# Systematic Error-Correcting Codes for Rank Modulation

Hongchao Zhou, Moshe Schwartz, *Senior Member, IEEE*, Anxiao (Andrew) Jiang, *Senior Member, IEEE*, and Jehoshua Bruck, *Fellow, IEEE*

***Abstract*— The rank-modulation scheme has been recently proposed for efficiently storing data in nonvolatile memories. In this paper, we explore $[n, k, d]$ systematic error-correcting codes for rank modulation. Such codes have length $n$, $k$ information symbols, and minimum distance $d$. Systematic codes have the benefits of enabling efficient information retrieval in conjunction with memory-scrubbing schemes. We study systematic codes for rank modulation under Kendall's $\tau$-metric as well as under the $\ell_\infty$-metric. In Kendall's $\tau$-metric, we present $[k+2, k, 3]$ systematic codes for correcting a single error, which have optimal rates, unless systematic perfect codes exist. We also study the design of multierror-correcting codes, and provide a construction of $[k+t+1, k, 2t+1]$ systematic codes, for large-enough $k$. We use nonconstructive arguments to show that for rank modulation, systematic codes achieve the same capacity as general error-correcting codes. Finally, in the $\ell_\infty$-metric, we construct two $[n, k, d]$ systematic multierror-correcting codes, the first for the case of $d = O(1)$ and the second for $d = \Theta(n)$. In the latter case, the codes have the same asymptotic rate as the best codes currently known in this metric.**

***Index Terms*— Flash memory, rank modulation, error-correcting codes, permutations, metric embeddings, Kendall's $\tau$-metric, $\ell_\infty$-metric, systematic codes.**

## I. Introduction

THE rank-modulation scheme has been recently proposed for storing data efficiently and robustly in nonvolatile memories (NVMs) [12]. Its applications include flash memories [5], which are currently the most widely used family of NVMs, and several emerging NVM technologies, such as phase-change memories [3]. The rank-modulation scheme uses the relative order of cell levels to represent data, where a cell level denotes a floating-gate cell's threshold voltage for flash memories and denotes a cell's electrical resistance for resistive memories (such as phase-change memories). Consider $n$ memory cells, where for $i \in [n] = \{1, 2, \ldots, n\}$, let $c_i \in \mathbb{R}$ denote the level of the $i$th cell. It is assumed that no two cells have the exact same level, which is easy to realize in practice. Let $S_n$ denote the set of all $n!$ permutations over $[n]$. The $n$ cell levels induce a permutation $[f_1, f_2, \ldots, f_n] \in S_n$, where $c_{f_1} > c_{f_2} > \cdots > c_{f_n}$. The rank-modulation scheme uses such permutations to represent data. It enables memory cells to be programmed efficiently and robustly, from lower levels to higher levels, without the risk of over-programming. It also makes it easier to adjust cell levels when noise appears without erasing cells, and makes the stored data more robust to asymmetric errors that change cell levels in the same direction [12], [13], [29].

Error-correcting codes are essential for data reliability. An error-correcting code is a set of elements in a metric space, no two of which are too close together under its distance measure. In the case of rank modulation, the space is $S_n$. As for the distance measure, it is usually chosen in such a way that small (common) errors in the physical medium correspond to a small distance in the metric space. In the context of rank modulation for NVMs, the two most studied distance functions are Kendall's $\tau$-distance, and the $\ell_\infty$-distance. It was suggested in [13] that small charge-constrained errors correspond to a small distance in Kendall's $\tau$-metric. In contrast, in [29] it was shown that small limited-magnitude errors correspond to a small $\ell_\infty$-distance.

Some results are known on error-correcting codes for rank modulation equipped with Kendall's $\tau$-distance. In [13], a single-error-correcting code is constructed based on metric embedding, whose size is provably within half of the optimal size. In [2], the capacity of rank modulation codes is derived for the full range of minimum distance between codewords, and the existence of codes whose sizes are within a constant factor of the sphere-packing bound for any fixed number of errors is shown. Some explicit constructions of error-correcting codes have been proposed and analyzed in [23]. We also mention that the Ulam metric has been suggested as a generalization of Kendall's $\tau$-metric and was recently studied in the context of error-correcting codes in [7].

There has also been some work on error-correcting codes for rank modulation equipped with the $\ell_\infty$-distance. In [17] and [29] some general constructions and bounds were given. A relabeling scheme, improving the distance

Manuscript received October 25, 2013; revised August 14, 2014; accepted October 12, 2014. Date of publication October 28, 2014; date of current version December 22, 2014. This work was supported in part by the U.S.-Israel Binational Science Foundation under Grant 2010075, in part by the National Science Foundation (NSF) under Grant CCF-1218005, in part by the NSF CAREER under Award CCF-0747415, and in part by the NSF under Grant CCF-1217944. This paper was presented at the 2012 IEEE International Symposium on Information Theory.

H. Zhou is with the Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: hongchao@mit.edu).

M. Schwartz is with the Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Beersheba 8410501, Israel (e-mail: schwartz@ee.bgu.ac.il).

A. Jiang is with the Department of Computer Science and Engineering, Texas A&M University, College Station, TX 77843 USA (e-mail: ajiang@cse.tamu.edu).

J. Bruck is with the Department of Electrical Engineering, California Institute of Technology, Pasadena, CA 91125 USA (e-mail: bruck@paradise.caltech.edu).

Communicated by B. S. Rajan, Associate Editor for Coding Theory.

Digital Object Identifier 10.1109/TIT.2014.2365499

TABLE I
COMPARISON OF LENGTH-$n$ CODES IN KENDALL'S $\tau$-METRIC

| Source | # of Errors | Size | Asympt. Size ($t$ constant) | Properties | Restrictions |
|---|---|---|---|---|---|
| [13] | 1 | $\geqslant \frac{n!}{2n-1}$ | $\Omega(n!/n)$ | E | |
| [2] | $t$ | $\geqslant \frac{n!}{f_t(n)}$ | $\Omega(n!/n^t)$ | E | $n-2$ is a power of a prime |
| [23] | $t$ | $\geqslant \frac{2^m}{(m+1)^t}$ | $\Omega(n!/(n^t \log^t n))$ | C | |
| Const. A | 1 | $\frac{n!}{n(n-1)}$ | $\Omega(n!/n^2)$ | SC | $n-1$ or $n-2$ is prime |
| Const. B | 1 | $\frac{n!}{n(n-1)}$ | $\Omega(n!/n^2)$ | SC | |
| Const. C | $t$ | $\frac{n!}{n(n-1)\ldots(n-2t)}$ | $\Omega(n!/n^{2t+1})$ | SC | See Const. C. |
| Th. 19 | $t$ | $\frac{n!}{n(n-1)\ldots(n-2t)}$ | $\Omega(n!/n^{2t+1})$ | SE | |
| Const. D & Th. 14 | $t$ | $\frac{n!}{n(n-1)\ldots(n-t)}$ | $\Omega(n!/n^{t+1})$ | SC | large-enough $n$ |

- Properties: E – Existential, C – Constructive, S – Systematic
- $f_t(n) = t(t+2-(t \bmod 2))\frac{(n-2)^{t+1}-1}{n-3}$
- $m = (n+1)\lfloor \log_2 n \rfloor - 2^{\lfloor \log_2 n \rfloor + 1} + 2$

of codes was suggested in [30]. Several counting problems, mainly concerning ball size under the $\ell_\infty$-metric, and optimal anticodes, were studied in [15], [16], [26], and [27].

In this paper, we study *systematic* error-correcting codes for rank modulation as a new approach for code design. In the more common error-correcting setting over vectors equipped with the Hamming distance function, an $[n, k, d]$ systematic code is a subset of length-$n$ vectors whose projection onto a given set of $k$ coordinates has all possible length-$k$ vectors appearing exactly once. Additionally, distinct codewords are at least distance $d$ apart. These $k$ positions are referred to as the *information symbols*, whereas the rest of the positions are called *redundancy symbols*. If the code is linear, it is well known (for example, see [21]) that any code has an equivalent code with the same parameters that is also systematic.

We shall be interested in the analog of systematic codes in the space of permutations with either Kendall's $\tau$-distance or the $\ell_\infty$-distance. Loosely speaking, in an $[n, k, d]$ systematic code (either in Kendall's $\tau$-metric or the $\ell_\infty$-metric), when projecting the $n$-permutation codewords onto the $k$ information symbols, each possible $k$-permutation appears exactly once. Additionally, there is a minimum-distance guarantee of $d$ between distinct codewords, which allows correction of up to $t$ errors, where $t = \lfloor (d-1)/2 \rfloor$. A more rigorous definition will follow in the next section.

Systematic codes for rank modulation are mainly motivated by *memory-scrubbing* applications. In such schemes, when reading information from the memory, the users assume no errors are present. To make certain this assumption holds, an independent background process periodically reads information from the memory and rewrites it with corrections if needed (see for example [8, p. 578]). Memory scrubbing is common in DRAM, and has also been studied for conventional flash memory [1], [11].

In the context of a memory-scrubbing scheme with rank modulation, since every permutation induced by the information symbols appears in exactly one codeword, and since we assume no noise when reading, the information symbols of codewords can be mapped efficiently to data. Thus, no decoding is required, and the mapping of permutations to data by the readers may be done via enumerative source coding (e.g., by ordering permutations lexicographically) in linear time [6], [22]. In contrast, the independent memory-scrubbing process does employ a decoding procedure.

The main contributions of this work are the design of systematic codes, and the analysis of their performance. In Kendall's $\tau$-metric we present families of $[k+2,\ k, 3]$ systematic codes for correcting a single error. We show that they have optimal parameters among systematic codes, unless *perfect* systematic single-error-correcting codes, which meet the sphere-packing bound, exist. We also study the design of systematic codes that correct multiple errors, and provide constructions for a wide range of parameters. In particular, we show a construction for $[k+t+1, k, 2t+1]$ systematic codes capable of correcting $t$ errors. Furthermore, we prove that systematic codes have the same capacity as general error-correcting codes. This result establishes that, asymptotically, systematic codes are as strong in their error-correction capability as general codes. The systematic codes we present in this work have the same encoding and decoding complexity as the non-systematic codes presented in [23], but they incur a rate penalty. The main constructions we present for Kendall's $\tau$-metric are summarized in Table I, and compared with previously-known results. The asymptotic code size is given assuming the number of correctable errors is a constant $t$. To facilitate the comparison with previous works, Table I parametrizes codes by their length $n$, whereas in the rest of the paper we use the number of information symbols, $k$, as the main code parameter.

We also consider the $\ell_\infty$-metric, and provide two constructions for systematic codes. The first construction is for $[n, k, d]$ systematic codes with $d = O(1)$, and the second is for the case of $d = \Theta(n)$. We show that the asymptotic rate of the second construction equals that of the best codes currently known. The main constructions we present for the $\ell_\infty$-metric are summarized in Table II, and compared with previously-known results.

