Constant-Weight Gray Codes for Local Rank Modulation

Moshe Schwartz
Electrical and Computer Engineering
Ben-Gurion University
Beer Sheva 84105, Israel
schwartz@ee.bgu.ac.il

***Abstract*—We consider the local rank-modulation scheme in which a sliding window going over a sequence of real-valued variables induces a sequence of permutations. The local rank-modulation, as a generalization of the rank-modulation scheme, has been recently suggested as a way of storing information in flash memory.**

We study constant-weight Gray codes for the local rank-modulation scheme in order to simulate conventional multi-level flash cells while retaining the benefits of rank modulation. We provide necessary conditions for the existence of cyclic and cyclic optimal Gray codes. We then specifically study codes of weight 2 and upper bound their efficiency, thus proving that there are no such asymptotically-optimal cyclic codes. In contrast, we study codes of weight 3 and efficiently construct codes which are asymptotically-optimal.

I. Introduction

In a recent series of papers [16], [17], [21], [23], the rank-modulation scheme was suggested as a way of storing information in flash-memory devices. Basically, instead of a conventional multi-level flash cell in which the charge level of a single cell is measured and quantized to a symbol from the input alphabet, in the rank-modulation scheme the permutation induced by the relative charge levels of several cells is the stored information. The scheme, first described in [16] in the context of flash memory, works in conjunction with a simple cell-programming operation called "push-to-the-top", which raises the charge level of a single cell above the rest of the cells. It was suggested in [16] that this scheme eliminates the over-programming problem in flash memories, reduces corruption due to retention, and speeds up cell programming.

This is certainly not the first time permutations have been used for modulation purposes. Permutations have been used as codewords as early as the works of Slepian [20] (later extended in [1]), in which permutations were used to digitize vectors from a time-discrete memoryless Gaussian source, and Chadwick and Kurz [6], in which permutations were used in the context of signal detection over channels with non-Gaussian noise (especially impulse noise). Further early studies include works such as [1]–[3], [5], [7], [8]. More recently, permutations were used for communicating over powerlines (for example, see [22]), and for modulation schemes for flash memory [16], [17], [21], [23].

An important application for rank-modulation in the context of flash memory was described in [16]. A set of n cells, over which the rank-modulation scheme is applied, is used to simulate a single conventional multi-level flash cell with $n!$ levels corresponding to the alphabet $\{0,1,\dots,n!-1\}$. The simulated cell supports an operation which raises its value by 1 modulo $n!$. This is the only required operation in many rewriting schemes for flash memories (see [4], [13]–[15], [24]). This operation is realized by a Gray code traversing the $n!$ states where, physically, the transition between two adjacent states in the Gray code is achieved by using a single "push-to-the-top" operation.

Most generally, a gray code is a sequence of distinct elements from an ambient space such that adjacent elements in the sequence are "similar". Ever since their original publication by Gray [11], the use of Gray codes has reached a wide variety of areas. For a survey on Gray codes the reader is referred to [18].

A drawback to the rank-modulation scheme is the need for a large number of comparisons when reading the induced permutation from a set of n cell-charge levels. Instead, in a recent work [23] the n cells are partially viewed through a sliding window resulting in a sequence of small permutations. We call this the *local rank-modulation scheme*. The aim of this work is to study Gray codes for the local rank-modulation scheme. The paper is organized as follows: In Section II the exact setting, notation, and definitions are presented. We study, in Section III, necessary conditions for the existence of Gray codes for our setting. In Section IV we give constructions for Gray codes of low weight and study their efficiency. We conclude in Section V with a summary of the results. Due to space limitations, all proofs have been omitted, though the statements of supporting lemmas provide a sketch of the proof procedure for the main results.

II. Definitions and Notation

A. Local Rank Modulation

Let us consider a sequence of t real-valued variables, $\mathbf{c}=(c_0,c_1,\dots,c_{t-1})$, $c_i\in\mathbb{R}$, where we further assume $c_i\neq c_j$ for all $i\neq j$. The t variables induce a permutation $f_{\mathbf{c}}\in S_t$, where S_t denotes the set of all permutations over $[t]=\{1,2,\dots,t\}$. The permutation $f_{\mathbf{c}}$ is uniquely defined by the constraints $c_{f_{\mathbf{c}}(i)-1}>c_{f_{\mathbf{c}}(j)-1}$ for all $i<j$, i.e., if we sort $\mathbf{c}$ in descending order, $c_{j_1}>c_{j_2}>\cdots>c_{j_t}$ then $f_{\mathbf{c}}(i)=j_i+1$ for all $1\leqslant i\leqslant t$.

