be closed subsets of $\mathbb{A}$. Let $\mathbb{E}_{\mathbf{x},\delta} = \{\mathbf{y} \in \mathbb{A} : \|\mathbf{y} - \mathbf{x}\| < \delta\}$ be an open ball with radius δ and center $\mathbf{x}$ and let $\|\mathbf{x} - \mathbf{y}\|$ denote the Euclidean distance between $\mathbf{x}$ and $\mathbf{y}$. Suppose $\mathbb{C} \subseteq \mathbb{D}$ and $\|\mathbb{C}\| = \|\mathbb{D}\| > 0$. If, $\forall\, \delta > 0$ and $\forall\, \mathbf{x} \in \mathbb{D}$, $\|\mathbb{D} \cap \mathbb{E}_{\mathbf{x},\delta}\| > 0$, then $\mathbb{C} = \mathbb{D}$.

Proof: Suppose the opposite is true; that is, suppose $\mathbb{C}$ is strictly contained in $\mathbb{D}$. Let $\mathbf{x}' \in \mathbb{A}$ be a point in $\mathbb{A}$ such that $\mathbf{x}' \in \mathbb{D}$, but $\mathbf{x}' \notin \mathbb{C}$. Since $\mathbb{C}$ is closed, there exists an open ball $\mathbb{E}_{\mathbf{x}',\delta'}$ such that $\mathbb{E}_{\mathbf{x}',\delta'} \cap \mathbb{C} = \emptyset$. $\|\mathbb{D} \cap \mathbb{E}_{\mathbf{x}',\delta'}\| > 0$ implies that $\|\mathbb{D}\| \geq \|\mathbb{C}\| + \|\mathbb{D} \cap \mathbb{E}_{\mathbf{x}',\delta'}\| > \|\mathbb{C}\|$, which contradicts the assumption that $\|\mathbb{C}\| = \|\mathbb{D}\|$. Therefore, $\mathbb{C} = \mathbb{D}$. ■

Appendix B
Proof of Lemma III.5

$$\Delta^{\langle n \rangle}(u) = \|\mathbb{A}\|^{-1} \cdot \int_{\mathbf{x} \in \mathbb{H}(u)} \mathrm{d}\mathbf{x}$$
$$= \int_0^{a_{n-1}} \int_0^{a_{n-2}} \cdots \int_0^{a_1} \left(1 - \frac{u}{\prod_{j=1}^{n-1}(1 - x_j)}\right) \mathrm{d}x_1 \cdots \mathrm{d}x_{n-1} \quad \text{(B.1)}$$

where

$$a_i \triangleq 1 - \frac{u}{\prod_{j=i+1}^{n-1}(1 - x_j)} \text{ for all } i = 1, 2, \ldots, n-1 \quad \text{(B.2)}$$

is the normalized volume of region $\mathbb{H}(u)$ when the number of cells is n. One can verify the claim for $n = 2$, namely

$$\Delta^{\langle 2 \rangle}(u) = \int_0^{1-u} \left(1 - \frac{u}{1 - x_1}\right) \mathrm{d}x_1 = 1 - u - u \ln\left(\frac{1}{u}\right). \quad \text{(B.3)}$$

Now, let the hypothesis be true for some $n-1$, $n \geq 3$. Then from (B.1),

$$\Delta^{\langle n \rangle}(u) = \int_0^{1-u} \Delta^{\langle n-1 \rangle}\left(\frac{u}{1 - x_{n-1}}\right) \mathrm{d}x_{n-1}$$
$$= \int_0^{1-u} \left(1 - \left(\frac{u}{1-x}\right) \sum_{i=0}^{n-2} \frac{1}{i!} \left[-\ln\left(\frac{u}{1-x}\right)\right]^i\right) \mathrm{d}x \quad \text{(B.4)}$$
$$= (1 - u) - \sum_{i=0}^{n-1} \frac{(-1)^i}{i!} \Psi_i(u) \quad \text{(B.5)}$$

where the equality labeled (B.4) is true from the induction hypothesis and

$$\Psi_i(u) \triangleq \int_0^{1-u} \frac{u}{1-x} \left[\ln\left(\frac{u}{1-x}\right)\right]^i \mathrm{d}x = -\frac{u}{i+1} (\ln u)^{i+1}. \quad \text{(B.6)}$$

