Permutation Codes Correcting a Single Burst Deletion II: Stable Deletions

Yeow Meng Chee, San Ling, Tuan Thanh Nguyen, Van Khu Vu, and Hengjia Wei
School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore
Emails: {ymchee, lingsan, nguyentu001, vankhu001, hjwei}@ntu.edu.sg

***Abstract*—We construct permutation codes capable of correcting bursts of stable deletions. For correcting a single burst of exactly s stable deletions, our code has size $\frac{sn!}{((2s)!n)^2}$, while the upper bound is $\frac{n!}{s!(n-s+1)}$. We also construct permutation codes for the cases of single burst of up to s stable deletions, and up to b bursts of at most s stable deletions each.**

I. Introduction

Error-correcting codes for permutations have attracted recent attention due to their application in rank modulation schemes for flash memories. Flash memories store information in arrays of memory cells. In traditional single-level cell (SLC) devices each cell stores only one bit of information while newer flash memories have multi-level cells (MLC) that can store more than one bit per cell by choosing between multiple levels of electrical charge to apply to the floating gates of its cells [4]. This increases the storage density of flash memories. However, since the charge levels in an MLC are closer together than in an SLC, MLC flash memories are more prone to errors due to charge leakage. In order to combat this kind of errors, Jiang *et al.* [7] proposed a permutation coded rank modulation scheme. In this setup, information is carried by the relative values between the cells rather than by their absolute levels. After the work of [7], the permutation coded rank modulation scheme has been extended to tolerate other kinds of errors [1, 5, 8, 15].

Recently, Gabrys *et al.* [6] considered deletion errors in rank modulated flash memories. These errors can occur when cells are corrupted and the charge levels cannot be read correctly. More specifically, in a rank modulated flash memory, information is written in blocks of n cells, and is represented as the relative values of the charge levels. Hence, each block stores a permutation of length n. A *deletion* occurs when the charge level in a cell cannot be read and the location of that cell is not known. Deletions can be further classified according to how the deletion errors affect the information stored in the remaining cells:

(i) *Stable deletion*: In such a deletion, the absolute values of the cells are known.
(ii) *Unstable deletion*: In such a deletion, only the relative values of the remaining cells are known. This results in a new permutation of length $n-1$.

Example 1. Suppose that the third symbol 3 is deleted from the permutation $(7,5,3,2,4,6,1)$. In a stable deletion, the remaining components give $(7,5,2,4,6,1)$ whereas in an unstable deletion, the remaining components give $(6,4,2,3,5,1)$.

Gabrys *et al.* [6] showed that codes based upon the Ulam distance can be used to correct stable deletions. They also constructed a family of asymptotically optimal codes capable of correcting a single unstable deletion.

More recently Chee *et al.* [3] studied the related problem of deletion bursts in rank modulated flash memories, that is, a block of deletions that occur in consecutive cells. The authors presented a class of permutation codes that can correct a single burst of up to s unstable deletions, for general s. The motivation behind considering bursts of deletions is that as flash memories scale, the parasitic capacitance of adjacent cells increases, which can cause corruptions in a cell to bleed to adjacent cells, through capacitative coupling [9, 12].

Our focus in this paper is on single burst of stable deletions. We construct the first known family of permutation codes capable of correcting a single burst of exactly s stable deletions. The size of our codes is $\frac{sn!}{((2s)!n)^2}$, while the upper bound is $\frac{n!}{s!(n-s+1)}$. This is extended to a family of codes capable of correcting a single burst of at most s stable deletions whose size is $\frac{s!n!}{((4s)!)^2 n^{2s-1}}$. We also give an approach to correct δn bursts of at most s stable deletions each. Our codes have rate $1-\delta(s+1)$.

II. Preliminaries

A. Definitions

For integers $a \leq b$, $[a,b]$ denotes the set $\{a, a+1, a+2, \ldots, b\}$. Let n be a positive integer and $\mathcal{S}_n$ be the set of all permutations on the set $[1,n]$.