Given a sequence of n variables, $\mathbf{c}=(c_0,c_1,\dots,c_{n-1})$, we define a window of size t at position p to be $\mathbf{c}_{p,t}=$

978-1-4244-7891-0/10/$26.00 ©2010 IEEE

$(c_p, c_{p+1}, \dots, c_{p+t-1})$, where the indices are taken modulo n, and also $0 \leqslant p \leqslant n-1$, and $1 \leqslant t \leqslant n$.

We now define the *(s,t,n)-local rank-modulation (LRM) scheme*, which we do by defining the *demodulation* process. Let $s \leqslant t \leqslant n$ be positive integers, with $s|n$. Given a sequence of n distinct real-valued variables, $\mathbf{c} = (c_0, c_1, \dots, c_{n-1})$, the demodulation maps $\mathbf{c}$ to the sequence of n/s permutations from S_t as follows: $\mathbf{f}_\mathbf{c} = (f_{\mathbf{c}_{0,t}}, f_{\mathbf{c}_{s,t}}, f_{\mathbf{c}_{2s,t}}, \dots, f_{\mathbf{c}_{n-s,t}})$.

In the context of flash memory we shall consider the n variables, $\mathbf{c} = (c_0, c_1, \dots, c_{n-1})$, to be the charge-level readings from n flash cells. The demodulated sequence, $\mathbf{f}_\mathbf{c}$, will stand for the original information which was stored in the n cells. This approach will serve as the main motivation for this paper, as it was also for [16], [17], [21], [23].

We say a sequence $\mathbf{f}$ of n/s permutations over S_t is (s,t,n)-*LRM realizable* if there exists $\mathbf{c} \in \mathbb{R}^n$ such that $\mathbf{f} = \mathbf{f}_\mathbf{c}$, i.e., it is the demodulated sequence of $\mathbf{c}$ under the (s,t,n)-LRM scheme. Except for the degenerate case of $s = t$, not every sequence is realizable.

When $s = t = n$, the (n,n,n)-LRM scheme degenerates into a single permutation from S_n. This was the case in most of the previous works using permutations for modulation purposes. A slightly more general case, $s = t < n$ was discussed by Ferriera *et al.* [10] in the context of permutation trellis codes. Finally, the most general case was defined by Wang *et al.* [23] (though in a slightly different manner where indices are not taken modulo n, i.e., with no wrap-around). In [23], the sequence of permutations was studied under a charge-difference constraint called *bounded rank-modulation*, and mostly with parameters $s = t-1$, i.e., an overlap of one position between adjacent windows.

Finding out the induced permutation from a sequence of t real-valued readings requires at least $\Omega(t \log t)$ comparisons. Thus, to get the simplest hardware implementation we will consider the case of $t = 2$ throughout the paper. The only non-trivial case to consider is therefore $s = 1$, i.e., a $(1,2,n)$-LRM scheme. Demodulated sequences of permutations in this scheme contain only the permutations $[1,2]$ and $[2,1]$, and a single comparison between the charge levels of two adjacent flash memory cells is required to find the permutation. We will conveniently associate the logical value 1 with the permutation $[1,2]$, and 0 with $[2,1]$, thus forming a simple mapping between length n binary sequences and permutation sequences from the $(1,2,n)$-LRM scheme. It is easily seen that the only two binary sequences not mapped to $(1,2,n)$-LRM sequences are the all-ones and all-zeros sequences.

B. Gray Codes for $(1,2,n)$-LRM

Generally speaking, a *Gray code*, G, is a sequence of distinct states (codewords), $G = g_0, g_1, \dots, g_{N-1}$, from an ambient state space, $g_i \in S$, such that adjacent states in the sequence differ by a "small" change. What constitutes a "small" change usually depends on the code's application.

Since we are interested in building Gray codes for flash memory devices with the $(1,2,n)$-LRM scheme, our ambient space, which we denote as $S(n)$, is the set of all realizable sequences under $(1,2,n)$-LRM. This is simply the set of all the binary sequences of length n, excluding the all-ones and all-zeros sequences, i.e.,

$$S = S(n) = \{0,1\}^n - \{0^n, 1^n\}.$$

Each of the codewords, $g_i \in G$, is a string of n bits which we shall denote as $g_i = g_{i,0}, g_{i,1}, \dots, g_{i,n-1}$. Throughout the paper we will assume the index j in $g_{i,j}$ is taken modulo n, and when appropriate, the index i is taken modulo N.