Then, replacing (B.6) in (B.5), we get the required result for n. Hence, the lemma is true for all $n \geq 2$ by induction. ■

Appendix C
Proof of Proposition III.6

Set the derivative of $f_k(u)$ with respect to u to 0 and solve for u to get

$$u = 0 \text{ or } \left(1 - u - \frac{1}{\tau_k} u \ln u\right) = 0 \quad \text{(C.1)}$$

with $\tau_k = -\frac{k-1}{k}$ as defined in the statement of the proposition. For $u = 0$, $f_k(0) = 0$; thus $u_k^* \neq 0$. The other expression in (C.1) is equivalent to $e^{\frac{\tau_k}{u}} \frac{\tau_k}{u} = \tau_k e^{\tau_k}$. But, by the definition of the Lambert W function, $e^{W(\tau_k e^{\tau_k})} W(\tau_k e^{\tau_k}) = \tau_k e^{\tau_k}$; therefore, $u = \frac{\tau_k}{W(\tau_k e^{\tau_k})}$ is a root of (C.1). Since $\tau_k e^{\tau_k} \in \left(-\frac{1}{e}, -\frac{1}{2\sqrt{e}}\right] \subset \left(-\frac{1}{e}, 0\right)$, $W(\tau_k e^{\tau_k})$ is multi-valued and takes two values $W_0(\tau_k e^{\tau_k})$ and $W_{-1}(\tau_k e^{\tau_k})$ corresponding to the two branches of the Lambert W function. One can check that the root $\frac{\tau_k}{W_0(\tau_k e^{\tau_k})}$ equals 1 and is a local minimum. The other root, denoted as $u_k^* = \frac{\tau_k}{W_{-1}(\tau_k e^{\tau_k})}$, corresponds to the local maximum. ■

Appendix D
Proof of Proposition III.7

An asymptotic expression is needed for the last term in (68) as it depends on n. From [19],

$$\mathrm{P}(n, \alpha n) = \frac{1}{2} \operatorname{erfc}\left(-\zeta\sqrt{n/2}\right) - \mathrm{R}(n, \zeta), \tag{D.1}$$

where $\zeta = -\left(2\left(\alpha - 1 - \log_e \alpha\right)\right)^{\frac{1}{2}}$ for $\alpha < 1$, erfc is the complementary error function and function R has the following asymptotic expansion that is valid for all $\alpha \geq 0$:

$$\mathrm{R}(n, \zeta) \sim \frac{e^{-\frac{1}{2}n\zeta^2}}{\sqrt{2\pi n}} \sum_{k=0}^{\infty} d_k(\zeta) n^{-k} \tag{D.2}$$

where d_k is a constant that depends only on ζ and k. The following asymptotic expression for the complementary error function holds for $x \to \infty$:

$$\operatorname{erfc}(x) \sim \frac{e^{-x^2}}{x\sqrt{\pi}} \left[1 + \sum_{k=1}^{\infty} (-1)^k \frac{(2k-1)!!}{(2x^2)^k}\right], \tag{D.3}$$

where $(2k-1)!!$ is the double factorial. Thus,

$$\begin{aligned} \mathrm{P}(n, \alpha n) &= \frac{e^{-\frac{\zeta^2 n}{2}}}{\sqrt{2\pi n}} \sum_{k=0}^{\infty} \left((-1)^k \frac{(2k-1)!!}{\zeta^{2k+1}} - d_k(\zeta)\right) n^{-k}, \\ &\triangleq \frac{e^{-\frac{\zeta^2 n}{2}}}{\sqrt{2\pi n}} \sum_{k=0}^{\infty} \mathrm{D}_k(\zeta) n^{-k}, \end{aligned} \tag{D.4}$$