For a permutation $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n) \in \mathcal{S}_n$ and a set of positions $I \subseteq [1,n]$, define $\sigma(I) = \{\sigma_i : i \in I\}$. For $a \in [1,n]$ and $I \subseteq [1,n]$, the integer $a(I) \in [1,n]$ is defined so that $a(I) = a - |\{i \in I : i < a\}|$. For example, if $\sigma = (5,1,4,2,6,3)$ and $I = \{3,4\}$, then $\sigma(I) = \{2,4\}$, $5(\sigma(I)) = 5-2 = 3$, $1(\sigma(I)) = 1-0 = 1$, $6(\sigma(I)) = 6-2 = 4$ and $3(\sigma(I)) = 3-1 = 2$.

Definition 1. Assume that $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n) \in \mathcal{S}_n$ and $I \subseteq [1,n]$ be a set of size s. We say the permutation

978-1-5090-4096-4/17/$31.00 ©2017 IEEE

σ suffers s *stable deletions* in I, resulting in the vector $\tilde{\sigma} = (\tilde{\sigma}_1, \tilde{\sigma}_2, \ldots, \tilde{\sigma}_{n-s})$ if for all $k \in [1, n] \setminus I$ and $i = k(I)$, we have $\tilde{\sigma}_i = \sigma_k$. If $I = [a, b]$ for some a and b, then σ suffers a *stable burst deletion* of length s.

For example, let $\sigma = (2, 1, 3, 5, 6, 4)$ and $I = \{3, 4\}$. If σ suffers a stable deletion in I, then it results in $(2, 1, 6, 4)$.

Definition 2. A code $\mathcal{C} \subseteq \mathcal{S}_n$ is called an *s-SD permutation code* if it can correct up to s stable deletions, or an *s-SBD permutation code* if it can correct a single stable burst deletion of length s, or an *≤s-SBD permutation code* if it can correct a single stable burst deletion of length up to s.

Throughout this paper, when $n = rc$ for appropriate integers r and c, a permutation $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n)$ is also written as the array

$$\begin{bmatrix} \sigma_1 & \sigma_{r+1} & \cdots & \sigma_{(j-1)r+1} & \cdots & \sigma_{(c-1)r+1} \\ \sigma_2 & \sigma_{r+2} & \cdots & \sigma_{(j-1)r+2} & \cdots & \sigma_{(c-1)r+2} \\ \vdots & \vdots & \cdots & \vdots & \ddots & \vdots \\ \sigma_r & \sigma_{2r} & \cdots & \sigma_{jr} & \cdots & \sigma_n \end{bmatrix}.$$

Hence, we may speak of an $r \times c$ permutation, and refer to rows and columns of an $r \times c$ permutation. The ith row of an $r \times c$ permutation σ is denoted by

$$\sigma_{(r,i)} = (\sigma_i, \sigma_{r+i}, \ldots, \sigma_{n-r+i}), \text{ and}$$

the jth column of σ is denoted by

$$\sigma^{(r,j)} = (\sigma_{(j-1)r+1}, \sigma_{(j-1)r+2}, \ldots, \sigma_{jr}).$$

Observe that a stable burst deletion of length s in an $s \times t$ permutation σ is a deletion of exactly one deletion from each row σ and the deletions span at most two adjacent columns of σ.

For two vectors $\mathbf{u} = (u_1, u_2, \ldots, u_m)$ and $\mathbf{v} = (v_1, v_2, \ldots, v_n)$, the *concatenation* of $\mathbf{u}$ and $\mathbf{v}$ is the vector $\mathbf{u}||\mathbf{v} = (u_1, u_2, \ldots, u_m, v_1, v_2, \ldots . v_n)$.

B. 1-SD Permutation Codes

For a positive integer $a \in \mathbb{Z}_{n+1}$, let

$$\mathcal{C}_a(n) = \left\{ \mathbf{u} \in \{0,1\}^n : \sum_{i=1}^{n} i u_i \equiv a \bmod (n+1) \right\},$$

where u_i is the ith component of $\mathbf{u}$. Then $\mathcal{C}_a(n)$ are the family of binary codes known as the *Varshamov-Tenengolts (VT) codes* [10]. These codes are capable of correcting a single deletion. It is known that the choice $a = 0$ maximizes the cardinality of the codes.