The rest of the paper is organized as follows. In Section II we provide the basic notation and definitions used throughout the paper. In Section III we study systematic codes in Kendall's $\tau$-metric. We turn in Section IV to explore

TABLE II
COMPARISON OF CODES IN THE $\ell_\infty$-METRIC

| Source | Length | Min. Dist. | Size | Properties |
|---|---|---|---|---|
| [29] | $n$ | $d$ | $(\lceil n/d \rceil !)^{n \bmod d} (\lfloor n/d \rfloor !)^{d-(n \bmod d)}$ | C |
| Const. E | $n$ | $d$ | $\lceil n/d \rceil !$ | SC |
| Const. F | $n+k$ | $d$ | $\max_k \left\{ k! \;\middle|\; k! \leqslant (\lceil n/d \rceil !)^{n \bmod d} (\lfloor n/d \rfloor !)^{d-(n \bmod d)} \right\}$ | SC |

• Properties: C – Constructive, S – Systematic

systematic codes in the $\ell_\infty$-metric. We conclude in Section V and present some open problems.

## II. NOTATION AND DEFINITIONS

Let $[n] = \{1, 2, \ldots, n\}$, and $S_n$ denote the set of permutations over $[n]$. A permutation $f \in S_n$ is represented in single-line notation by $f = [f_1, f_2, \ldots, f_n]$, where $f(i) = f_i$. We also denote the identity permutation by $\mathrm{Id} = [1, 2, \ldots, n]$. Finally, we denote by $f^{-1}$ the inverse of the permutation $f$, i.e., the permutation sending $f(i)$ to $i$.

Consider a metric over the permutations $S_n$ with a distance function $d : S_n \times S_n \to \mathbb{N} \cup \{0\}$. An $(n, M, d_{\min})$-code is a subset $C \subseteq S_n$ such that $|C| = M$, and $d(f, g) \geqslant d_{\min}$ for all $f, g \in C$, $f \neq g$. We say $M$ is the *size* of the code, and $d_{\min}$ is the minimum distance of the code.

In this work we shall consider two distance functions: Kendall's $\tau$-distance, and the $\ell_\infty$-distance. The latter $\ell_\infty$-distance function is easily defined for all $f, g \in S_n$ by

$$d_\infty(f, g) = \max \{|f(i) - g(i)| \mid i \in [n]\}.$$

For the former, Kendall's $\tau$-distance function, assume $f \in S_n$ is some permutation. An *adjacent transposition* on $f$ switches the values of $f(i)$ and $f(i+1)$ for some $i \in [n-1]$. Kendall's $\tau$-distance [14] between $f$ and $g$, denoted by $d_K(f, g)$, is defined as the minimal number of adjacent transpositions required to transform $f$ into $g$. This is sometimes also called the *bubble-sort distance*.

We recall that in the rank-modulation scheme we have $n$ memory cells labeled by $[n]$, and the level of the $i$th cell is denoted by $c_i \in \mathbb{R}$. Assume $c_{i_1} > c_{i_2} > \cdots > c_{i_n}$, then the permutation stored by the $n$ cells is $[i_1, i_2, \ldots, i_n] \in S_n$ (see [12]). Assume a permutation $f \in S_n$ was stored, but a distorted version of it, $g \in S_n$, was eventually read. It was noted in [13] that small charge-constrained errors translate to small Kendall's $\tau$-distance. In contrast, it was suggested in [29] and [30], that small limited-magnitude errors translate to small $\ell_\infty$-distance on the inverse permutation. This difference between storing the permutation or its inverse will play a role in defining two versions of systematic codes.

In order to define systematic codes we need to define two types of projections. Let $A = \{a_1, a_2, \ldots, a_m\} \subseteq [n]$ be any subset, $a_1 < a_2 < \cdots < a_m$. For any permutation $f \in S_n$, we define $f|_A$ to be the permutation in $S_m$ that preserves the relative order of the sequence $f(a_1), f(a_2), \ldots, f(a_m)$. Intuitively, to compute $f|_A$ we keep only the *coordinates* of $f$ that appear in $A$, and then relabel the entries to $[m]$ while keeping the relative order. In a similar fashion we define

$$f|^A = \left(f^{-1}|_A\right)^{-1}.$$

To calculate $f|^A$ we keep only the *values* of $f$ from $A$, and then relabel the entries to $[m]$ while keeping relative order.

*Example 1: Let $n = 6$ and consider the permutation*

$$f = [6, 1, 3, 5, 2, 4] \in S_6.$$

*We take $A = \{3, 5, 6\}$. We then have*

$$f|_A = [2, 1, 3],$$

*since we keep positions* 3, 5, *and* 6, *of $f$, giving us* $[3, 2, 4]$, *and then relabel these to get* $[2, 1, 3]$.

*Similarly, we have*

$$f|^A = [3, 1, 2],$$

*since we keep the values* 3, 5, *and* 6, *of $f$, giving us* $[6, 3, 5]$, *and then relabel these to get* $[3, 1, 2]$. □

We are now in a position to define systematic codes in two different ways, depending on the metric.

*Definition 2: An $[n, k, d]$ systematic code, $C$, for Kendall's $\tau$-metric, is an $(n, k!, d)$ code such that*

$$\left\{ f|^{[k]} \;\middle|\; f \in C \right\} = S_k.$$

*We call $[k]$ the* information symbols *of the code, and $\{k+1, k+2, \ldots, n\}$ the* redundancy symbols *of the code. The* redundancy *of the code is defined as the number of redundancy symbols, i.e., $n - k$.*

The notation $[n, k, d]$ uses square brackets, to distinguish from the $(n, M, d)$ notation of [2], [13], [29], and [30] in which $M$ denotes the number of codewords. We briefly comment that the same notation, in the context of codes over vector spaces, is used for linear codes. There, also, a linear $[n, k, d]$ systematic code has $k$ information symbols and redundancy $n - k$. However, the codes in this paper are over permutations, and are, by no means, linear.

If we have an $[n, k, d]$ systematic code in Kendall's $\tau$-metric, reading just the levels of the first $k$ cells and comparing them, enables us to ascertain the relative positions of the values $1, 2, \ldots, k$ in the stored permutation, and there is a unique codeword with this relative ordering. More precisely, assume a codeword $f \in C$ is stored. If the levels we read from the first $k$ cells are $c_1, c_2, \ldots, c_k$, and their ordering is $c_{i_1} > c_{i_2} > \cdots > c_{i_k}$, then $i_1$ appears before $i_2$ in the codeword, appearing before $i_3$, and so on, until $i_k$, i.e., $f^{-1}(i_1) < f^{-1}(i_2) < \cdots < f^{-1}(i_k)$.

In contrast, in the setting of limited-magnitude errors and the $\ell_\infty$-metric [29], the *inverse* of the permutation read from the cells is protected by an error-correcting code. Thus, if $g$ is the codeword we want to store, we would physically write its inverse $g^{-1}$ to the cells using the rank-modulation scheme. Then, reading just the levels of the first $k$ cells, $c_1, c_2, \ldots, c_k$, gives us the relative ordering of $g(1), g(2), \ldots, g(k)$. This motivates the following definition.

*Definition 3: An $[n, k, d]$ systematic code, $C$, for the $\ell_\infty$-metric, is an $(n, k!, d)$ code such that*

$$\left\{ g|_{[k]} \;\middle|\; g \in C \right\} = S_k.$$

*We call $[k]$ the* information coordinates *of the code, and $\{k+1, k+2, \ldots, n\}$ the* redundancy coordinates *of the code.*

## III. Systematic Codes in Kendall's $\tau$-Metric

This section is devoted to the study of systematic codes in Kendall's $\tau$-metric. In Section III-A we introduce further notation and some useful lemmas. In Section III-B we study systematic single-error-correcting codes. We turn, in Section III-C, to the case of general systematic error-correcting codes. Finally, in Section III-D, we analyze the capacity of systematic codes.

### A. Preliminaries

We let $\mathbb{Z}_n$ denote the set of integers $\{0, 1, \ldots, n-1\}$, as well as the additive group over these integers with addition modulo $n$. It is well known (see [13], and references therein) that there is a one-to-one correspondence between the permutations of $S_n$ and *factoradic* representations, which are mixed-radix vectors from

$$\mathbb{Z}_n! = \mathbb{Z}_1 \times \mathbb{Z}_2 \times \cdots \times \mathbb{Z}_{n-1} \times \mathbb{Z}_n.$$

Let $f \in S_n$ be any permutation. The factoradic representation corresponding to $f$ is a vector $v = [v_1, \ldots, v_n] \in Z_n!$ such that $v_i \in \mathbb{Z}_i$ equals

$$v_i = \left| \left\{ j \;\middle|\; j < i \text{ and } f^{-1}(j) > f^{-1}(i) \right\} \right|,$$

i.e., $v_i$ counts the number of elements of lesser value than $i$, but which appear to the right of $i$ in the permutation $f = [f_1, f_2, \ldots, f_n]$. We note that $v_1 = 0$ always, and is thus redundant in the representation, but we keep it to make the notation simpler. From now on, we denote the factoradic representation of $f \in S_n$ by $\Phi(f) \in \mathbb{Z}_n!$, and the $i$th element of $\Phi(f)$ by $\Phi(f)_i$.

We now crucially observe that, in a systematic scheme, setting the levels of the first $k$ cells determines exactly the first $k$ entries of the factoradic representation of the permutation stored by the $n$ cells. This is true regardless of the levels of the last $n-k$ cells. More succinctly, for any $f \in S_n$, and for all $1 \leqslant i \leqslant k \leqslant n$,

$$\Phi(f|^{[k]})_i = \Phi(f)_i.$$

*Example 4: Let $n = 6$ and $k = 4$. Take*

$$f = [6, 1, 3, 2, 5, 4] \in S_n.$$

*We then have*

$$\Phi(f) = [0, 0, 1, 0, 1, 5],$$

*as well as*

$$f|^{[k]} = [1, 3, 2, 4] \quad \text{and} \quad \Phi(f|^{[k]}) = [0, 0, 1, 0].$$

*We observe that the first $k$ coordinates of $\Phi(f)$ and $\Phi(f|^{[k]})$ are the same.* □

Another well-known fact (used by [2] and [13]) is the following metric embedding:

$$d_K(f, g) \geqslant d_1(\Phi(f), \Phi(g)) = \sum_{i=1}^{n} |\Phi(f)_i - \Phi(g)_i|, \quad (1)$$

where $d_K$ is Kendall's $\tau$-distance, and $d_1$ is the $\ell_1$-distance. The following lemma gives a more refined version of (1), taking into account the partition into information symbols and redundancy symbols.

*Lemma 5: Given $f, g \in S_n$, and $1 \leqslant k \leqslant n$,*

$$d_K(f, g) \geqslant d_K(f|^{[k]}, g|^{[k]}) + \sum_{i=k+1}^{n} |\Phi(f)_i - \Phi(g)_i|.$$

*Proof:* The proof is by induction on $r = n - k$. As the base case, the inequality is clearly satisfied for $r = 0$, i.e., $n = k$. Now consider the inductive step. Suppose that the inequality holds for some $r - 1 = n - k - 1$, and we will now show that it also holds for $r = n - k$.