$$\begin{aligned} \frac{1}{n}\log_2\left[\mathrm{P}(n, \alpha n)\right] \sim &-\frac{\zeta^2}{2}\log_2(e) + \frac{\log_2(2\pi n)}{2n} \\ &+ \frac{\log_2\left(\sum_{k=0}^{\infty} \mathrm{D}_k(\zeta) n^{-k}\right)}{n}, \end{aligned} \tag{D.5}$$

and from (68),

$$\begin{aligned} \lim_{n\to\infty} \frac{1}{n} \log_2 S_2\left(e^{-\alpha n}\right) = \log_2 (q-1)^2 - (2\alpha - 1)\log_2(e) \\ + \log_2(\alpha). \end{aligned} \tag{D.6}$$

The total codebook cardinality in (D.6) is maximized when $\alpha = 0.5$ and is equal to

$$\lim_{n\to\infty} \frac{1}{n} \log_2 S_2\left(e^{-0.5n}\right) = \log_2 (q-1)^2 - 1. \tag{D.7}$$

For $\alpha > 1$, $\zeta = +\left(2\left(\alpha - 1 - \log_e \alpha\right)\right)^{\frac{1}{2}}$ so that $\lim_{n\to\infty} \frac{1}{n} \log_2\left[\mathrm{P}(n, \alpha n)\right] = 0$. ■

Appendix E
Proof of Proposition IV.2

For convenience of notation, we drop the asterisk from v_k^*. For $k \geq 2$, the optimal hyperbola parameter, v_k, satisfies $\Delta(v_k) = v_k \cdot \Delta(v_{k-1}) = v_k \cdot \Theta_k$ where $\Theta_k \triangleq \prod_{m=1}^{k-1} v_m$. Using Lemma III.5 for $n = 2$, we get $\frac{-1}{v_k} \cdot \exp\left(\frac{-1}{v_k}\right) = -e^{-(1+\Theta_k)} \triangleq \xi_k$. By the definition of Lambert W function, $-v_k^{-1} = W(\xi_k)$. However, $W_0(\xi_k) \neq -{v_k}^{-1}$ since $W_0(x) > -1$ for all x. By induction on k, ξ_k lies in the domain of W_{-1}. Therefore $v_k = -\left(W_{-1}(\xi_k)\right)^{-1} \in (0, 1]$. ■

Appendix F
Proof of correctness of Algorithm 2

Consider stage i where the algorithm assigns messages to points in the codebook for write i. Let $\mathbf{x}_k$ denote the k^{th} pivot point in the codebook $\mathcal{L}_{i-1}$ and $\tilde{V}_k = |\mathcal{L}_i(\mathbf{x}_k)|$ denote the number of points in codebook $\mathcal{L}_i$ accessible from $\mathbf{x}_k$. In this proof, we will show that at any iteration k of the i^{th} stage, the number of points in $\mathcal{L}_i(\mathbf{x}_k)$ that have not been assigned a message at any iteration $\ell < k$ is larger than the number of messages that point $\mathbf{x}_k$ cannot encode at the start of the k^{th} iteration; that is, there are enough points that have not been assigned a message such that $\mathbf{x}_k$ satisfies the condition in (104) at the end of the k^{th} iteration.