As in [14], the *signature* of $\mathbf{u} = (u_1, u_2, \ldots, u_n)$ is the binary vector $\alpha(\mathbf{u}) = (\alpha(u_1), \ldots, \alpha(u_{n-1}))$ of length $n - 1$, where $\alpha(u_i) = 1$ if $u_{i+1} \geq u_i$, and 0 otherwise, for all $i \in [1, n-1]$. For example, if $\mathbf{u} = (1, 2, 1, 3, 2, 3)$, then $\alpha(\mathbf{u}) = (1, 0, 1, 0, 1)$. Levenshteĭn [11] showed that

$$\mathcal{P}_a(n) = \{\sigma \in \mathcal{S}_n : \alpha(\sigma) \in \mathcal{C}_a(n-1)\}$$

is a 1-SD permutation code. Since the union of these codes over $a \in \mathbb{Z}_n$ is exactly $\mathcal{S}_n$, one of them has size at least $n!/n = (n-1)!$.

Tenengolts [14] also generalized the binary VT codes to nonbinary ones. For any $a \in \mathbb{Z}_n$ and $b \in \mathbb{Z}_q$, let

$$\mathcal{C}_{a,b}(n; q) = \left\{ \mathbf{u} \in [1, q]^n : \alpha(\mathbf{u}) \in \mathcal{C}_a(n-1) \text{ and } \sum_{i=1}^{n} u_i \equiv b \pmod{q} \right\}.$$

The codes $\mathcal{C}_{a,b}(n; q)$ are capable of correcting a single deletion, and and one of them has size at least q^{n-1}/n.

C. Upper Bound

Let $A_{\mathsf{SBD}}(n, s)$ denote the maximum size of an s-SBD permutation code in $\mathcal{S}_n$.

For a permutation σ in $\mathcal{S}_n$, let $\mathcal{D}_s(\sigma)$ be the set of all vectors of length $n - s$ received as a result of a stable burst deletion of length s. Let $\mathcal{D}_s(\mathcal{S}_n) = \bigcup_{\sigma \in \mathcal{S}_n} \mathcal{D}_s(\sigma)$.

A code $\mathcal{C}$ is an s-SBD permutation code if and only if for distinct $\sigma, \sigma' \in \mathcal{C}$, we have $\mathcal{D}_s(\sigma) \cap \mathcal{D}_s(\sigma') = \emptyset$. It is easy to see that for each permutation σ, we have $|\mathcal{D}_s(\sigma)| = n - s + 1$. We also have $|\mathcal{D}_s(\mathcal{S}_n)| = n!/s!$, since $\mathcal{D}_s(\mathcal{S}_n)$ is the set of all sequences consisting of $n - s$ distinct symbols from an alphabet of size n. Consequently, we have the following.

Theorem 1. *Let $n > s$ be positive integers. Then*

$$A_{\mathsf{SBD}}(n, s) \leq \frac{n!}{s!(n-s+1)}.$$

III. s-SBD PERMUTATION CODES

We first review the work of Schoeny *et al.* [13] on binary codes that can correct a single burst of length s. In their construction, a codeword $\mathbf{u}$ of length n is treated as an $s \times (n/s)$ array, which is transmitted column-by-column. Thus, a burst deletion of length s in a codeword deletes exactly one bit from each row. Moreover, a deletion can span at most two columns. Therefore, information about the position of a deletion in a single row provides information about the positions of the deletions in the remaining rows. Schoeny *et al.* took advantage of this correlation to construct binary codes that can correct a single burst deletion. More precisely, their construction is as follows:

- The first row is a codeword in a VT code in which the length of the longest run is restricted to be at most L.
- Each of the remaining rows is a codeword in a modified version of the VT code, called *shifted VT code*, which is able to correct a single deletion in each row once the position of the deletion is known to be within $L + 1$ consecutive positions.

The authors took $L = \log(n/s) + 1$ to obtain a class of codes with low redundancy. One may follow their approach to construct s-SBD permutation codes as follows. Take the first row of an $s \times (n/s)$ permutation to be from a nonbinary VT code (so we only take the codewords that have different components). Then the position of the deletion in the first row can be determined exactly. Thus deletions in the remaining $s-1$ rows are known to within two consecutive positions. To recover these deletions, one can use the sum $\sum_{u_j \in \mathbf{u}_{(s,i)}} u_j \equiv b_i \pmod{n}$ to find the symbols and then use the shifted VT codes to determine their positions. However, this incurs huge redundancy when n is large. Actually, the size of the codes constructed above is $\Omega(n!/n^{s+1})$ while the upper bound is $O(n!/n)$. In this section, we give a construction which gives codes of size $\Omega(n!/n^2)$.