Consider a sequence of $d_K(f, g)$ adjacent transpositions that changes the permutation $f$ into the permutation $g$. Of these transpositions, assume that $\alpha$ adjacent transpositions involve the integer $n$, and $\beta$ adjacent transpositions do not involve $n$. Clearly,

$$d_K(f, g) = \alpha + \beta.$$

Since the integer $n$ needs to be moved from position $n - \Phi(f)_n$ to position $n - \Phi(g)_n$, we get

$$\alpha \geqslant |\Phi(f)_n - \Phi(g)_n|.$$

Note that those adjacent transpositions that involve $n$ do not change the relative order of the integers $[n-1]$ in the permutation. Thus, to transform the integers $[n-1]$ from their relative order in $f$ to their relative order in $g$, by the induction assumption, we get

$$\beta \geqslant d_K(f|^{[k]}, g|^{[k]}) + \sum_{i=k+1}^{n-1} |\Phi(f)_i - \Phi(g)_i|.$$

That leads to the conclusion. ■

*Example 6: Let $n = 3$ and $k = 2$ and consider*

$$f = [1, 3, 2] \text{ and } g = [2, 1, 3].$$

*In this case, the inequality of Lemma 5 becomes an equality since*

$$\begin{aligned} &d_K([1, 3, 2], [2, 1, 3]) = 2 \\ &\quad = 1 + |1 - 0| = d_K([1, 2], [2, 1]) + |\Phi(f)_3 - \Phi(g)_3|. \end{aligned}$$

*The equality, however, does not always hold. For instance, if*

$$f' = [1, 3, 2] \text{ and } g' = [2, 3, 1],$$

*we get*

$$d_K([1,3,2],[2,3,1]) = 3 > 1 + |1-1| = d_K([1,2],[2,1]) + |\Phi(f)_3 - \Phi(g)_3|.$$

□

We now present an inequality for ball sizes in $S_n$, which will be useful for the analysis of systematic codes. Given a permutation $f \in S_n$, the ball of radius $r$ centered at $f$, is defined by

$$\mathfrak{B}_r(f) = \{g \in S_n \mid d_K(f,g) \leqslant r\},$$

for any $0 \leqslant r \leqslant \binom{n}{2}$. We recall that the maximum distance for any two permutations in $S_n$ is $\binom{n}{2}$ (for example, see [13]). A simple relabeling argument suffices to show that the size of a ball does not depend on the choice of its center. Therefore, we will use $|\mathfrak{B}_r(n)|$ to denote $|\mathfrak{B}_r(f)|$ for any $f \in S_n$.

An exact expression for $|\mathfrak{B}_r(n)|$ is known [19]. However, for our purposes, we will use the inequality of the following lemma.

*Lemma 7: For all $n \geqslant 1$ and $0 \leqslant r \leqslant \binom{n}{2}$,*

$$|\mathfrak{B}_r(n)| \leqslant \binom{n+r-1}{n-1}.$$

*Proof:* Since the center of a ball does not affect its size, consider the ball centered at the identity, $\mathfrak{B}_r(\mathrm{Id})$. It follows from (1) that

$$|\mathfrak{B}_r(\mathrm{Id})| \leqslant |\{f \in S_n \mid d_1(\Phi(f), \Phi(\mathrm{Id})) \leqslant r\}|. \tag{2}$$

Since, conveniently, $\Phi(\mathrm{Id})$ is the all-zero vector, we have for any $f \in S_n$ that

$$d_1(\Phi(f), \Phi(\mathrm{Id})) = \sum_{i=1}^{n} \Phi(f)_i.$$

We further note that $\Phi(f)_1 = 0$ always.

Thus, the right-hand side of (2) is upper bounded by the number of non-negative-integer vectors of length $n-1$ whose entry sum is at most $r$. This is easily seen to be the same as the number of ways $r$ identical balls can be thrown into $n$ non-identical bins, and hence,

$$|\mathfrak{B}_r(n)| \leqslant \binom{n+r-1}{n-1}.$$

■

### B. Systematic Single-Error-Correcting Codes

We start by presenting two constructions for systematic $[k+2, k, 3]$ codes, capable of correcting a single error. The first construction uses a direct manipulation of the permutations to construct the codewords, and is somewhat restricted in its choice of parameters. In contrast, the second construction uses a metric embedding technique, and applies to all parameters. We then show the codes are optimal unless perfect single-error-correcting codes exist.

*Construction A: Let $k \geqslant 3$ be an integer such that either $k$ or $k+1$ is a prime. For any $f \in S_k$, and for any integer $j \geqslant 1$, we define the following function:*

$$\rho_j(f) = \left(\sum_{i=1}^{k} (2i-1)^j f(i)\right) \bmod m, \tag{3}$$

*where $m = k$ if $k$ a prime and $m = k+1$ if $k+1$ is a prime.*

*We construct the code*

$$C = \left\{ f \in S_{k+2} \;\middle|\; \Phi(f)_{k+j} = \rho_j(f|^{[k]}), \text{ for all } j \in [2] \right\}.$$

□

*Theorem 8: The code $C$ from Construction A is a systematic $[k+2, k, 3]$ code in Kendall's $\tau$-metric.*

*Proof:* We easily observe that the information symbols $[k]$ are unconstrained, and so

$$\left\{ f|^{[k]} \;\middle|\; f \in C \right\} = S_k.$$

Furthermore, since a choice of the order of the information symbols determines the positions of the two redundancy symbols uniquely, we also have $|C| = k!$.

It now only remains to show that the minimum distance of $C$ is 3. We know that either $k$ is a prime, or $k+1$ is a prime. Let us first handle the former case. Let $f, g \in C$ be two codewords, $f \neq g$. We divide our proof into three cases, depending on $d_K(f|^{[k]}, g|^{[k]})$.

*a) Case 1: $d_K(f|^{[k]}, g|^{[k]}) = 1$.* In this case, we can write

$$f|^{[k]} = [a_1, a_2, \ldots, a_i,\ a_{i+1}, \ldots, a_k],$$
$$g|^{[k]} = [a_1, a_2, \ldots, a_{i+1},\ a_i, \ldots, a_k]$$

for some $i \in [k-1]$, i.e., $f|^{[k]}$ and $g|^{[k]}$ differ by an adjacent transposition of the $i$th and $(i+1)$st elements.

Let us now define $\Delta = a_{i+1} - a_i$. It follows that

$$\Phi(f)_{k+1} - \Phi(g)_{k+1} \equiv 2\Delta \pmod{k}.$$

Since $1 \leqslant |\Delta| \leqslant k-1$ and $k \geqslant 3$ is a prime, we know that $2\Delta$ is not a multiple of $k$. As a result, we get

$$|\Phi(f)_{k+1} - \Phi(g)_{k+1}| \geqslant 1.$$

Similarly, we have

$$\begin{aligned}\Phi(f)_{k+2} - \Phi(g)_{k+2} &\equiv (2i-1)^2 a_i + (2i+1)^2 (a_i + \Delta) \\ &\quad -(2i-1)^2 (a_i + \Delta) - (2i+1)^2 a_i \\ &\equiv 8i\Delta \pmod{k}.\end{aligned}$$

Again, $8i\Delta$ is not a multiple of $k$ since $1 \leqslant i, |\Delta| \leqslant k-1$ and $k \geqslant 3$ is a prime. This implies that

$$|\Phi(f)_{k+2} - \Phi(g)_{k+2}| \geqslant 1.$$

Thus, by Lemma 5, we get

$$\begin{aligned}d_K(f,g) &\geqslant d_K(f|^{[k]}, g|^{[k]}) + \sum_{i=k+1}^{k+2} |\Phi(f)_i - \Phi(g)_i| \\ &\geqslant 1 + 1 + 1 = 3.\end{aligned}$$

*b) Case 2: $d_K(f|^{[k]}, g|^{[k]}) = 2$.* Let us denote

$$f|^{[k]} = [a_1, a_2, \ldots, a_k].$$

By our assumption, there exist $1 \leqslant i, j \leqslant k-1$ such that $g$ is obtained from $f$ as a result of two adjacent transpositions: one exchanging locations $i$ and $i+1$, and one exchanging locations $j$ and $j+1$. We distinguish between two cases.

In the first case, $\{i, i+1\} \cap \{j, j+1\} = \emptyset$. Without loss of generality, assume $i < j$, and then we have

$$g|^{[k]} = [a_1, \ldots, a_{i+1}, a_i, \ldots, a_{j+1}, a_j, \ldots, a_k].$$

Let us define $\Delta_1 = a_{i+1} - a_i$, and $\Delta_2 = a_{j+1} - a_j$. Then we get

$$\Phi(f)_{k+1} - \Phi(g)_{k+1} \equiv 2(\Delta_1 + \Delta_2) \pmod{k}.$$

If $\Delta_1 + \Delta_2$ is not a multiple of $k$, then by the same reasoning as before,

$$|\Phi(f)_{k+1} - \Phi(g)_{k+1}| \geqslant 1.$$

If $\Delta_1 + \Delta_2$ is a multiple of $k$, then we can write $\Delta_2 \equiv -\Delta_1 \pmod{k}$. Hence,

$$\begin{aligned}\Phi(f)_{k+2} - \Phi(g)_{k+2} \equiv\ & (2i-1)^2 a_i + (2i+1)^2 (a_i + \Delta_1) \\ & + (2j-1)^2 a_j + (2j+1)^2 (a_j - \Delta_1) \\ & - (2i-1)^2 (a_i + \Delta_1) - (2i+1)^2 a_i \\ & - (2j-1)^2 (a_j - \Delta_1) - (2j+1)^2 a_j \\ \equiv\ & 8(i-j)\Delta_1 \pmod{k}.\end{aligned}$$

Since $8(i-j)\Delta_1$ is not a multiple of $k$, we have

$$|\Phi(f)_{k+2} - \Phi(g)_{k+2}| \geqslant 1.$$

In the second case, $\{i, i+1\} \cap \{j, j+1\} \neq \emptyset$. Thus, either

$$g|^{[k]} = [a_1, \ldots, a_{i+2},\ a_i, a_{i+1}, \ldots, a_k],$$

or

$$g|^{[k]} = [a_1, \ldots, a_{i+1},\ a_{i+2}, a_i, \ldots, a_k],$$

for some $i \in [k-2]$. By defining $\Delta_1 = a_{i+2} - a_{i+1}$ and $\Delta_2 = a_{i+2} - a_i$ in the first case, or $\Delta_1 = a_{i+1} - a_i$ and $\Delta_2 = a_{i+2} - a_i$ in the second case, and with the same arguments as above, it can be proved that either

$$|\Phi(f)_{k+1} - \Phi(g)_{k+1}| \geqslant 1,$$

or

$$|\Phi(f)_{k+2} - \Phi(g)_{k+2}| \geqslant 1.$$

Combining all the cases together, by Lemma 5, we get

$$\begin{aligned}d_K(f, g) &\geqslant d_K(f|^{[k]}, g|^{[k]}) + \sum_{i=k+1}^{k+2} |\Phi(f)_i - \Phi(g)_i| \\ &\geqslant 2 + 1 = 3.\end{aligned}$$

*c) Case 3:* $d_K(f|^{[k]}, g|^{[k]}) \geqslant 3$. This is the easiest case, since by Lemma 5,

$$d_K(f, g) \geqslant d_K(f|^{[k]}, g|^{[k]}) \geqslant 3.$$

Finally, we note that if $k+1$ is a prime, we can repeat the proof in its entirety, replacing mod$k$ with mod$(k+1)$. ■

Before continuing to the next construction we would like to consider encoding and decoding algorithms for the code from Construction A. For the encoding procedure, we start by mapping an integer from $\mathbb{Z}_{k!}$ to a permutation $f' \in S_k$. This may be accomplished in linear time [22]. Then, using the description of Construction A, the two redundancy symbols are easily placed in their correct position, and we receive a codeword $f \in C$ such that $f|^{[k]} = f'$.