The transition between adjacent states in the Gray code is directly motivated by the flash memory application, and was previously described and used in [16]. This transition is the "push-to-the-top" operation, which takes a single flash cell and raises its charge level above all others.

In our case, however, since we are considering a *local* rank-modulation scheme, the "push-to-the-top" operation merely raises the charge level of the selected cell above those cells which are comparable with it. As the window size is $t = 2$, these cells are the ones directly before and after the selected cell. Thus, we define the set of allowed transitions as $T = \{\tau_0, \tau_1, \dots, \tau_{n-1}\}$, which is a set of functions, $\tau_j : S \to S$, where τ_j represents a "push-to-the-top" operation performed on the j-th cell. If $v = v_0 v_1 \dots v_{n-1} \in S(n)$, then $v' = v'_0 v'_1 \dots v'_{n-1} = \tau_j(v)$ if

$$v'_k = \begin{cases} 0 & k = j \\ 1 & k \equiv j+1 \pmod{n} \\ v_k & \text{otherwise.} \end{cases}$$

Definition 1. *A* Gray code, G, for $(1,2,n)$-LRM *is a sequence of distinct length-n binary codewords, $G = g_0, g_1, \dots, g_{N-1}$, where $g_i \in S(n)$. For all $0 \leqslant i \leqslant N-2$, we further require that $g_{i+1} = \tau_j(g_i)$ for some j. If $g_0 = \tau_j(g_{N-1})$ for some j, then we say the code is* cyclic*. We call N the* size *of the code, and say G is* optimal *if $N = 2^n - 2$.*

When we perform a "push-to-the-top" operation on the j-th cell, let us denote its initial charge level as c_j, and its resulting charge level as c'_j. We set c'_j to $\max\{c_{j-1}, c_{j+1}\} + \delta$, where $\delta > 0$. Two important issues of concern are the difference in charge levels involved in a "push-to-the-top" operation, and cell saturation. In the former, the higher $c'_j - c_j$ is, the more risk of disturbing neighboring cells, while in the latter, the higher we set c'_j, the less number of updates to the cell before it saturates. Both concerns benefit from a value of δ as low as possible. Let us assume that a limited resolution exists and thus δ is bounded from below by a constant, which w.l.o.g., we can assume is 1 (after a proper scaling).

Let us now assume an optimal setting in which a "push-to-the-top" operation on the j-th cell sets $c'_j = \max\{c_{j-1}, c_{j+1}\} + 1$. A general Gray code for $(1,2,n)$-LRM may result in $c'_j - c_j$ to be exponential in n, for some transition from g_i to g_{i+1}. The same motivation in the case of (n,n,n)-LRM was discussed in [16], where a balanced variant of Gray codes was constructed to avoid the problem. We present a different variant of Gray codes to address the same issue.

First, for any binary string $v = v_0 v_1 \dots v_{n-1}$, we call the number of 1's in v the *weight* of v and denote it as $\mathrm{wt}(v)$. We also denote by $S(n, w)$ the set of length-n binary strings of weight w. We now define our variant of Gray codes:

Definition 2. *A* constant-weight Gray code for $(1, 2, n)$-LRM, *$G = g_0, g_1, \dots, g_{N-1}$, is a Gray code for $(1, 2, n)$-LRM for which $g_i \in S(n, w)$ for all i.*

Definition 3. *Let G be a constant-weight Gray code for $(1, 2, n)$-LRM with weight w and size N. The* efficiency *of G is defined as $\mathrm{Eff}(G) = N/\binom{n}{w}$. If $\mathrm{Eff}(G) = 1$ then we say G is* optimal. *If $\mathrm{Eff}(G) = 1 - o(1)$, where $o(1)$ denotes a function that tends to 0 as $n \to \infty$, then we say G is* asymptotically optimal.