Since a stable deletion does not change the values of other positions, the decoder always knows which symbols are deleted from the permutation. What the decoder has to do is to determine their positions. In our construction, this is achieved with the help of nonbinary VT codes and a map μ which records the ordering of the symbols in pairs of adjacent columns of permutations.

Definition 3. Let $\mathbf{u} = (u_1, u_2, \ldots, u_m)$ be an integer vector of length m. Let $\pi(\mathbf{u})$ be the permutation in $\mathcal{S}_m$ defined by

$$\pi(\mathbf{u})_i = |\{j : u_j < u_i, 1 \le j \le m\}|.$$

So $\pi(\mathbf{u})$ is a permutation that records the ordering of the symbols in $\mathbf{u}$.

For example, let $\mathbf{u} = (6, 2, 5, 1, 8, 4)$. Then $\pi(\mathbf{u}) = (5, 2, 4, 1, 6, 3)$.

Definition 4. Let $\ell : \mathcal{S}_m \to [1, m!]$ such that $\ell(\sigma)$ is the lexicographic rank of σ in $\mathcal{S}_m$. For a vector $\mathbf{u}$ of length m, let $\mu : \mathbb{Z}^m \to [1, m!]$ defined by

$$\mu(\mathbf{u}) = \ell(\pi(\mathbf{u})).$$

It is clear that $\mu(\mathbf{u})$ records the ordering of the symbols in $\mathbf{u}$. From $\mu(\mathbf{u})$ and the set $\{u_i : 1 \le i \le m\}$, we can determine the sequence $\mathbf{u}$. For example, if $m = 4$ and $\mu(\mathbf{u}) = 2$, then $\pi(\mathbf{u}) = (1, 2, 4, 3)$. Furthermore, if $\{u_1, u_2, u_3, u_4\} = \{1, 3, 4, 5\}$, then $\mathbf{u} = (1, 3, 5, 4)$.

A. Code Construction

We assume that $n = st$, where $s \ge 2$ and t is even.

Construction 1. *Let $a \in \mathbb{Z}_t$, $b \in \mathbb{Z}_n$, and $c, d \in \mathbb{Z}_{(2s)!}$. Let $\mathcal{C}_s(n; a, b, c, d)$ be the set of all $s \times t$ permutations $\sigma \in \mathcal{S}_n$ such that the following holds:*

(i) The first row $\sigma_{(s,1)}$ of σ is a codeword (with different components) of Tenengolts' nonbinary VT-code as defined in subsection II-B, that is,

$$\sigma_{(s,1)} \in \mathcal{C}_{a,b}(t; n).$$

(ii) Pairs of adjacent columns of σ satisfy the conditions
(1) $\sum_{j=1}^{t/2} \mu(\sigma^{(s,2j-1)} || \sigma^{(s,2j)}) \equiv c \bmod (2s)!$, and
(2) $\sum_{j=1}^{t/2-1} \mu(\sigma^{(s,2j)} || \sigma^{(s,2j+1)}) \equiv d \bmod (2s)!$.

Theorem 2. *The permutation code $\mathcal{C}_s(n; a, b, c, d)$ from Construction 1 is an s-SBD permutation code over $\mathcal{S}_n$.*

Proof. We establish the theorem by giving a decoding algorithm for $\mathcal{C}_s(n; a, b, c, d)$ that recovers from a burst of stable deletions of length s.

Suppose a stable burst deletion of length s occurs in a codeword $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n) \in \mathcal{C}_s(n; a, b, c, d)$, giving $\tilde{\sigma} = (\tilde{\sigma}_1, \tilde{\sigma}_2, \ldots, \tilde{\sigma}_{(t-1)s})$. Let $\rho = (\tilde{\sigma}_1, \tilde{\sigma}_{s+1}, \ldots, \tilde{\sigma}_{(t-2)s+1})$, and $\gamma_j = (\tilde{\sigma}_{(j-1)s+1}, \tilde{\sigma}_{(j-1)s+2}, \ldots, \tilde{\sigma}_{js})$, for $1 \le j \le t-1$. So ρ is the first row of the received codeword and γ_j is the jth column. Since the first row of σ is a codeword of a single-deletion correcting code by construction, we can recover the first row of σ from ρ. From the recovered row, we can also determine exactly where the deletion has taken place since all entries of the row are distinct.