Decoding may be done efficiently as well. Assume $f \in C \subseteq S_{k+2}$ was transmitted, while $g \in S_{k+2}$ was received, where $d_K(f, g) \leqslant 1$. A trivial decoding algorithm can check the $k+2$ permutation in the ball of radius 1 centered around $g$, and find the unique codeword $f$ in it. This algorithm takes $O(k^2)$ steps.

We can do better than that, using the decoding algorithm we now describe. Let $\hat{g} \in C$ be the unique codeword in $C$ having the same order of information symbols as $g$, i.e., $\hat{g}|^{[k]} = g|^{[k]}$. If $d_K(\hat{g}, g) \leqslant 1$, then $f = \hat{g}$ is the correct decoding. Otherwise, $d_K(f|^k, g|^k) = 1$, and we can write

$$\begin{aligned}f|^{[k]} &= [a_1, \ldots, a_i,\ a_{i+1}, \ldots, a_k], \\ g|^{[k]} &= [a_1, \ldots, a_{i+1},\ a_i, \ldots, a_k],\end{aligned}$$

for some $i \in [k-1]$.

Since a single adjacent transposition changed the order of two information symbols, we deduce no redundancy symbols were moved, and thus,

$$\Phi(f)_{k+1} = \Phi(g)_{k+1} \text{ and } \Phi(f)_{k+2} = \Phi(g)_{k+2}.$$

According to the proof of Theorem 8,

$$\begin{aligned}\Phi(g)_{k+1} - \Phi(\hat{g})_{k+1} &\equiv 2(a_{i+1} - a_i) \pmod{m}, \\ \Phi(g)_{k+2} - \Phi(\hat{g})_{k+2} &\equiv 8i(a_{i+1} - a_i) \pmod{m},\end{aligned}$$

where $m$ is the prime in $\{k, k+1\}$. Combining the two equations together we get

$$\Phi(g)_{k+2} - \Phi(\hat{g})_{k+2} \equiv 4i \left(\Phi(g)_{k+1} - \Phi(\hat{g})_{k+1}\right) \pmod{m},$$

and we can easily solve for $i$, thus recovering the coordinate of the adjacent transposition. This decoding algorithm runs in $O(k)$ steps. We illustrate the decoding algorithm with the following example.

*Example 9: Let $k = 4$, and assume we would like to encode $[4, 1, 3, 2]$. Thus, by Construction A, we look for a permutation $f \in S_6$ such that*

$$\begin{aligned}\Phi(f)_5 &= \left(\sum_{i=1}^{4} (2i-1) f(i)\right) \bmod 5 = 1, \\ \Phi(f)_6 &= \left(\sum_{i=1}^{4} (2i-1)^2 f(i)\right) \bmod 5 = 1.\end{aligned}$$

*We therefore transmit*

$$f = [4, 1, 3, 5, 6, 2],$$

*and let us assume the received permutation is*

$$g = [4, 3, 1, 5, 6, 2],$$

*due to an adjacent transposition in positions 2 and 3. We extract the information symbols from $g$ to obtain,*

$$g|^{[k]} = [4, 3, 1, 2],$$

*and use that to construct the codeword*

$$\hat{g} = [4, 6, 3, 5, 1, 2].$$

*Since $d_K(\hat{g}, g) > 1$, we deduce that two information symbols changed positions. Since*

$$\Phi(g)_{k+1} = 1 \quad \Phi(g)_{k+2} = 1$$
$$\Phi(\hat{g})_{k+1} = 2 \quad \Phi(\hat{g})_{k+2} = 4,$$

*we solve*

$$1 - 4 \equiv 4i(1 - 2) \pmod 5,$$

*resulting in the correct positions of the adjacent transposition, $i = 2$ and $i + 1 = 3$.* □

Another strategy for constructing rank-modulation codes for Kendall's $\tau$-metric, which was already employed by [2] and [13], is to first construct a code $C^*$ with minimum $\ell_1$-distance $d$ in $\mathbb{Z}^n$, and then take

$$C = \Phi^{-1}(C^* \cap \mathbb{Z}_n!),$$

i.e., exactly those permutations whose factoradic representations are $C^* \cap \mathbb{Z}_n!$. Since by (1), the distance can only increase, the resulting set of permutations is a code with minimum Kendall's $\tau$-distance of at least $d$. The main challenge with this approach is to ensure a large intersection of $C^*$ with $\mathbb{Z}_n!$.

For the construction of systematic codes we shall employ the same methods, however, now we have an additional challenge: we also require the intersection $C^* \cap \mathbb{Z}_n!$ to have at least one vector for each possible prefix from $\mathbb{Z}_k!$.

*Construction B: Let $k \geqslant 2$ be some integer. For a vector $x = (x_1, x_2, \ldots, x_{k+1}) \in \mathbb{Z}^{k+1}$, and for all $m \in \mathbb{Z}$, we denote*

$$s_m(x) = \left( \sum_{i=1}^{m} i x_i \right) \bmod (2k + 3).$$

*We construct a subset $C' \subseteq \mathbb{Z}^{k+1}$ defined by*

$$C' = \{x \in \mathbb{Z}^{k+1} | x_k = \lfloor s_{k-1}(2x)/3 \rfloor, \\ x_{k+1} = s_{k-1}(2x) \bmod 3\}.$$

*We denote by*

$$C^* = \left\{(0, x_1, x_2, \ldots, x_{k+1}) \mid (x_1, x_2, \ldots, x_{k+1}) \in C'\right\},$$

*the prepending of 0 to all the codewords of $C'$. The constructed code is*

$$C = \Phi^{-1}\left(C^* \cap \mathbb{Z}_{k+2}!\right).$$

□

*Theorem 10: For all $k \geqslant 2$, the code $C$ from Construction B is a $[k + 2, k, 3]$ systematic code in Kendall's $\tau$-metric.*

*Proof:* Consider the perfect $(k + 1)$-dimensional single-error-correcting code in the $\ell_1$-metric described by Golomb and Welch in [9] and given by,

$$C'' = \left\{ x = (x_1, x_2, \ldots, x_{k+1}) \in \mathbb{Z}^{k+1} \;\middle|\; s_{k+1}(x) = 0 \right\}.$$

We contend that $C' \subseteq C''$, i.e., that $C'$ is also a single-error-correcting code in the $\ell_1$-metric. Indeed, let $x = (x_1, \ldots, x_{k+1}) \in C'$ be a codeword in $C'$. Then, noting that

$$k \equiv 3(k + 1) \pmod{2k + 3},$$

and working modulo $2k + 3$, and we get

$$\begin{aligned} s_{k+1}(x) &\equiv s_{k-1}(x) + k x_k + (k + 1) x_{k+1} \\ &\equiv s_{k-1}(x) + k \lfloor s_{k-1}(2x)/3 \rfloor \\ &\quad + (k + 1)\,(s_{k-1}(2x) \bmod 3) \\ &\equiv s_{k-1}(x) + (k + 1)(3 \lfloor s_{k-1}(2x)/3 \rfloor \\ &\qquad + (s_{k-1}(2x) \bmod 3)) \\ &\equiv s_{k-1}(x) + (k + 1) s_{k-1}(2x) \\ &\equiv s_{k-1}(x) + 2(k + 1) s_{k-1}(x) \\ &\equiv (2k + 3) s_{k-1}(x) \equiv 0 \pmod{2k + 3}. \end{aligned}$$

Thus, $x \in C''$, and so $C' \subseteq C''$.

We note how the first $k - 1$ coordinates of the codewords of $C'$ are unconstrained. Thus, for all $1 \leqslant i \leqslant k - 1$ we can set $x_i \in \mathbb{Z}_{i+1}$ arbitrarily in any one of $k!$ ways. Furthermore, for any $x \in C'$,

$$0 \leqslant \lfloor s_{k-1}(2x)/3 \rfloor \leqslant \frac{2(k + 1)}{3} \leqslant k,$$

as well as

$$0 \leqslant s_{k-1}(2x) \bmod 3 \leqslant k + 1.$$

It follows that $x_k \in \mathbb{Z}_{k+1}$ and $x_{k+1} \in \mathbb{Z}_{k+2}$. Hence, after prepending a 0 to the codewords to obtain $C^*$, we get

$$\left|C^* \cap \mathbb{Z}_{k+2}!\right| = k!.$$

Finally, prepending the 0 does not change the minimum distance, and so $C^*$ has minimum $\ell_1$-distance of 3, and therefore, so does the final constructed code $C = \Phi^{-1}(C^* \cap \mathbb{Z}_{k+2}!)$. ■

Encoding the code from Construction B is extremely easy. In the factoradic representation we arbitrarily fill in the first $k - 1$ entries. The last two digits are determined by the first $k - 1$ digits, and a 0 is then prepended. We then convert the factoradic representation to a permutation, which is the desired codeword. The entire procedure takes $O(k)$ steps if we use [22] to convert from the factoradic representation to permutations.

The decoding process is simple as well. Given a permutation read from the channel, we first translate it to its factoradic representation and remove the leading 0. The remaining $k + 1$ coordinates are decoded using any simple procedure for decoding the Golomb-Welch code [9]. Again, the entire procedure takes $O(n)$ steps.

We note that the two constructions may produce different codes. As an example, the $[5, 3, 3]$ code from Construction A contains the codewords

$$f = [1, 4, 3, 2, 5] \quad \text{and} \quad g = [2, 3, 4, 1, 5].$$

However,

$$\Phi(f) = [0, 0, 1, 2, 0] \quad \text{and} \quad \Phi(g) = [0, 1, 1, 1, 0].$$

Since

$$d_1(\Phi(f), \Phi(g)) = 2,$$

the code cannot have originated from Construction B.

An obvious question to ask is how good are the parameters of the codes presented in Construction A and Construction B. Any $(n, M, d)$ code (systematic or not) has to satisfy the ball-packing bound:

$$M \leqslant \frac{n!}{|\mathfrak{B}_r(n)|}, \tag{4}$$

where $r = \lfloor (d-1)/2 \rfloor$. Codes attaining (4) with equality are called *perfect*. We thus reach the following simple corollary:

*Corollary 11: The $[k+2, k, 3]$ systematic codes of Construction A and Construction B have optimal size, unless perfect systematic single-error-correcting codes exist in Kendall's $\tau$-metric.*

Example of perfect codes in other metrics are quite rare (see [21]). In Kendall's $\tau$-metric there is a simple $(3, 2, 3)$ that is perfect:

$$C = \{[1, 2, 3], [3, 2, 1]\}.$$

This code is also systematic, i.e., a $[3, 2, 3]$-code. However, beside this code, no other perfect code has been found yet. It was recently shown in [4], that no perfect codes exist in $S_n$ under Kendall's $\tau$-metric when $n$ is a prime, or when $4 \leqslant n \leqslant 10$.