The transitions between adjacent states in the constant-weight variant have a very simple form: a size-2 window in g_i which contains 10 is transformed in g_{i+1} into 01, i.e., "pushing" a 1 to the right. Since we are interested in creating cyclic counters, we will be interested in cyclic Gray codes. It should be noted that Gray codes with a weaker restriction, allowing 1 to be pushed in either direction, have been studied in the past (see [12] and references therein). It can shown that under the constant-weight restriction, for any "push-to-the-top" operation, $c'_j - c_j \leqslant \left\lceil \frac{\max\{w, n-w\}}{\min\{w, n-w\}} \right\rceil$.

III. Necessary Conditions

We first present a simple necessary condition for the existence of a cyclic Gray code, and then expand it in the case of cyclic optimal codes.

Theorem 4. *Let G be a cyclic constant-weight Gray code of size N for $(1, 2, n)$-LRM. Then $n | N$.*

Proof: We prove the claim using a first-moment coloring argument. For any $v = v_0 v_1 \dots v_{n-1} \in S(n, w)$, we define the color of v as

$$\chi(v) = \left(\sum_{j=0}^{n-1} j \cdot v_j \right) \bmod n.$$

If $v, v' \in S(n, w)$ and $v' = \tau_j(v)$ for some j, then it follows that $\chi(v') \equiv \chi(v) + 1 \pmod{n}$.

Let us now denote $G = g_0, g_1, \dots, g_{N-1}$. By the previous argument, $i \equiv i' \pmod{n}$ if an only if $\chi(g_i) = \chi(g_{i'})$. Since the code is cyclic, it follows that $N \equiv 0 \pmod{n}$. ∎

We can use Theorem 4 to rule out the existence of cyclic optimal codes in certain cases.

Theorem 5. *If w is a prime, then there are no cyclic optimal weight-w Gray codes for $(1, 2, n)$-LRM for which $\gcd(n, w) \neq 1$.*

The divisibility condition set in Theorem 4 is not strong enough. For example, if we take $n = 12$ and $w = 6$, then indeed $12 | \binom{12}{6}$, and the possible existence of a cyclic optimal code with these parameters is not ruled out. However, by the conditions described in the following theorem it is ruled out.

Theorem 6. *If a cyclic optimal constant-weight Gray code for $(1, 2, n)$-LRM exists, then there are exactly $\binom{n}{w}/n$ strings of each color in $S(n, w)$.*

The following theorem may be thought of as an extension of Theorem 5 to the case of w not a prime.

Theorem 7. *For any fixed weight w, there are at most a finite a number of cyclic optimal weight-w Gray codes for $(1, 2, n)$-LRM for which $\gcd(n, w) \neq 1$.*

IV. Low-Weight Analysis and Constructions

In this section we study constant-weight Gray codes for $(1, 2, n)$-LRM having low weight, $w \leqslant 3$ (and by flipping bits and reversing strings, for all $w \geqslant n - 3$). In the first trivial case of $w = 1$, there exists a cyclic optimal code for all n. As we shall later see, the next two cases $w = 2, 3$ behave radically different: for $w = 2$ we will show that even cyclic *asymptotically-optimal* codes do not exist, while for $w = 3$ we will construct cyclic asymptotically-optimal codes.

A. The Case of $w = 2$

For the case of $w = 2$ a brute-force approach will suffice. For all $n \geqslant 2$, let us define the graph $\mathcal{G}_n$ whose vertex set is $S(n, 2)$ and an edge $v \to v'$ exists iff $v' = \tau_j(v)$ for some $0 \leqslant j \leqslant n - 1$.

For convenience, we index the vertices in the following way: $v_{k,\ell}$, where $1 \leqslant k \leqslant (n-1)/2$ and $0 \leqslant \ell \leqslant n - 1$, denotes the vertex corresponding to the string having 1's in positions ℓ and $\ell + k$. We shall conveniently refer to the first index as the *row* index, and the second index as the *column* index.

Using this indexing method the graph $\mathcal{G}_n$ takes on a simple form for odd $n \geqslant 5$ (the case $n = 3$ is more degenerate):

- A vertex of the form $v_{1,\ell}$ has a single outgoing edge to $v_{2,\ell}$.
- A vertex of the form $v_{k,\ell}$, $1 < k < (n-1)/2$, has two outgoing edges to $v_{k+1,\ell}$ and $v_{k-1,\ell+1}$.
- A vertex of the form $v_{(n-1)/2,\ell}$ has two outgoing edges to $v_{(n-3)/2,\ell+1}$ and $v_{(n-1)/2,\ell+(n+1)/2}$.