Suppose the deletion in the first row of σ occured at position $(j_1 - 1)s + 1$, where $2 \le j_1 \le t-1$. Then we have

(i) $\sigma^{(s,j)} = \gamma_j$ for $1 \le j < j_1 - 1$, and
(ii) $\sigma^{(s,j)} = \gamma_{j-1}$ for $j_1 + 1 \le j \le t$,

since the s deletions are adjacent and occur across at most two adjacent columns, $\sigma^{(s,j_1-1)}$ and $\sigma^{(s,j_1)}$, of σ. There remain two columns $\sigma^{(s,j_1-1)}$ and $\sigma^{(s,j_1)}$ to recover. The symbols appearing in these two columns can be deduced from the other columns as the burst deletion is stable. Knowing the ordering of these symbols in $\sigma^{(s,j_1-1)} || \sigma^{(s,j_1)}$ would enable us to recover $\sigma^{(s,j_1-1)}$ and $\sigma^{(s,j_1)}$. This ordering is precisely encoded in $\mu(\sigma^{(s,j_1-1)} || \sigma^{(s,j_1)})$, which can be computed as follows:

when j_1 is even,

$$\mu(\sigma^{(s,j_1-1)} || \sigma^{(s,j_1)}) \equiv c - \sum_{j \in [1,t/2] \setminus \{j_1/2\}} \mu(\sigma^{(s,2j-1)} || \sigma^{(s,2j)}) \bmod (2s)!; \text{ and}$$

when j_1 is odd,

$$\mu(\sigma^{(s,j_1-1)} || \sigma^{(s,j_1)}) \equiv d - \sum_{j \in [1,t/2-1] \setminus \{(j_1-1)/2\}} \mu(\sigma^{(s,2j)} || \sigma^{(s,2j+1)}) \bmod (2s)!.$$

When $j_1 = 1$, all the s deletions occur in the first column of σ, and can be recovered with knowledge of $\mu(\sigma^{(s,1)} || \sigma^{(s,2)})$. Similarly, when $j_1 = t$, the deletions occur in the last two columns of σ and can be recovered with knowledge of $\mu(\sigma^{(s,t-1)} || \sigma^{(s,t)})$. Both of $\mu(\sigma^{(s,1)} || \sigma^{(s,2)})$ and $\mu(\sigma^{(s,t-1)} || \sigma^{(s,t)})$ can be computed as above, using condition (1). ■

Corollary 1. *There exists an s-SBD permutation code of size at least $\dfrac{s}{((2s)!)^2} \cdot \dfrac{n!}{n^2}$.*

Proof. We have $\cup_{a \in \mathbb{Z}_t, b \in \mathbb{Z}_n, c,d \in \mathbb{Z}_{(2s)!}} \mathcal{C}_s(n; a, b, c, d) = \mathcal{S}_n$, where $t = n/s$. There is a choice of a, b, c, d such that the size of $\mathcal{C}_s(n; a, b, c, d)$ is at least as large as the average. ■

When $s = 2$, we have a better result. Due to space constraints, we defer its proof to the full paper.

Theorem 3. *There exists a 2-SBD permutation code of size at least $n!/n^2$.*

IV. $\leq s$-SBD PERMUTATION CODES

In this section, we consider the problem of correcting a burst of at most s stable deletions. When $s = 2$, the intersection of $\mathcal{P}_a(n)$ and the code from Theorem 3 gives a ≤ 2-SBD permutation code.

Theorem 4. *There exists a ≤ 2-SBD permutation code of size at least $n!/n^3$.*

For general s, one can take the intersection of a family of $\mathcal{C}_i(n; a_i, b_i, c_i, d_i)$, where $2 \leq i \leq s$, together with $\mathcal{P}_{a_1}(n)$ to construct a $\leq s$-SBD permutation code. Such a code has size at least $\frac{s!n!}{\prod_{i=2}^{s}((2i)!)^2 n^{2s-1}}$. Below, we show that we can do a little better. Assume that $i \mid n$ for all $1 \leq i \leq s$, and $n = 2ts$ for some even integer t.