Let S_k^- (or S_k^+) be the number of points in $\mathcal{L}_i(\mathbf{x}_k)$ that were not assigned any messages up to any iteration $\ell < k$ (or $\ell \leq k$). The set $\mathcal{L}_{\text{lost}}(\mathbf{x}_{k+1})$ denotes the points in $\mathcal{L}_i$ that were accessible from $\mathbf{x}_k$ but are not accessible from $\mathbf{x}_{k+1}$. Partition $\mathcal{L}_{\text{lost}}(\mathbf{x}_{k+1})$ into sets of points that were assigned (or were not assigned) messages up to some iteration $\ell \leq k$. Denote the cardinality of these sets as $A_{k,k+1}$ and $S_{k,k+1}$, respectively. Finally, let $G_{k,k+1}$ be the number of points in $\mathcal{L}_i$ accessible from $\mathbf{x}_{k+1}$ but not accessible from $\mathbf{x}_k$. By the definitions above, for all ℓ,

$$\tilde{V}_\ell = \tilde{V}_{\ell-1} - (A_{\ell-1,\ell} + S_{\ell-1,\ell}) + G_{\ell-1,\ell}, \tag{F.1}$$

$$S_\ell^- = |\mathcal{L}_{\text{spare}}(\mathbf{x}_\ell)| = S_{\ell-1}^+ - S_{\ell-1,\ell} + G_{\ell-1,\ell} \geq S_{\ell,\ell+1}, \tag{F.2}$$

$$S_\ell^+ = S_\ell^- - A_{\ell-1,\ell} = S_\ell^- - |\mathcal{M}_{\text{lost}}(\mathbf{x}_\ell)|. \tag{F.3}$$

Now, from (F.1), (F.2), and (F.3),

$$S_{k+1}^- - S_0^- = \left(\tilde{V}_{k+1} - \tilde{V}_0\right) + A_{k,k+1} - A_{-1,0}. \tag{F.4}$$

Since $S_0^- = A_{-1,0} = \tilde{V}_0 = M_i$ and $\tilde{V}_{k+1} \geq M_i$ by definition of the first pivot point, we have

$$|\mathcal{L}_{\text{spare}}(\mathbf{x}_{k+1})| = S_{k+1}^- \geq A_{k,k+1} = |\mathcal{M}_{\text{lost}}(\mathbf{x}_{k+1})| \tag{F.5}$$

and $S_{k+1}^+ \geq 0$. Thus, at every iteration, the algorithm has enough spare points that have not yet been assigned a message to assign all the missing messages. ■

References

[1] R. L. Rivest and A. Shamir, "How to reuse a write-once memory," *Information and Control*, vol. 55, Dec. 1982.
[2] J. K. Wolf, A. D. Wyner, J. Ziv, and J. Körner, "Coding for a write-once memory," *AT&T Bell Labs. Tech. J.*, vol. 63, no. 6, pp. 1089 – 1112, 1984.
[3] A. Jiang, V. Bohossian, and J. Bruck, "Rewriting Codes for joint information storage in flash memories," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 5300 – 5313, Oct. 2010.
[4] S. Kayser, E. Yaakobi, P. H. Siegel, A. Vardy, and J. K. Wolf, "Multiple-write WOM-codes," in *Proc. 48th Ann. Allerton Conf. on Comm., Control and Computing*, Monticello, IL, Sep. 29 - Oct. 1, 2010.
[5] Y. Wu and A. Jiang, "Position modulation code for rewriting write-once memories," *IEEE Trans. Inf. Theory*, vol. 57, no. 6, pp. 3692 – 3697, Jun. 2011.
[6] E. Yaakobi, S. Kayser, P. H. Siegel, A. Vardy, and J. K. Wolf, "Efficient two-write WOM-codes," in *IEEE Inf. Theory Workshop*, Dublin, Ireland, Aug. 2010.
[7] Q. Huang, S. Lin, and K. A. S. Abdel-Ghaffar, "Error-correcting codes for flash coding," *IEEE Trans. Inf. Theory*, vol. 57, no. 9, pp. 6097 – 6108, Sep. 2011.
[8] R. Gabrys, E. Yaakobi, L. Dolecek, P. H. Siegel, A. Vardy, and J. K. Wolf, "Non-binary WOM-codes for multilevel flash memories," in *IEEE Inf. Theory Workshop*, Paraty, Brazil, pp. 40 – 44, Oct. 16 - 20, 2011.