It is now evident that there is a one-to-one correspondence between simple paths in $\mathcal{G}_n$ and Gray codes. A simple construction for an optimal code which is (in general) *not cyclic* is the following.

Construction 1. *Let $n \geqslant 3$ be an odd integer. We construct the following code $G = g_0, g_1, \dots, g_{N-1}$. We first set $g_0 = v_{1,0}$, and then set g_{i+1} as a function of $g_i = v_{k,\ell}$ according to the following rules:*

- *If k is odd and $k < (n-1)/2$, then $g_{i+1} = v_{k+1,\ell}$.*
- *If k is odd and $k = (n-1)/2$, then $g_{i+1} = v_{k,\ell+(n+1)/2}$.*
- *If k is even and $\ell < n - k/2$, then $g_{i+1} = v_{k-1,\ell+1}$.*
- *If k is even and $\ell = n - k/2$, then $g_{i+1} = v_{k+1,\ell}$.*

Theorem 8. *The code from Construction 1 is an optimal constant-weight Gray code for $(1, 2, n)$-LRM with $w = 2$.*

Theorem 9. *Let G be a cyclic constant-weight Gray code for $(1, 2, n)$-LRM with $w = 2$, $n \geqslant 7$. Then $\mathrm{Eff}(G) \leqslant \frac{3}{4} + o(1)$.*

While the upper bound on the efficiency presented in Theorem 9 is $\frac{3}{4} + o(1)$, we conjecture that it actually is $o(1)$.

B. The Case of $w = 3$

In this section we turn to constructing asymptotically-optimal cyclic constant-weight Gray codes for $(1,2,n)$-LRM with $w = 3$. The construction will use a method originally used for constructing single-track Gray codes in [9] and later in [19]. In fact, the resulting codes will have the single-track property as well.

If $v = v_0 v_1 \dots v_{n-1}$ is a length n word over some alphabet, let E denote the *cyclic-shift operator* defined as:

$$Ev = v_{n-1} v_0 v_1 \dots v_{n-2}.$$

The orbits under E are called *necklaces*. They are said to be *full period* if the smallest positive integer i such that $E^i v = v$ is $i = n$. A full-period necklace contains n distinct strings.

We say a Gray code $G = g_0, g_1, \dots, g_{N-1}$ has the *single-track* property if in the matrix whose i-th row is g_i, all the columns are cyclic shifts of each other. A variant of the following method was suggested in [9] for constructing single-track Gray codes, and it applies equally-well to our set of allowed transitions.

Lemma 10. *Let $G' = g'_0, g'_1, \dots, g'_{N'-1}$ be a Gray code for $(1,2,n)$-LRM where $g'_{i+1} = \tau_{j_i}(g'_i)$ for all $0 \leqslant i \leqslant N'-2$. If the strings in G' are representatives of distinct full-period necklaces, and $E^\ell g'_0 = \tau_{j_{N'-1}} g'_{N'-1}$, $\gcd(\ell, n) = 1$, then the following is a cyclic single-track Gray code:*

$$G = G', E^\ell G', E^{2\ell} G', \dots, E^{(n-1)\ell} G',$$

where $E^j G' = E^j g'_0, \dots, E^j g'_{N'-1}$.

We define the mapping $\psi : S(n,3) \to \mathbb{Z}_n^3$ as follows: for a binary string v of length n and weight 3 with 1's in positions $0 \leqslant i_0 < i_1 < i_2 \leqslant n-1$, let

$$\psi(v) = (i_1 - i_0, i_2 - i_1, i_0 - i_2)$$

where subtraction is made modulo n. The set $\{\psi(v) \mid v \in S(n,3)\}$ is the set of points $(d_0, d_1, d_2) \in \mathbb{Z}^3$ that are on the hyperplane $d_0 + d_1 + d_2 = n$ restricted to $1 \leqslant d_0, d_1, d_2 \leqslant n-2$. We call $\psi(v)$ the *configuration* of v. We note that if $\gcd(n,3) = 1$, then $S(n,3)$ contains only full-period strings, and otherwise, all strings are full-period except those with configuration $(n/3, n/3, n/3)$. We denote by $S^*(n,3)$ the set of full-period strings from $S(n,3)$.