Construction 2. *Let $c, d \in \mathbb{Z}_{(4s)!}$, and let $\mathbf{a} = (a_i)_{i=1}^{s}$ and $\mathbf{b} = (b_j)_{j=2}^{s}$ be two sequences of nonnegative integers such that $a_i \in \mathbb{Z}_{n/i}$, for $i = 1, 2, \ldots, s$, and $b_j \in \mathbb{Z}_n$, for $j = 2, 3, \ldots, s$. Let $\mathcal{D}_s(n; \mathbf{a}, \mathbf{b}, c, d)$ be the set of all permutations $\sigma \in \mathcal{S}_n$ such that the following holds:*

(i) $\sigma \in \mathcal{P}_{a_1}(n)$.

(ii) When σ is viewed as an $i \times (n/i)$ permutation, its first row $\sigma_{(i,1)}$ belongs to $\mathcal{C}_{a_i,b_i}(n/i; n)$, for $i = 2, 3, \ldots, s$.

(iii) When viewed as a $2s \times t$ permutation, pairs of adjacent columns of σ satisfy the conditions

(1) $\sum_{j=1}^{t/2} \mu(\sigma^{(2s,2j-1)} || \sigma^{(2s,2j)}) \equiv c \bmod (4s)!$, and

(2) $\sum_{j=1}^{t/2-1} \mu(\sigma^{(2s,2j)} || \sigma^{(2s,2j+1)}) \equiv d \bmod (4s)!$.

Theorem 5. *The permutation code $\mathcal{D}_s(n; \mathbf{a}, \mathbf{b}, c, d)$ from Construction 2 is a $\leq s$-SBD permutation code.*

Proof. We establish the theorem by giving a decoding algorithm for $\mathcal{D}_s(n; \mathbf{a}, \mathbf{b}, c, d)$ that recovers from a burst of stable deletions of length up to s.

Suppose a stable burst deletion of length $\ell \leq s$ occurs in a codeword $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n) \in \mathcal{D}_s(n; \mathbf{a}, \mathbf{b}, c, d)$, giving $\tilde{\sigma} = (\tilde{\sigma}_1, \tilde{\sigma}_2, \ldots, \tilde{\sigma}_{n-\ell})$. The value of ℓ can be determined from the length of $\tilde{\sigma}$.

If $\ell = 1$, we can recover σ since $\sigma \in \mathcal{P}_{a_1}(n)$.

If $\ell \geq 2$, we view σ as an $\ell \times (n/\ell)$ permutation and recover the first row of σ and the location of the deletion as in the proof of Theorem 2. Suppose $\sigma_{j_1\ell+1}$ (in the first row of σ) is deleted.

For $j_1 \neq 0$, since the ℓ deletions are adjacent, we necessarily have

(i) $\sigma_i = \tilde{\sigma}_i$, for $1 \leq i \leq j_1\ell - \ell + 1$; and

(ii) $\sigma_i = \tilde{\sigma}_{i-\ell}$, for $j_1\ell + \ell + 1 \leq i \leq n$.

It remains to determine the sequence $\sigma_{(j_1-1)\ell+2}$, $\sigma_{(j_1-1)\ell+2}$, $\ldots, \sigma_{(j_1+1)\ell}$ of length $2\ell - 1$. To do this, view σ as a $2s \times (n/2s)$ permutation. Then the sequence to be determined spans at most two adjacent columns of σ and can be recovered as in the proof of Theorem 2.

For $j_1 = 0$, the deletions occur in the first ℓ positions. So the sequence to be determined spans the first column of the $2s \times (n/2s)$ permutation σ and still can be recovered as in the proof of Theorem 2. ■

Corollary 2. *There exists a $\leq s$-SBD permutation code of size at least $\frac{s!n!}{((4s)!)^2 n^{2s-1}}$.*

V. CORRECTING MULTIPLE BURST DELETIONS OF LENGTHS UP TO s

In this section, we consider the problem of correcting multiple stable burst deletions, each of length up to s. Our approach is based on the permutation interleaving technique. For a permutation code $\mathcal{C}$, let us denote its minimum Hamming distance by $d_H(\mathcal{C})$. The maximum size of a permutation code $\mathcal{C} \subseteq \mathcal{S}_n$ with $d_H(\mathcal{C}) \geqslant d$ is denoted by $A(n, d)$.