To summarize, the $[k+2, k, 3]$ codes presented have minimal redundancy among systematic codes, unless there exists a perfect systematic $[k+1, k, 3]$ single-error-correcting code. Furthermore, compared with the single-error-correcting code presented in [13], the codes presented here have more efficient encoding and decoding algorithms.

### C. Multi-Error-Correcting Codes

After studying systematic single-error-correcting codes, we turn to consider systematic codes capable of correct more than one error. We will first describe an explicit construction for a wide range of parameters, and then turn to a greedy algorithm leading us to prove a non-constructive existence result.

The systematic single-error-correcting code in Construction A may be generalized in a straightforward way: for $1 \leqslant k \leqslant n$ and $r \geqslant 1$ integers we define,

$$C = \left\{ f \in S_{k+r} \;\middle|\; \Phi(f)_{k+j} = \rho_j(f|^{[k]}) \text{ for all } j \in [r] \right\},$$

where $\rho_j(\cdot)$ is given by (3). This gives us a family of codes, including a $[10, 4, 5]$ systematic code, and a $[14, 4, 7]$ systematic code. However, a general analysis of these codes is difficult.

We therefore return to the strategy of metric embedding: An $\ell_1$-metric code is constructed in such a way as to allow all possible values in the first few information entries, and then the other positions are determined as a function of the information entries.

*Construction C: Let $p$ be a prime, $m \geqslant 2$, $1 \leqslant t \leqslant \frac{p-3}{2}$, and $p + tm \leqslant n \leqslant p^m$. Arbitrarily choose $\alpha_1, \alpha_2, \ldots, \alpha_{n-1}$ to be $n-1$ distinct non-zero elements of* $\mathrm{GF}(p^m)$. *We define*

$$H = \begin{bmatrix} 1 & 1 & \ldots & 1 \\ \alpha_1 & \alpha_2 & \ldots & \alpha_{n-1} \\ \alpha_1^2 & \alpha_2^2 & \ldots & \alpha_{n-1}^2 \\ \vdots & \vdots & \ldots & \vdots \\ \alpha_1^t & \alpha_2^t & \ldots & \alpha_{n-1}^t \end{bmatrix}.$$

*Viewing* $\mathrm{GF}(p^m)$ *as the vector space* $\mathrm{GF}(p)^m$, *we can think of any entry of the form $\alpha_i^j$ in $H$ as a column vector of length $m$ over* $\mathrm{GF}(p)$. *Thus, we shall consider $H$ to be a $(t+1)m \times (n-1)$ matrix over* $\mathrm{GF}(p)$. *We denote*

$$k = n - \operatorname{rank} H.$$

*We construct a subset $C' \subseteq \mathbb{Z}^{n-1}$ defined by*

$$C' = \left\{ x \in \mathbb{Z}^{n-1} \;\middle|\; Hx \equiv 0 \pmod{p} \right\},$$

*where the entries of $Hx$ are computed modulo $p$.*

*We define the mapping $\mu : \mathbb{Z}^{n-1} \to \mathbb{Z}^n$ as follows,*

$$\mu(x_1, x_2, \ldots, x_{n-1}) = (0, x_1, \ldots, x_{k-1}, x_k \bmod p, \ldots, x_{n-1} \bmod p),$$

*i.e., prepending a zero and reducing the last $n-k$ entries modulo $p$. We then set*

$$C^* = \left\{ \mu(x) \;\middle|\; x \in C' \right\}.$$

*The constructed code is*

$$C = \Phi^{-1}(C^* \cap \mathbb{Z}_n!).$$

□

*Theorem 12: The code $C$ from Construction C is an $[n, k, 2t+2]$ systematic code in Kendall's $\tau$-metric capable of correcting $t$ errors. Furthermore, the code's redundancy satisfies $n - k \leqslant tm + 1$.*

*Proof:* The matrix $H$ is nothing but the parity-check matrix for a BCH code over $\mathrm{GF}(p)$. Since the code is linear, we can find a $(k-1) \times (n-1)$ generator matrix $G$ for the code, and in particular, we can require that it be systematic, i.e.,

$$G = [I_{k-1} | A],$$

where $I_{k-1}$ is the $(k-1) \times (k-1)$ identity matrix, and $A$ is a $(k-1) \times (n-k)$ matrix over $\mathrm{GF}(p)$. As a side note, getting to this systematic form, we may be required to permute the coordinates of the code. Since the order of elements $\alpha_1, \ldots, \alpha_{n-1}$, which are used to construct $H$, is irrelevant, we assume it is chosen so that no change of order of coordinates is required.

Let us denote by $C''$ the code whose generator matrix is $G$. We recall the definition of the Lee-distance measure over $\mathrm{GF}(p)$: Given two vectors $x, x' \in \mathrm{GF}(p)^{n-1}$,

$$d_L(x, x') = \sum_{i=1}^{n-1} \min\left(x_i - x_i', x_i' - x_i\right),$$

where subtraction is done in $\mathrm{GF}(p)$. It was shown in [24, Th. 1], that when $t + 1 \leqslant (p-1)/2$, the minimum Lee distance of the code $C''$ is at least $2t+2$. Since we require $t \leqslant (p-3)/2$, this is also the case here.

Our next goal is to transform $C''$ to a code over $\mathbb{Z}^{n-1}$ with a minimum $\ell_1$-distance guarantee. Since we have the code $C''$ reside in the $(n-1)$-dimensional cube $\mathrm{GF}(p)^{n-1}$, we can place copies of that cube and tile the entire space $\mathbb{Z}^{n-1}$. This is known as Construction A of [20], and the resulting code is exactly

$$C' = \left\{ x \in \mathbb{Z}^{n-1} \;\middle|\; Hx \equiv 0 \pmod{p} \right\},$$

where the entries of $Hx$ are computed modulo $p$. Again, by [20], the codewords of $C'$ are spanned (using linear combinations with integer coefficients) by the generating matrix

$$G' = \begin{bmatrix} I_{k-1} & A \\ 0 & pI_{n-k} \end{bmatrix}.$$

Thus, the minimum Lee distance of $2t+2$ between codewords of $C''$, and our requirement that $2t+2 \leqslant p-1$, guarantee a minimum $\ell_1$-distance of $2t+2$ between codewords of $C'$.

A quick inspection of $G'$ reveals that, due to the first $k-1$ rows, any prefix of $k-1$ integers may be completed to a length-$n$ codeword in $C'$. Furthermore, given a codeword in $C'$, by reducing its last $n-k$ entries modulo $p$ we obtain another codeword of $C'$, due to the last $n-k$ rows of $G'$. It follows that

$$C^* = \{\mu(x) \mid x \in C'\},$$

is a subset of the codewords of $C'$ with a 0 prepended.

At this point we contend that

$$|C^* \cap \mathbb{Z}_n!| = k!.$$

To prove this, we need to show that

$$x_i \bmod p \in \mathbb{Z}_{i+1}, \tag{5}$$

for all $k \leqslant i \leqslant n-1$. It is well known (see also [24]) that when $H$ is viewed as a $(t+1)m \times (n-1)$ matrix over GF$(p)$,

$$n-k = \operatorname{rank} H \leqslant tm+1. \tag{6}$$

Thus, to prove (5), it suffices to verify it for the smallest possible value of $k$, which by (6), is $n-tm-1$. Since, in the construction, we required $p+tm \leqslant n$,

$$x_k \bmod p \leqslant p-1 \leqslant n-tm-1 \leqslant k,$$

and necessarily (5) holds. Hence, $C$ is indeed an $[n, k, 2t+2]$ systematic code in Kendall's $\tau$-metric, with redundancy at most $tm+1$. ∎

Again, encoding and decoding are easily done. For an encoding procedure, take any vector $(0|u) \in \mathbb{Z}_k!$ and map it to

$$(0|u) \mapsto (0|u|uA \bmod p) \in C^*.$$

The permutation whose factoradic representation is given by this vector is the encoded permutation.

For a decoding procedure, map the received permutation to its factoradic representation, and use the decoding for the Lee-metric code (essentially, a BCH decoding procedure) given in [24].

We also note that for the least redundancy, we would like to choose $m=2$ in Construction C. To show that there are infinitely many parameters for which the construction works we present the following corollary.

*Corollary 13: For any $t \geqslant 1$, and $6t+3 \leqslant n \leqslant (2t+3)^2$, there exists a prime $p$, such that the requirements of Construction C are satisfied with $m=2$, and therefore there exists an $[n, k, 2t+2]$ systematic code with redundancy at most $2t+1$.*

*Proof:* We recall Bertrand's postulate (for example, see [10, section 22.3]), stating that for any integer $s > 1$, there exists a prime $s < p < 2s$. Given $t$ and $n$ we would like to find a prime $p$ to satisfy the requirements of Construction C with parameter $m=2$. These requirements

$$p \geqslant 2t+3, \tag{7}$$

$$p \leqslant n-2t, \tag{8}$$

$$p^2 \geqslant n. \tag{9}$$

If we have $n-2t+1 \geqslant 2(2t+2)$, then by Bertrand's postulate, there is a prime $p$ satisfying (7) and (8). Rearranged, this becomes $n \geqslant 6t+3$. Finally, if we require $n \leqslant (2t+3)^2$, then $p^2 \geqslant (2t+3)^2 \geqslant n$, satisfying (9). ∎

Along the same lines, but using two embeddings, one after the other, we present a construction transforming systematic binary codes under the Hamming metric, into systematic codes of permutations under Kendall's $\tau$-metric. We recall the definition of the Hamming-distance measure over $\{0,1\}^m$: Given two vectors $x, x' \in \{0,1\}^m$,

$$d_H(x, x') = |\{i \in [m] \mid x_i \neq x'_i\}|.$$

The construction is a simple modification of the construction given in [23]. The main idea for the first embedding is to use a mapping $\mathcal{G}_m : \mathbb{Z}_{2^m} \to \{0,1\}^m$ such that for any two integers $t_1, t_2 \in \mathbb{Z}_{2^m}$,

$$|t_1 - t_2| \geqslant d_H(\mathcal{G}_m(t_1), \mathcal{G}_m(t_2)), \tag{10}$$

where $d_H(\cdot,\cdot)$ denotes the Hamming distance function. By convention, $\mathcal{G}_0$ is the mapping returning the unique vector of length 0. A simple way of creating such a mapping is to use the encoding function for an optimal Gray code (see [25] for a survey of Gray codes). Additionally, before presenting the construction we require the following two identities from [19, section 5.3.1, eq. (3)] and [18, section 1.2.4, Example 42(b)], respectively:

$$\sum_{i=1}^{\ell} \lceil \log_2 i \rceil = \ell \lceil \log_2 \ell \rceil - 2^{\lceil \log_2 \ell \rceil} + 1,$$

$$\sum_{i=1}^{\ell} \lfloor \log_2 i \rfloor = (\ell+1) \lfloor \log_2 \ell \rfloor - 2^{\lfloor \log_2 \ell \rfloor + 1} + 2.$$

It is also easily seen that

$$\sum_{i=1}^{\ell} \left( \lceil \log_2 i \rceil - \lfloor \log_2 i \rfloor \right) = \ell - \lfloor \log_2 \ell \rfloor - 1.$$