Since $\psi(v)$, $E\psi(v)$, and $E^2\psi(v)$, (corresponding to a cyclic rotation of the axes of $\mathbb{Z}^3$), represent strings from the same necklace, for any $v \in S^*(n,3)$, let $\psi'(v)$ stand for the unique $(d_0, d_1, d_2) \in \{\psi(v), E\psi(v), E^2\psi(v)\}$ for which $d_1 \leqslant \lfloor n/3 \rfloor$ and $d_2 > \lfloor n/3 \rfloor$. Thus, there is a simple one-to-one mapping from $\{\psi'(v) \mid v \in S^*(n,3)\}$ to the set of full-period necklaces. We call $\psi'(v)$ the *canonical configuration* of v.

Simple counting reveals that there are a total of $\frac{(n-1)(n-2)}{2}$ configurations. When $\gcd(n,3) = 1$ there are $\frac{(n-1)(n-2)}{6} = \frac{1}{n}\binom{n}{3}$ canonical configurations which is exactly the number of weight-3 full-period necklaces. When $\gcd(n,3) \neq 1$, there are $\frac{(n-1)(n-2)-2}{6}$ canonical configurations.

Lemma 11. *Let $\Delta = (d_0, d_1, d_2)$ be a canonical configuration, and assume $\Delta' \in \{(d_0+1, d_1-1, d_2), (d_0, d_1+1, d_2-1), (d_0-1, d_1, d_2+1)\}$ is also a canonical configuration. Then for any $v \in S^*(n,3)$ such that $\psi'(v) = \Delta$ there exists $v' \in S^*(n,3)$ such that $\psi'(v') = \Delta'$ and $v' = \tau_j(v)$ for some $0 \leqslant j \leqslant n-1$.*

We now intend to find a long cycle over canonical configurations which, by Lemma 11, will result in a Gray code of representatives of distinct full-period necklaces. The latter will be used with Lemma 10 to generate a cyclic constant-weight Gray code for $(1,2,n)$-LRM.

Construction 2. *Let $n \geqslant 9$ be an integer. We construct the following sequence of canonical configurations $\Gamma = \Delta_0, \Delta_1, \dots, \Delta_{N'-1}$. We first set $\Delta_0 = (1, 1, n-2)$, and then set Δ_{i+1} as a function of $\Delta_i = (d_0, d_1, d_2)$ according to the following rules:*

- *If $d_0 = 1$ and $d_1 < 3\lfloor \lfloor n/3 \rfloor / 3 \rfloor$, then set $\Delta_{i+1} = (d_0, d_1+1, d_2-1)$.*
- *Else, if $d_1 \equiv 0 \pmod 3$, then set $\Delta_{i+1} = (d_0+1, d_1-1, d_2)$.*
- *Else, if $d_1 \equiv 2 \pmod 3$ and $d_2 > \lfloor n/3 \rfloor + 1$, then set $\Delta_{i+1} = (d_0, d_1+1, d_2-1)$.*
- *Else, if $d_1 \equiv 2 \pmod 3$ and $d_2 = \lfloor n/3 \rfloor + 1$ and $d_1 > 1$, then set $\Delta_{i+1} = (d_0+1, d_1-1, d_2)$.*
- *Else, if $d_1 \equiv 1 \pmod 3$ and $d_0 > 2$, then set $\Delta_{i+1} = (d_0-1, d_1, d_2+1)$.*
- *To complete the cycle, if $\Delta_i = (1, 2, n-3)$, then set $\Delta_{i+1} = (1, 1, n-2)$.*

Figure 1 shows an illustration of Construction 2.

Lemma 12. *The path from Construction 2 visits only canonical configurations, each visited no more than once, and has length N' given by*

$$N'(n) = \begin{cases} \frac{n^2-5n+18}{6} & n \equiv 0 \pmod 9 \\ \frac{n^2-5n+22}{6} & n \equiv 1 \pmod 9 \\ \frac{n^2-5n+24}{6} & n \equiv 2 \pmod 9 \\ \frac{n^2-7n+30}{6} & n \equiv 3 \pmod 9 \\ \frac{n^2-7n+30}{6} & n \equiv 4 \pmod 9 \\ \frac{n^2-7n+28}{6} & n \equiv 5 \pmod 9 \\ \frac{n^2-9n+36}{6} & n \equiv 6 \pmod 9 \\ \frac{n^2-9n+32}{6} & n \equiv 7 \pmod 9 \\ \frac{n^2-9n+26}{6} & n \equiv 8 \pmod 9 \end{cases} \qquad (1)$$