Construction 3. *Let n and s be positive integers. Assume that n is a multiple of $s+1$, say $n = (s+1)t$. Let b be a positive integer less than t. For $i = 1, 2, \ldots, s+1$, let $\mathcal{E}_i$ be a permutation code on the set $\{i, (s+1)+i, 2(s+1)+i, (t-1)(s+1)+i\}$ with $d_H(\mathcal{E}_i) \geq b+1$, and let $\mathcal{C}(n)$ be the set of all $(s+1) \times t$ permutations $\sigma \in \mathcal{S}_n$ such that the i-th row of σ is a codeword in $\mathcal{E}_i$, that is,*

$$\mathcal{C}(n) = \{\sigma \in \mathcal{S}_n : \sigma_{(s+1,i)} \in \mathcal{E}_i,\ i = 1, 2, \ldots, s+1\}.$$

Theorem 6. *The code $\mathcal{C}(n)$ in Construction 3 is a permutation code over $\mathcal{S}_n$ which can correct up to b stable burst deletions, each of length up to s. The size of this code is*

$$|\mathcal{C}(n)| = A\left(\frac{n}{s+1}, b+1\right)^{s+1}.$$

Proof. Suppose a codeword $\sigma = (\sigma_1, \sigma_2, \ldots, \sigma_n) \in \mathcal{C}(n)$ suffers up to b stable burst deletions, each of length at most s, resulting in $\tilde{\sigma} = (\tilde{\sigma}_1, \tilde{\sigma}_2, \ldots, \tilde{\sigma}_m)$, where $n - bs \leq m \leq n$.

Viewing σ as an $(s+1) \times t$ permutation, observe that any pair of symbols from adjacent rows of σ differs by exactly one modulo $(s+1)$. Hence, $\sigma_i - \sigma_{i-1} \equiv 1 \bmod (s+1)$ for all $i \in [1, n]$ (with the convention that $\sigma_0 = s+1$. We can use this observation and the fact that each of the burst deletions have length at most s to determine the positions and lengths of the burst deletions as follows. Let $\tilde{\sigma}_0 = s+1$. For each $i \in [1, m]$, if $\tilde{\sigma}_i - \tilde{\sigma}_{i-1} \equiv 1 \bmod (s+1)$, then there is no burst deletion at position i of σ. Otherwise, there is a burst deletion at position i of σ, whose length is given by $s_i \equiv \tilde{\sigma}_i - \tilde{\sigma}_{i-1} - 1 \bmod (s+1)$.

Now, for each position i of a burst deletion, the number of symbols that have been deleted up to and including the burst deletion at position i is $\sum_{k=1}^{i} s_k$. Hence, we have

$$\sigma_{i+\sum_{k=1}^{i} s_k} = \tilde{\sigma}_i,$$

since the position immediately after a burst deletion cannot be the position of another burst deletion. This allows us to determine σ partially from $\tilde{\sigma}$. If we use "*" to denote every undetermined symbol in the $(s+1) \times t$ permutation σ, then there are at most b "*"s in each row, since a burst deletion of length at most s cannot result in two deletions from the same row. We can therefore regard each row of σ as having undergone at most b (stable) erasures. Gabrys *et al.* [6] have shown that a permutation code can correct b stable erasures if and only if its Hamming distance is greater than b. By construction, the i-th row of every permutation in $\mathcal{C}(n)$ forms $\mathcal{E}_i$, which is a permutation code with $d_H(\mathcal{E}_i) \geq b+1$. Therefore, we can correct all the erasures in each row of σ. ■

For a permutation code $\mathcal{C} \subseteq \mathcal{S}_n$, its *rate* is defined as

$$R(\mathcal{C}) = \lim_{n\to\infty} \frac{\log|\mathcal{C}|}{\log n!}.$$

Farnoud *et al.* [5] showed that

$$\lim_{n\to\infty} \frac{\log A(n,d)}{\log n!} = 1 - \lim_{n\to\infty} \frac{d}{n}.$$

Hence, we have the following result.