*Construction D: Let $C'$ be an $(n', 2^{k'}, d)$ binary systematic code in the Hamming metric, where the first $k'$ coordinates are systematic. Furthermore, let $k$ and $n$ be integers such that*

$$k' = \sum_{i=1}^{k} \lceil \log_2 i \rceil = k \lceil \log_2 k \rceil - 2^{\lceil \log_2 k \rceil} + 1, \tag{11}$$

$$\begin{aligned} n' &= \sum_{i=1}^{k} \lceil \log_2 i \rceil + \sum_{i=k+1}^{n} \lfloor \log_2 i \rfloor \\ &= (n+1) \lfloor \log_2 n \rfloor - 2^{\lfloor \log_2 n \rfloor + 1} \\ &\quad + k - \lfloor \log_2 k \rfloor + 1. \end{aligned} \tag{12}$$

*We conveniently define*

$$\lambda(i) = \begin{cases} \lceil \log_2 i \rceil & 1 \leqslant i \leqslant k, \\ \lfloor \log_2 i \rfloor & k+1 \leqslant i \leqslant n. \end{cases}$$

*We now construct the following code,*

$$C = \left\{ f \in S_n \;\middle|\; \mathcal{G}_{\lambda(1)}(\Phi(f)_1) || \ldots || \mathcal{G}_{\lambda(n)}(\Phi(f)_n) \in C' \right\}.$$

*where || denotes vector concatenation. In particular, the notation implies that for any $f \in C$,*

$$0 \leqslant \Phi(f)_i \leqslant 2^{\lfloor \log_2 i \rfloor} - 1 \leqslant i - 1,$$

*for all $k+1 \leqslant i \leqslant n$.* □

*Theorem 14: The code $C$ from Construction D is an $[n, k, d]$ systematic code in Kendall's $\tau$-metric.*

*Proof:* The length of the code is obviously $n$. Let us try to build a codeword $f \in C$. We note that the first $k$ symbols of $\Phi(f)$ form a binary vector of length $k'$ after being Gray-mapped and concatenated. Since the first $k'$ bits of the code $C'$ are systematic, any such $k'$-prefix may be uniquely completed to form a codeword in $C'$ by adding appropriate $n'-k'$ redundancy bits. These redundancy bits can be divided into sets of size $\lfloor \log_2 i \rfloor$, with $k+1 \leqslant i \leqslant n$. Thus, the reverse Gray mapping of these sets uniquely determines $\Phi(f)_{k+1}, \ldots, \Phi(f)_n$, and therefore, $f$ as well. It follows that $C$ is indeed a systematic code of length $n$ and $k$ information symbols.

Finally, let $f, g \in C$ be two distinct codewords. Then, using (1) and (10) we get

$$\begin{aligned} d_K(f,g) &\geqslant \sum_{i=1}^{n} |\Phi(f)_i - \Phi(g)_i| \\ &\geqslant \sum_{i=1}^{n} d_H\left(\mathcal{G}_{\lambda(i)}(\Phi(f)_i), \mathcal{G}_{\lambda(i)}(\Phi(g)_i)\right) \\ &\geqslant d. \end{aligned}$$

Thus, $C$ is an $[n, k, d]$ systematic code. ■

Using Construction D we obtain the following codes.

*Theorem 15: For any fixed $t \geqslant 1$, and for large-enough $k$, Construction D produces a $[k+t+1, k, 2t+1]$ systematic code in Kendall's $\tau$-metric.*

*Proof:* Fix the values of $k$ and $t$, and denote $n = k+t+1$. When plugging these values into Construction D, (12) becomes

$$\begin{aligned} n' = (k+t+2)\lfloor \log_2(k+t+1) \rfloor - 2^{\lfloor \log_2(k+t+1) \rfloor + 1} \\ + k - \lfloor \log_2 k \rfloor + 1. \end{aligned}$$

We now take an $[n', k', d]$ BCH code (that can be made systematic). It is well known (see [21]) that the number of errors such a code is capable of correcting is at least

$$\left\lfloor \frac{n'-k'}{\lceil \log_2(n'+1) \rceil} \right\rfloor.$$

One can now verify that without the floor function,

$$\lim_{k \to \infty} \frac{n'-k'}{\lceil \log_2(n'+1) \rceil} = t+1.$$

Thus, for a large-enough value of $k$, the BCH code is capable of correcting at least $t$ errors, and $d \geqslant 2t+1$. A direct application of Theorem 14 completes the proof. ■

We comment that encoding and decoding is done in a similar manner to the previous constructions. The complexity of these procedures is dominated by the complexity of encoding and decoding a BCH code, and is equal to the complexity of the corresponding procedure suggested in [23]. When fixing the number of correctable errors, $t$, we observe that the size of the length-$n$ code from Theorem 15 is $\Omega(n!/n^{t+1})$, which is slightly worse than $\Omega(n!/(n^t \log^t n))$, the size of the corresponding non-systematic code from [23].

As a final note, Construction D together with a binary BCH code, cannot cover the case of $d = \Omega(n)$. However, other constructions following the same basic idea may prove useful for this regime.

### D. Capacity of Systematic Codes

In this section, we prove that for rank modulation under Kendall's $\tau$-metric, systematic error-correcting codes achieve the same capacity as general error-correcting codes.

In [2], Barg and Mazumdar derived the capacity of general error-correcting codes for rank modulation under Kendall's $\tau$-metric. Let $A(n,d)$ denote the maximum size of an $(n, M, d)$ code. We define the capacity of error-correcting codes of minimum distance $d$ as

$$\mathsf{cap}(d) = \lim_{n \to \infty} \frac{\ln A(n,d)}{\ln n!}.$$

It was shown in [2] that

$$\mathsf{cap}(d) = \begin{cases} 1 & \text{if } d = O(n), \\ 1-\epsilon & \text{if } d = \Theta(n^{1+\epsilon}) \text{ with } 0 < \epsilon < 1, \\ 0 & \text{if } d = \Theta(n^2). \end{cases}$$

Turning to systematic codes, let $k(n,d)$ denote the maximum number of information symbols in systematic codes of length $n$ and minimum distance $d$. Such codes are $[n, k(n,d), d]$ systematic codes, and have $k(n,d)!$ codewords. The *capacity* of systematic codes of minimum distance $d$ is defined as

$$\mathsf{cap}_{\mathrm{sys}}(d) = \lim_{n \to \infty} \frac{\ln k(n,d)!}{\ln n!}.$$

Before proceeding with the main result of this section, we require the existence of codes with certain parameters. We show this existence by means of a Gilbert-Varshamov-like procedure.

*Theorem 16: Let $2 \leqslant k < n$ and $d \geqslant 1$ be integers such that*

$$\sum_{i=1}^{d-1} \binom{k+i-2}{i} \binom{d-i-1+n-k}{n-k} 2^{\min(d-i-1, n-k)} < \frac{n!}{k!}.$$

*Then there exists an $[n, k, d]$ systematic code in Kendall's $\tau$-metric.*

*Proof:* The proof strategy has three main parts. We first describe a procedure for constructing a code. We then prove that a successful run of the procedure indeed produces a code

with the right parameters. Finally, we prove that given the theorem requirements, the procedure is always successful.

We start by describing the procedure. We define $C_0 = \emptyset$. For all $i \geqslant 1$, the procedure searches for $f \in S_n$ such that

$$\min_{g \in C_{i-1}} d_K(f, g) \geqslant d, \tag{13}$$

and

$$f|^{[k]} \notin \left\{ g|^{[k]} \;\middle|\; g \in C_{i-1} \right\}. \tag{14}$$

If we can continue the process, increasing $i$ by 1 at each iteration, and reach $i = k!$, then the procedure is successful, and the constructed code is $C = C_{k!}$. Otherwise, the procedure declares a failure.

Let us now show that a successful run of the procedure produces an $[n, k, d]$ systematic code. We start with an empty set, and at each stage we add a codeword that is not to close to previously chosen codewords. Thus, the minimum distance of the resulting code is $d$. The second requirement at each step, is that the information symbols do not repeat those of a previously chosen codeword. Thus,

$$\left\{ f|^{[k]} \;\middle|\; f \in C \right\} = S_k,$$

and the code is systematic.

For the final part of the proof, we would like to show that given the theorem requirements, the procedure is always successful. For any permutations $h \in S_k$ there are exactly $n!/k!$ permutations $f \in S_n$ such that $f|^{[k]} = h$. At each step $i$ of the procedure we shall arbitrarily choose $h \in S_k$ such that

$$h \notin \left\{ g|^{[k]} \;\middle|\; g \in C_{i-1} \right\}.$$

We shall then try to find $f \in S_n$ such that $f|^{[k]} = h$, i.e., $f$ satisfies requirement (14). Our goal is to show there is at least one such $f$ that also satisfies the requirement of (13).

Given any such $h \in S_k$, let us upper bound the number of permutations $f$ such that $f|^{[k]} = h$ but $f$ does not satisfy (13). Let $g \in C_{i-1}$ be a codeword chosen in some previous iteration, and assume $d_K(f, g) \leqslant d - 1$. Let us denote

$$d_K(f|^{[k]}, g|^{[k]}) = j \leqslant d - 1.$$

By Lemma 5, in order for us to have $d_K(f, g) \leqslant d - 1$, we must have

$$\sum_{t=1}^{n-k} |\Phi(f)_{k+t} - \Phi(g)_{k+t}| \leqslant d - j - 1.$$

Thus, we would like to count the number of integer vectors of length $n-k$, whose $\ell_1$ weight is at most $d - j - 1$. Choosing the magnitudes of the entries of such a vector is equivalent to the number of ways $d - j - 1$ identical balls can be placed in $n - k + 1$ non-identical bins. We also need to choose the sign for the non-zero entries of such a vector, and there are at most $\min(d - j - 1, n - k)$ such entries. It follows, that an upper bound on the number of permutations $f \in S_n$ such that $f|^{[k]} = h$, and $d_K(f|^{[k]}, g|^{[k]}) = j$ for the given $g$, is

$$\binom{d - j - 1 + n - k}{n - k} 2^{\min(d-j-1, n-k)}.$$

Let $N_j$ denote the number of permutations $g \in C_{i-1}$ such that $d_K(f|^{[k]}, g|^{[k]}) = j$. If we had this number, then by a simple union bound, the total number of permutations $f \in S_n$ such that $f|^{[k]} = h$, but (13) does not hold, is upper bounded by

$$\sum_{j=1}^{d-1} N_j \binom{d - j - 1 + n - k}{n - k} 2^{\min(d-j-1, n-k)}.$$

To continue our upper bound, we replace $N_j$ with the larger $N'_j$, where $N'_j$ denotes the number permutations $h' \in S_k$ such that $d_K(h', h) = j$. Our upper bound is now

$$\sum_{j=1}^{d-1} N'_j \binom{d - j - 1 + n - k}{n - k} 2^{\min(d-j-1, n-k)}. \tag{15}$$

We do not have a nice closed-form expression for $N'_j$, and so, we would like to upper-bound (15). According to Lemma 7,

$$\sum_{t=0}^{j} N'_t = |\mathfrak{B}_j(k)| \leqslant \binom{k + j - 1}{k - 1}.$$

To relax the problem, we replace $N'_1, \ldots, N'_{d-1}$ with variables $x_1, \ldots, x_{d-1}$ that are non-negative integers with the same constraints on their partial sums, i.e.,

$$\sum_{t=0}^{j} x_t \leqslant \binom{k + j - 1}{k - 1}, \tag{16}$$

and where $x_0 = N'_0 = 1$. We further define

$$F(j) = 2^{\min(d-j-1, n-k)} \binom{d - j - 1 + n - k}{n - k},$$

and note that $F(j)$ is a decreasing function in $j$. Thus, to upper bound (15), we need to find values for $x_1, \ldots, x_{d-1}$, subject to (16), that maximize

$$\sum_{j=1}^{d-1} x_j F(j).$$

Since $F(j)$ is decreasing in $j$, the maximization problem is easily solved by setting

$$x_j = \binom{k + j - 1}{k - 1} - \binom{k + j - 2}{k - 1} = \binom{k + j - 2}{k - 2},$$

for $k \geqslant 2$ and $1 \leqslant j \leqslant d - 1$.