Lemma 13. *Let $G' = g'_0, g'_1, \dots, g'_{N'-1}$ be a list of strings from $S^*(n,3)$ (whose existence is guaranteed by Lemma 11) such that $\Gamma = \psi'(g'_0), \psi'(g'_1), \dots, \psi'(g'_{N'-1})$ is the cyclic path from Construction 2. Let g^* be the string (whose existence is guaranteed by Lemma 11) such that $\psi'(g^*) = \psi'(g'_0)$ and $g^* = \tau_j(g'_{N'-1})$. Then $g^* = E^{N'/3} g'_0$.*

Theorem 14. *For all $n \geqslant 9$ such that $\gcd(n, N'(n)/3) = 1$, where $N'(n)$ is given by (1), there exists a cyclic constant-weight Gray code for $(1,2,n)$-LRM with $w = 3$ and size $N = n \cdot N'(n)$, which is also single-track.*

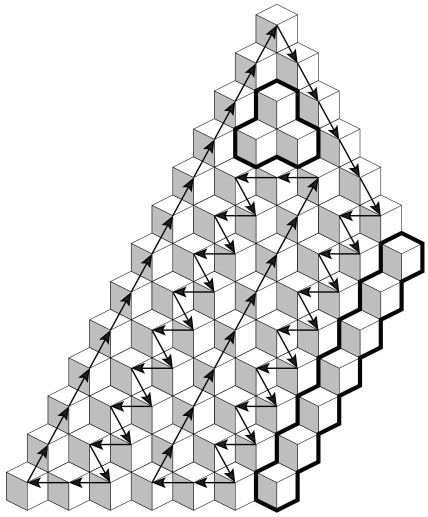

Figure 1. The path from Construction 2 over the canonical configurations for $n = 22$. The unvisited configurations are surrounded by a thick frame.

Lemma 15. *There are infinite values of $n \in \mathbb{N}$ for which $\gcd(n, N'(n)/3) = 1$. More specifically, it suffices that n satisfies one of the following:*

- $n \equiv 7, 11 \pmod{18}$
- $n \equiv 13, 31, 49, 67 \pmod{90}$
- $n \equiv 5, 23, 41, 59, 95, 113 \pmod{126}$
- $n \equiv 1, 19, 37, 73, 91, 109, 127, 145, 163, 181 \pmod{198}$
- $n \equiv 17, 35, 53, 71, 89, 107, 125, 161, 179, 197, 215, 233 \pmod{234}$

We note that the conditions described in Lemma 15 are not the only cases in which $\gcd(n, N'(n)/3) = 1$, but are just the ones easy to derive. For instance, when $n = 27$, we have $\gcd(n, N'(n)/3) = \gcd(27, 34) = 1$.

Corollary 16. *There exists an infinite family $\{G_i\}$ of constant-weight Gray codes for $(1, 2, n_i)$-LRM with $w = 3$, $n_{i+1} > n_i$, for which $\mathrm{Eff}(G_i) = 1 - o(1)$.*

The codes from Theorem 14 turn out to be optimal in the cases of $n = 10, 11$ with sizes $N = 120, 165$ respectively.

V. Conclusion

We presented the general framework of (s, t, n)-local rank modulation and focused on the specific case of $(1, 2, n)$-LRM which is both the least-hardware-intensive, and the simplest one to translate between binary strings and permutations. We studied constant-weight Gray codes, which guarantee a bounded charge difference in any "push-to-the-top" operation. These are used to simulate conventional multi-level flash cells.

Using coloring and counting arguments we derived necessary conditions for the existence of cyclic and cyclic optimal constant-weight Gray codes for $(1, 2, n)$-LRM. While cyclic optimal Gray codes exist (trivially) for $w = 1$, we showed that for $w = 2$ their efficiency is upper bounded by $\frac{3}{4} + o(1)$. In contrast, for $w = 3$ asymptotically-optimal codes exist with efficiency $1 - o(1)$. The codes we constructed also come with a relatively simple updating algorithm.

References

[1] T. Berger, F. Jelinek, and J. K. Wolf, "Permutation codes for sources," *IEEE Trans. on Inform. Theory*, vol. IT-18, no. 1, pp. 160–169, Jan. 1972.

[2] I. F. Blake, "Permutation codes for discrete channels," *IEEE Trans. on Inform. Theory*, vol. 20, pp. 138–140, 1974.

[3] I. F. Blake, G. Cohen, and M. Deza, "Coding with permutations," *Inform. and Control*, vol. 43, pp. 1–19, 1979.