[9] A. Shpilka,“Capacity achieving two-write WOM codes,” in *Latin American Symp. on Theor. Informatics*, Peru, Apr., 2012.

[10] Y. Cassuto and E. Yaakobi, “Short q-ary WOM codes with hot/cold write differentiation,” in *Proc. IEEE International Symp. on Inf. Theory*, Cambridge, MA, pp. 1391 – 1395, Jul. 1 - 6, 2012.

[11] B. M. Kurkoski, “Notes on a lattice-based WOM construction that guarantees two writes,” in *Proc. 34th Symp. Information Theory and Applications*, Ousyuku, Iwate, Japan, pp. 520 – 524, Nov. - Dec. 2011.

[12] F. Fu and A. J. Han Vinck, “On the capacity of generalized write-once memory with state transitions described by an arbitrary directed acyclic graph,” *IEEE Trans. Inf. Theory*, vol. 45, no. 1, pp. 308 – 313, Jan. 1999.

[13] A. Bhatia, A. R. Iyengar, and P. H. Siegel, “Multilevel 2-cell t-write codes,” in *Proc. IEEE Inf. Theory Workshop*, Lausanne, Switzerland, pp. 247 – 251, Sep. 3 - 7, 2012.

[14] B. Kurkoski, “Lattice-based WOM codebooks that allow two writes,” in *International Symp. on Inf. Theory and its Applications*, Honolulu, Hawaii, pp. 101 – 105, Oct. 28 - 31, 2012.

[15] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*, New York, NY, Springer-Verlag, 3rd ed., 1999.

[16] G. D. Forney, R. G. Gallager, G. R. Lang, F. M. Longstaff, and S. U. H. Qureshi, “Efficient modulation for band-limited channels,” *IEEE J. Sel. Areas Commun.*, JSAC-2, pp. 632 – 647, Sep. 1984.

[17] G. D. Forney and L.-F. Wei, “Multidimensional constellations – Part 1: Introduction, figures of merit, and generalized cross constellations,” *IEEE J. Sel. Areas Commun.*, JSAC-7, pp. 877 – 892, Aug. 1989.

[18] R. L. Corless, G. H. Gonnet, D. E. G. Hare, D. J. Jeffrey, and D. Knuth, “On the Lambert W function,” *Adv. in Comp. Math.* 5(4):329–359, 1996.

[19] N. M. Temme, “Uniform asymptotic expansions of the incomplete Gamma functions and the incomplete Beta function,” *Math. of Computation*, vol. 29, no. 132, pp. 1109 – 1114, Oct. 1975.

[20] J. J. Ashley, R. Karabed, and P. H. Siegel, “Complexity and sliding-block decodability,” *IEEE Trans. Inf. Theory*, vol. 42, no. 6, pp. 1925 – 1947, Nov. 1996.

[21] M. Garey, and D. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, San Fransisco, CA: W. H. Freeman, 1979.

[22] Y. Wu, “Low complexity codes for writing a write-once memory twice,” in *Proc. IEEE International Symp. on Inf. Theory*, Austin, TX, pp. 1928 – 1932, Jun. 13 - 18, 2010.

[23] M. Qin, E. Yaakobi, and P. H. ‘Siegel, “Constrained codes that mitigate inter-cell interference in read/write cycles for flash memories,” in *IEEE J. Sel. Areas Commun.*, 2014.

Aman Bhatia (S’11) is a Ph.D. candidate at the University of California San Diego. He received his B.Tech degree in Electrical Engineering from the Indian Institute of Technology, Kanpur, India, in 2009; and his M.S. degree in Electrical and Computer Engineering from the University of California San Diego in 2011. His interests include coding and information theory. His current work focuses on various coding techniques for data storage.