Corollary 3. *Let $b = \delta n$, for $1 < \delta < 1$, and let s be fixed. Then the code $\mathcal{C}(n)$ in Construction 3 has asymptotic rate*

$$R(\mathcal{C}(n)) = 1 - \delta(s+1).$$

We now compare the rate of $\mathcal{C}(n)$ with codes capable of correcting up to bs deletions. It is known [6] that a permutation code can correct up to bs deletions if and only if the code has minimum Ulam distance $bs+1$. Farnoud *et al.* [5] provided a construction for permutation codes with minimum Ulam distance $bs+1$ which were enlarged by Chee and Vu in [2]. In general, although the code in [2] has larger size than the code in [5], they have the same asymptotic rate of

$$R = 1 - 2^{-\lfloor 2/3\delta s \rfloor} - \frac{3\delta s}{2}\left\lfloor \log\frac{2}{3\delta s}\right\rfloor,$$

when $b = \delta n$. It follows that when δs is small, our codes have higher rate.

VI. Conclusion

In this paper, we initiate the investigation into permutation codes that are capable of correcting bursts of stable deletions. Codes in $\mathcal{S}_n$ capable of correcting a single burst of exactly s stable deletions have size upper bounded by $O(n!/n)$. We give a construction for a family of such codes whose size is $\Omega(n!/n^2)$. Our construction is also extended to deal with the cases of single burst of at most s stable deletions, and up to b bursts of at most s stable deletions each.

References

[1] A. Barg and A. Mazumdar. Codes in permutations and error correction for rank modulation. *IEEE Trans. Inform. Theory*, 56(7):3158–3165, 2010.

[2] Y. M. Chee and V. K. Vu. Breakpoint analysis and permutation codes in generalized Kendall tau and Cayley metrics'. In *ISIT 2016 – Proceedings of the 2014 IEEE International Symposium on Information Theory*, pages 2959–2963, 2014.

[3] Y. M. Chee, V. K. Vu, and X. Zhang. Permutation codes correcting a single burst deletion I: unstable deletions. In *ISIT 2015 – Proceedings of the 2015 IEEE International Symposium on Information Theory*, pages 1741–1745, 2015.

[4] B. Eitan and A. Roy. Binary and multilevel flash cells. In *Flash Memories*, pages 91–152. Springer, 1999.

[5] F. Farnoud, V. Skachek, and O. Milenkovic. Error-correction in flash memories via codes in the Ulam metric. *IEEE Trans. Inform. Theory*, 59(5):3003–3020, 2013.

[6] R. Gabrys, E. Yaakobi, F. Farnoud, F. Sala, J. Bruck, and L. Dolecek. Codes correcting erasures and deletions for rank modulation. *IEEE Trans. Inform. Theory*, 62(1):136–150, 2016.

[7] A. Jiang, R. Mateescu, M. Schwartz, and J. Bruck. Rank modulation for flash memories. *IEEE Trans. Inform. Theory*, 55(6):2659–2673, 2009.

[8] A. Jiang, M. Schwartz, and J. Bruck. Correcting charge-constrained errors in the rank-modulation scheme. *IEEE Trans. Inform. Theory*, 56(5):2112–2120, 2010.

[9] J.-D. Lee, S.-H. Hur, and J.-D. Choi. Effects of floating-gate interference on nand flash memory cell operation. *IEEE Electron. Device Lett.*, 23(5):264–266, 2002.

[10] V. I. Levenshteĭn. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Dokl.*, 10:707–710, 1965.

[11] V. I. Levenshteĭn. On perfect codes in deletion and insertion metric. *Discrete Math. Appl.*, 3:241–258, 1992.

[12] K. Prall. Scaling non-volatile memory below 30nm. In *2007 22nd IEEE Non-Volatile Semiconductor Memory Workshop*, 2007.

[13] C. Schoeny, A. Wachter, R. Gabrys, and E. Yaakobi. Codes correcting a burst of deletions or insertions. In *ISIT 2016 – Proceedings of the 2016 IEEE International Symposium on Information Theory*, pages 630–634, 2016.

[14] G. Tenengolts. Nonbinary codes, correcting single deletion or insertion. *IEEE Trans. Inform. Theory*, 30(5):766–769, 1984.

[15] H. Zhou, M. Schwartz, A. Jiang, and J. Bruck. Systematic error-correcting codes for rank modulation. *IEEE Trans. Inform. Theory*, 61(1):17–32, 2015.