[4] V. Bohossian, A. Jiang, and J. Bruck, "Buffer coding for asymmetric multi-level memory," in *Proceedings of the 2007 IEEE International Symposium on Information Theory (ISIT2007), Nice, France*, Jun. 2007, pp. 1186–1190.

[5] H. Chadwick and I. Reed, "The equivalence of rank permutation codes to a new class of binary codes," *IEEE Trans. on Inform. Theory*, vol. 16, no. 5, pp. 640–641, 1970.

[6] H. D. Chadwick and L. Kurz, "Rank permutation group codes based on Kendall's correlation statistic," *IEEE Trans. on Inform. Theory*, vol. IT-15, no. 2, pp. 306–315, Mar. 1969.

[7] G. Cohen and M. Deza, "Decoding of permutation codes," in *Intl. CNRS Colloquium, July, France*, 1977.

[8] M. Deza and P. Frankl, "On maximal numbers of permutations with given maximal or minimal distance," *J. Combin. Theory Ser. A*, vol. 22, 1977.

[9] T. Etzion and K. G. Paterson, "Near optimal single-track Gray codes," *IEEE Trans. on Inform. Theory*, vol. 42, no. 3, pp. 779–789, May 1996.

[10] H. C. Ferriera, A. J. H. Vinck, T. G. Swart, and I. de Beer, "Permutation trellis codes," *IEEE Trans. on Communications*, pp. 1782–1789, Nov. 2005.

[11] F. Gray, "Pulse code communication," March 1953, U.S. Patent 2632058.

[12] T. Hough and F. Ruskey, "An efficient implementation of the Eades, Hickey, Read adjacent interchange combination generation algorithm," *J. of Comb. Math. and Comb. Comp.*, vol. 4, pp. 79–86, 1988.

[13] A. Jiang, V. Bohossian, and J. Bruck, "Floating codes for joint information storage in write asymmetric memories," in *Proceedings of the 2007 IEEE International Symposium on Information Theory (ISIT2007), Nice, France*, Jun. 2007, pp. 1166–1170.

[14] A. Jiang and J. Bruck, "Joint coding for flash memory storage," in *Proceedings of the 2008 IEEE International Symposium on Information Theory (ISIT2008), Toronto, Canada*, Jul. 2008, pp. 1741–1745.

[15] A. Jiang, M. Langberg, M. Schwartz, and J. Bruck, "Universal rewriting in constrained memories," in *Proceedings of the 2009 IEEE International Symposium on Information Theory (ISIT2009), Seoul, Korea*, Jun. 2009, pp. 1219–1223.

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

[17] A. Jiang, M. Schwartz, and J. Bruck, "Correcting charge-constrained errors in the rank-modulation scheme," *IEEE Trans. on Inform. Theory*, 2010, to appear.

[18] C. D. Savage, "A survey of combinatorial Gray codes," *SIAM Rev.*, vol. 39, no. 4, pp. 605–629, Dec. 1997.

[19] M. Schwartz and T. Etzion, "The structure of single-track Gray codes," *IEEE Trans. on Inform. Theory*, vol. 45, no. 7, pp. 2383–2396, Nov. 1999.

[20] D. Slepian, "Permutation modulation," in *Proc. of the IEEE*, vol. 53, no. 3, 1965, pp. 228–236.

[21] I. Tamo and M. Schwartz, "Correcting limited-magnitude errors in the rank-modulation scheme," Jul. 2009, submitted to the IEEE Trans. on Inform. Theory. [Online]. Available: http://arxiv.org/pdf/0907.3387v2

[22] H. Vinck, J. Haering, and T. Wadayama, "Coded M-FSK for power line communications," in *Proceedings of the 2000 IEEE International Symposium on Information Theory (ISIT2000), Sorrento, Italy*, 2000, p. 137.

[23] Z. Wang, A. Jiang, and J. Bruck, "On the capacity of bounded rank modulation for flash memories," in *Proceedings of the 2009 IEEE International Symposium on Information Theory (ISIT2009), Seoul, Korea*, Jun. 2009, pp. 1234–1238.

[24] E. Yaakobi, P. H. Siegel, and J. K. Wolf, "Buffer codes for multi-level flash memory," in *Proceedings of the 2008 IEEE International Symposium on Information Theory (ISIT2008), Toronto, Canada*, 2008, poster.