As a result, the number of permutations $f \in S_n$, such that $f|^{[k]} = h$, but (13) does not hold, is upper bounded by

$$\sum_{i=1}^{d-1} \binom{k + i - 2}{i} \binom{d - i - 1 + n - k}{n - k} 2^{\min(d-i-1, n-k)}. \tag{17}$$

Since the total number of permutations $f \in S_n$ such that $f|^{[k]} = h$ is $n!/k!$, if (17) is strictly less than $n!/k!$ then there exists a permutation $f$ satisfying (13). Since we did not restrict $h$ in any way, this conclusion holds for any iteration of the procedure, and the procedure succeeds. ■

*Example 17: When $d = 3$ and $n = k + 2$, the inequality of Theorem 16 can be simplified to*

$$6\binom{k - 1}{1} + \binom{k}{2} < (k + 1)(k + 2),$$

*which holds for any $k \geqslant 2$. Therefore, there exists a $[k+2, k, 3]$ systematic code for any $k \geqslant 2$. Note that this result is consistent with the codes built in Construction A and Construction B.* □

*Example 18: When $d = 4$ and $n = k + 3$, the inequality of Theorem 16 can be simplified to*

$$40\binom{k-1}{1} + 8\binom{k}{2} + \binom{k+1}{3} < (k+1)(k+2)(k+3),$$

*which holds for all $k \geqslant 2$. Therefore, there exists a $[k+3, k, 4]$ systematic code for any $k \geqslant 2$.* □

Building on Theorem 16, we now state the following useful theorem.

*Theorem 19: There exists a $[k+d, k, d]$ systematic code in Kendall's $\tau$-metric, for any $k \geqslant 2$ and $d \geqslant 1$.*

*Proof:* Based on Theorem 16, to show that there exists a $[k+d, k, d]$ systematic code, we only need to prove

$$\sum_{i=1}^{d-1}\binom{k+i-2}{i}\binom{2d-1-i}{d}2^{d-i-1} < \frac{(k+d)!}{k!}$$

for $k \geqslant 2$ and $d \geqslant 1$. We note that the case $d = 1$ is trivial, and so we will assume throughout the rest of the proof that $d \geqslant 2$. Furthermore, to simplify the proof, we will prove a stronger claim,

$$\sum_{i=1}^{d-1}\binom{k+i}{i}\binom{2d-1-i}{d}2^{d-i-1} < \frac{(k+d)!}{k!}. \tag{18}$$

Let us define

$$\psi_d(k) = \frac{k!}{(k+d)!}\sum_{i=1}^{d-1}\binom{k+i}{i}\binom{2d-1-i}{d}2^{d-i-1}.$$

We contend the $\psi_d(k)$ is non-increasing in $k$, and to prove this claim we consider $\psi_d(k+1)/\psi_d(k)$ and note that

$$\begin{aligned}\frac{\psi_d(k+1)}{\psi_d(k)} &= \frac{\frac{(k+1)!}{(k+d+1)!}}{\frac{k!}{(k+d)!}} \cdot \frac{\sum_{i=1}^{d-1}\binom{k+1+i}{i}\binom{2d-1-i}{d}2^{d-i-1}}{\sum_{i=1}^{d-1}\binom{k+i}{i}\binom{2d-1-i}{d}2^{d-i-1}} \\ &= \frac{\frac{(k+1)!}{(k+d+1)!}}{\frac{k!}{(k+d)!}} \cdot \frac{\sum_{i=1}^{d-1}\frac{k+1+i}{k+1}\binom{k+i}{i}\binom{2d-1-i}{d}2^{d-i-1}}{\sum_{i=1}^{d-1}\binom{k+i}{i}\binom{2d-1-i}{d}2^{d-i-1}} \\ &\leqslant \frac{k+1}{k+d+1} \cdot \frac{k+d}{k+1} < 1.\end{aligned}$$

Thus, $\psi_d(k)$ is indeed a non-increasing function of $k$. If $\psi_d(2) < 1$ for all $d \geqslant 2$, then for any $k, d \geqslant 2$, we surely have $\psi_d(k) < 1$, which proves (18). So our task is to prove $\psi_d(2) < 1$, namely,

$$\sum_{i=1}^{d-1}\binom{2+i}{i}\binom{2d-1-i}{d}2^{d-i-1} < \frac{(2+d)!}{2!}, \tag{19}$$

for all $d \geqslant 2$.

For all $2 \leqslant d \leqslant 16$ we can show that the inequality holds by computing the exact values. In what follows, we show that the inequality also holds when $d > 16$. The left-hand side of (19) may be upper bounded by

$$\begin{aligned}&\sum_{i=1}^{d-1}\binom{2+i}{i}\binom{2d-1-i}{d}2^{d-i-1} \\ &\quad\leqslant \frac{d(d+1)(d+2)}{2}\binom{2d-2}{d}2^{d-2}.\end{aligned}$$

Thus, to prove (19), it suffices to prove

$$\binom{2d-2}{d}2^{d-2} < (d-1)!.$$

We define

$$\xi(d) = \frac{1}{(d-1)!}\binom{2d-2}{d}2^{d-2}.$$

We can numerically check that $\xi(17) < 1$, and since

$$\frac{\xi(d+1)}{\xi(d)} = \frac{4d(2d+1)}{d(d+1)(d-1)} < 1$$

for all $d > 16$, we have $\xi(d) < 1$ for all $d > 16$, and this completes the proof. ■

The following theorem is the main result of this section. It shows that systematic codes have the same capacity as general codes. This is done by using the systematic codes whose existence is guaranteed by Theorem 19.

*Theorem 20: The capacity of systematic codes of minimum distance $d$ is*

$$\mathsf{cap}_{\mathrm{sys}}(d) = \begin{cases} 1 & if\ d = O(n), \\ 1-\epsilon & if\ d = \Theta(n^{1+\epsilon})\ with\ 0 < \epsilon < 1, \\ 0 & if\ d = \Theta(n^2). \end{cases}$$

*Proof:* Since systematic codes are a special case of general error-correcting codes, we naturally have

$$\mathsf{cap}_{\mathrm{sys}}(d) \leqslant \mathsf{cap}(d).$$

Thus, to prove the claim, all that remains is to prove the other direction of the inequality.

According to Theorem 16, there exists an $[n, k, d]$ systematic code if $k$ is the maximum integer that satisfies

$$d\binom{k+d}{d}\binom{d+n-k}{n-k}2^n < \frac{n!}{k!}. \tag{20}$$

That is because

$$\begin{aligned}&\sum_{i=1}^{d-1}\binom{k+i-2}{i}\binom{d-i-1+n-k}{n-k}2^{\min(d-i-1,n-k)} \\ &\quad\leqslant d\binom{k+d}{d}\binom{d+n-k}{n-k}2^n,\end{aligned}$$

for all $n > k \geqslant 2$ and $d \geqslant 2$.

For such $k$, we have $k(n, d) \geqslant k$. For convenience, we define the constant

$$\alpha = \liminf_{n\to\infty}\frac{k}{n}.$$

We also recall the well-known Stirling's approximation [28],

$$\ln(m!) = m\ln m - O(m).$$

Thus, if $\alpha > 0$,

$$\begin{aligned}\mathsf{cap}_{\mathrm{sys}}(d) &\geqslant \liminf_{n\to\infty}\frac{\ln k(n,d)!}{\ln n!} \geqslant \liminf_{n\to\infty}\frac{\ln k!}{\ln n!} \\ &= \lim_{n\to\infty}\frac{\alpha n\ln(\alpha n) - O(n)}{n\ln n - O(n)} = \alpha.\end{aligned}$$

To prove the final conclusion, we will show that

$$\alpha \geqslant \begin{cases} 1 & \text{if } d = O(n), \\ 1-\epsilon & \text{if } d = \Theta(n^{1+\epsilon}), \\ 0 & \text{if } d = \Theta(n^2). \end{cases} \tag{21}$$

We note that the last case is trivial, and so we only have to prove the first two.

By our choice of $k$ and by (20), we have

$$d\binom{k+d+1}{d}\binom{d+n-k-1}{n-k-1}2^n \geqslant \frac{n!}{(k+1)!}.$$

It suffices now to consider the subsequence of $k$ such that $\lim_{n\to\infty} k/n = \alpha$. It follows that

$$\lim_{n\to\infty} \frac{\ln\left(d\binom{k+d+1}{d}\binom{d+n-k-1}{n-k-1}2^n\right)}{\ln\left(\frac{n!}{(k+1)!}\right)} \geqslant 1. \tag{22}$$

To prove the first case of (21) assume $d = O(n)$. Again, by Stirling's approximation, (22) becomes,

$$\begin{aligned} 1 &\leqslant \lim_{n\to\infty} \frac{\ln\left(d\binom{k+d+1}{d}\binom{d+n-k-1}{n-k-1}2^n\right)}{\ln\left(\frac{n!}{(k+1)!}\right)} \\ &\leqslant \lim_{n\to\infty} \frac{\ln\left(d\cdot 2^{k+d+1}\cdot 2^{d+n-k-1}\cdot 2^n\right)}{\ln(n!)-\ln(k!)-\ln(k+1)} \\ &= \lim_{n\to\infty} \frac{O(n)}{n\ln n - \alpha n\ln(\alpha n) - O(n)}. \end{aligned}$$

Since $\alpha$ is a constant, we must therefore have $\alpha = 1$.