Minghai Qin (S’11) received the B.E. degree in Electronic and Electrical Engineering from Tsinghua University, Beijing, China, in 2009. He is currently pursuing the Ph.D. degree in Electrical Engineering from the Department of Electrical and Computer Engineering at the University of California, San Diego, where he is associated with the Center for Magnetic Recording Research.

Aravind R. Iyengar (S’09–M’12) received his B.Tech degree in Electrical Engineering from the Indian Institute of Technology Madras, Chennai, in 2007; his M.S. and Ph.D. degrees in Electrical Engineering from the University of California in San Diego, La Jolla, where he was affiliated with the Center for Magnetic Recording Research, in 2009 and 2012 respectively. He is currently with Qualcomm Technologies Inc., Santa Clara, where he is involved in the design of baseband modems. In 2006, he was a visiting student intern at the École Nationale Supérieure de l’Electronique et de ses Applications (ENSEA), Cergy, France. He was a visiting doctoral student at the Communication Theory Laboratory at the École Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland in 2010. His research interests are in the areas of information and coding theory, and in signal processing and wireless communications. A. R. Iyengar was the recipient of the Sheldon Schultz Prize for Excellence in Graduate Student Research at the University of California, San Diego in 2012.

Brian M. Kurkoski Brian M. Kurkoski was born in Portland, Oregon, USA. He received the B.S. degree from the California Institute of Technology in 1993, and then worked in industry for several years. He received the M.S. and Ph.D. degrees from the University of California San Diego in 2000 and 2004, respectively. He was at the University of Electro-Communications in Tokyo, Japan, first as a postdoctoral researcher from 2004 to 2006, and then as an Associate Professor from 2007 to 2012. He is currently an Associate Professor at the Japan Advanced Institute of Science and Technology (JAIST). Since 2010, he has been an associate editor for IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences. He has organized or been technical co-chair for several data storage tracks and workshops. His research interests include coding theory, information theory and communication theory and their application to storage systems.

Paul H. Siegel (M’82–SM’90–F’97) received the S.B. and Ph.D. degrees in mathematics from the Massachusetts Institute of Technology (MIT), Cambridge, in 1975 and 1979, respectively.

He held a Chaim Weizmann Postdoctoral Fellowship at the Courant Institute, New York University. He was with the IBM Research Division in San Jose, CA, from 1980 to 1995. He joined the faculty at the University of California, San Diego in July 1995, where he is currently Professor of Electrical and Computer Engineering in the Jacobs School of Engineering. He is affiliated with the Center for Magnetic Recording Research where he holds an endowed chair and served as Director from 2000 to 2011. His primary research interests lie in the areas of information theory and communications, particularly coding and modulation techniques, with applications to digital data storage and transmission.

Prof. Siegel was a member of the Board of Governors of the IEEE Information Theory Society from 1991 to 1996 and from 2009 to 2011. He was re-elected for another 3-year term in 2012. He served as Co-Guest Editor of the May 1991 Special Issue on “Coding for Storage Devices” of the IEEE Transactions on Information Theory. He served the same Transactions as Associate Editor for Coding Techniques from 1992 to 1995, and as Editor-in-Chief from July 2001 to July 2004. He was also Co-Guest Editor of the May/September 2001 two-part issue on “The Turbo Principle: From Theory to Practice” of the IEEE Journal on Selected Areas in Comminucations.

Prof. Siegel was co-recipient, with R. Karabed, of the 1992 IEEE Information Theory Society Paper Award and shared the 1993 IEEE Communications Society Leonard G. Abraham Prize Paper Award with B. H. Marcus and J. K. Wolf. With J. B. Soriaga and H. D. Pfister, he received the 2007 Best Paper Award in Signal Processing and Coding for Data Storage from the Data Storage Technical Committee of the IEEE Communications Society. He holds several patents in the area of coding and detection, and was named a Master Inventor at IBM Research in 1994. He is an IEEE Fellow and a member of the National Academy of Engineering.