For the second case, assume $d = \Theta(n^{1+\epsilon})$ for $0 < \epsilon < 1$. We observe that for a large-enough $d$,

$$\begin{aligned} &\binom{k+d+1}{d}\binom{d+n-k-1}{n-k-1} \\ &\quad\leqslant \frac{(k+d+1)^{k+1}(d+n-k-1)^{n-k-1}}{(k+1)!(n-k-1)!} \\ &\quad\leqslant \frac{(2d)^{k+1}(2d)^{n-k-1}}{(k+1)!(n-k-1)!} \\ &\quad= \frac{(2d)^n}{(k+1)!(n-k-1)!}. \end{aligned}$$

By this observation, and by applying Stirling's approximation to (22), we get

$$\begin{aligned} 1 &\leqslant \lim_{n\to\infty} \frac{\ln\left(d\binom{k+d+1}{d}\binom{d+n-k-1}{n-k-1}2^n\right)}{\ln\left(\frac{n!}{(k+1)!}\right)} \\ &\leqslant \lim_{n\to\infty} \frac{n\ln(2d) - \ln\left((k+1)!(n-k-1)!\right)}{\ln\left(\frac{n!}{(k+1)!}\right)} \\ &= \lim_{n\to\infty} \frac{(1+\epsilon)n\ln n - n\ln n + O(n)}{n\ln n - k\ln k + O(n)} \\ &= \lim_{n\to\infty} \frac{\epsilon n\ln n - O(n)}{(1-\alpha)n\ln n - O(n)}. \end{aligned}$$

Thus, $\alpha \geqslant 1-\epsilon$, as we wanted to show. ■

## IV. Systematic Codes in the $\ell_\infty$-Metric

We recall that the definition of systematic codes in the $\ell_\infty$-metric differs from that in Kendall's $\tau$-metric. In an $[n, k, d]$ systematic code in the $\ell_\infty$-metric, when taking the first $k$ *coordinates* of the $k!$ codewords and relabeling the surviving $k$ elements to $[k]$, we obtain every permutation of $S_k$ exactly once.

The exact capacity for codes in the $\ell_\infty$-metric is unknown. There is a large gap between the lower and upper bounds on the size of optimal codes, mainly due to the lack of an asymptotic expression for the size of balls in this metric. Thus, to evaluate the parameters of our construction we will compare the rate of the constructed systematic codes with that of known general codes. Given an $(n, M, d)$ code $C$ in the $\ell_\infty$-metric, its *rate* is defined as (see [29])

$$R(C) = \frac{\log_2 M}{n}.$$

Note that this definition is somewhat different than that for Kendall's $\tau$-metric (see [2]). The reason for the difference is the fact that for a fixed normalized distance $\delta$, the size of the maximal code is only exponential in $n$ (see [29]). If we were to use the definition of rate as in Kendall's $\tau$-metric, then the resulting rate of all codes would be 0.

We present two constructions for systematic codes, where the first is adequate for distances $d = O(1)$, and where the second is intended for the $d = \Theta(n)$ case.

*Construction E: Let $1 \leqslant d \leqslant n$ be positive integers, and let $1 \leqslant k \leqslant \lceil n/d \rceil$ be an integer as well. We denote*

$$A_{k,d} = \{1, 1+d, 1+2d, \ldots, 1+(k-1)d\}.$$

*We construct the code*

$$\begin{aligned} C = \{(f_1, \ldots, f_n) \in S_n \,|\, &\{f_1, \ldots, f_k\} = A_{k,d}, \\ &f_{k+1} < f_{k+2} < \cdots < f_n\} \end{aligned}$$

□

*Theorem 21: The code C from Construction E is an $[n, k, d]$ systematic code in the $\ell_\infty$-metric.*

*Proof:* It is immediately evident that $|C| = k!$. Furthermore, every two distinct codewords in $C$ disagree on at least one of the first $k$ coordinates. Since all entries in the first $k$ coordinates leave a residue of 1 modulo $d$, the $\ell_\infty$-distance between distinct codewords is at least $d$. Finally, the projection onto the first $k$ coordinates is easily seen to provide all possible permutations from $S_k$ exactly once. ■

The optimal choice of $k$ in Construction E is obviously $k = \lceil n/d \rceil$, and it provides a code of size $\lceil n/d \rceil!$. This can be compared with Construction 1 of [29] which gives a code of size $(\lceil n/d \rceil!)^{n \bmod d}(\lfloor n/d \rfloor!)^{d-(n \bmod d)}$. If we denote the rate of the code from Construction E by $R$, and the rate of the code from Construction 1 of [29] by $R'$, then

$$\frac{R}{R'} = \frac{\log_2(\lceil n/d \rceil!)}{\log_2\left((\lceil n/d \rceil!)^{n \bmod d}(\lfloor n/d \rfloor!)^{d-(n \bmod d)}\right)} \geqslant \frac{1}{d}.$$

Encoding and decoding procedures for the code from Construction E is quite simple. We note that, up to a reordering of the coordinates, the codewords of $C$ from Construction E

are also codewords of the code from [29]. Thus, the simple decoding procedure from [29] will suffice: given a received $(f_1, \ldots, f_n) \in S_n$, we ignore $f_{k+1}, \ldots, f_n$, and for all $i \in [k]$, replace $f_i$ with the closest element from $A_{k,d}$. Encoding is equally simple: in the spirit of [29], we translate an integer from $\mathbb{Z}_{k!}$ to a permutation over the elements of $A_{k,d}$ (see for example [6], [22]), and place the remaining elements of $[n]\backslash A_{k,d}$ in ascending order. Thus, encoding and decoding for the code of Construction E has the same complexity as the corresponding procedures from [29].

We now turn to provide a construction suited for $d = \Theta(n)$.

*Construction F: Let $1 \leqslant d \leqslant n$ be positive integers. We recall Construction 1 from [29], of an $(n, M, d)$ code,*

$$C' = \{f \in S_n \mid f(i) \equiv i \pmod{d}, \text{ for all } i \in [n]\},$$

*where*

$$M = \left|C'\right| = (\lceil n/d \rceil !)^{n \bmod d} (\lfloor n/d \rfloor !)^{d-(n \bmod d)}.$$

*Let $k$ be the largest integer such that $k! \leqslant \left|C'\right|$, and let $C''$ be the set of all permutations over the set $\{n+1, n+2, \ldots, n+k\}$. Assume*

$$C' = \left\{f'_1, f'_2, \ldots, f'_{|C'|}\right\}$$
$$C'' = \left\{f''_1, f''_2, \ldots, f''_{k!}\right\}.$$

*We now construct the code*

$$C = \left\{f''_i \| f'_i \mid 1 \leqslant i \leqslant k!\right\},$$

*where $\|$ denotes vector concatenation.* □

*Theorem 22: The code $C$ from Construction F is an $[n+k, k, d]$ systematic code in the $\ell_\infty$-metric.*

*Proof:* The code is obviously of size $k!$, and by construction, the projection onto the first $k$ coordinates gives all possible permutations exactly once. Since $C'$ is a code with minimum distance $d$ (see [29] for proof), the code $C$ also has minimum distance of $d$ in the $\ell_\infty$-metric. ■

We now turn to analyze the asymptotic rate of the code from Construction F. Assume $d = \delta n$, where $\delta$ is a constant, $0 < \delta < 1$. By our choice of $k$, we have

$$\frac{1}{k+1}\left|C'\right| \leqslant |C| = \left|C''\right| = k! \leqslant \left|C'\right|. \tag{23}$$

Let $R$ denote the rate of $C$, and $R'$ denote the rate of $C'$, i.e.,

$$R = \frac{\log_2 k!}{n+k},$$
$$R' = \frac{\log_2 \left|C'\right|}{n}.$$

Thus, by (23),

$$\left(1 - \frac{\log_2(k+1)}{\log_2 |C'|}\right)\frac{n}{n+k} \leqslant \frac{R}{R'} \leqslant \frac{n}{n+k}.$$

Since $1 \leqslant k \leqslant n$, while (see [29])

$$\left|C'\right| \geqslant 2^{(1-\delta)n}$$

we have

$$\lim_{n\to\infty}\frac{R}{R'} = \lim_{n\to\infty}\frac{n}{n+k}.$$

At this point we need to bound $k$, and we contend that

$$k \leqslant \frac{n}{\log_2 \log_2 n}.$$

Let us assume, for $k = \lceil n/\log_2\log_2 n \rceil$, that we have

$$k! \leqslant \left|C'\right|.$$

We easily see that

$$\left|C'\right| \leqslant \left(\left\lceil \frac{n}{d} \right\rceil !\right)^d = \left(\left\lceil \frac{1}{\delta} \right\rceil !\right)^{\delta\lceil 1/\delta \rceil n} = \alpha^n$$

for some constant $\alpha > 1$.

On the other hand, we recall the well known bound

$$m! \geqslant \left(\frac{m}{e}\right)^m,$$

which holds for all positive integers $m$. Thus,

$$k! = \lceil n/\log_2\log_2 n \rceil ! \geqslant \left(\frac{n}{e\log_2\log_2 n}\right)^{\frac{n}{\log_2\log_2 n}}.$$

If indeed $k! \leqslant \left|C'\right|$ then necessarily

$$\left(\frac{n}{e\log_2\log_2 n}\right)^{\frac{n}{\log_2\log_2 n}} \leqslant \alpha^n,$$

and taking $\log_2$ of both sides gives us

$$\frac{n}{\log_2\log_2 n}\log_2\left(\frac{n}{e\log_2\log_2 n}\right) \leqslant n\log_2\alpha.$$

However, this last inequality certainly does not hold for large-enough $n$. It therefore follows that indeed

$$k \leqslant \frac{n}{\log_2\log_2 n}.$$

Finally,

$$\lim_{n\to\infty}\frac{R}{R'} = \lim_{n\to\infty}\frac{n}{n+k} \geqslant \lim_{n\to\infty}\frac{n}{n+\frac{n}{\log_2\log_2 n}} = 1.$$

Essentially, when $d = \Theta(n)$, we constructed systematic codes with the same asymptotic rate and minimum distance as the non-systematic codes appearing in [29], which are currently the best codes known asymptotically. Furthermore, the construction we presented can work with any other non-systematic error-correcting code, provided it has an exponential size when $d = \Theta(n)$.

## V. Conclusion

In this paper, we studied systematic error-correcting codes for rank modulation under two metrics: Kendall's $\tau$-metric, and the $\ell_\infty$-metric. In the former, we presented several constructions, and found the capacity of systematic codes. Efficient encoding and decoding schemes were also discussed. In the latter, two constructions were given, one of them asymptotically attaining the same rate as the best construction currently known in this metric.

Some open questions remain. In Kendall's $\tau$-metric we still do not know, given $n$ and $d$, what is the largest $[n, k, d]$ systematic code possible. We are also interested in the question of whether systematic perfect codes (or even general perfect

codes) exist. In the $\ell_\infty$-metric, we are still missing tight bounds, even asymptotically, on the parameters of general codes, as well as for systematic codes. Furthermore, the last construction lacks efficient encoding and decoding procedures.

## ACKNOWLEDGMENT

The authors would like to thank the associate editor and the anonymous reviewers, whose comments helped improve the presentation of the paper.

## REFERENCES

[1] M. Awasthi, M. Shevgoor, K. Sudan, B. Rajendran, R. Balasubramonian, and S. Viji, "Efficient scrub mechanisms for error-prone emerging memories," in *Proc. IEEE 18th Int. Symp. High Perform. Comput. Archit. (HPCA)*, New Orleans, LA, USA, Feb. 2012, pp. 1–12.

[2] A. Barg and A. Mazumdar, "Codes in permutations and error correction for rank modulation," *IEEE Trans. Inf. Theory*, vol. 56, no. 7, pp. 3158–3165, Jul. 2010.

[3] G. W. Burr *et al.*, "Phase change memory technology," *J. Vac. Sci. Technol. B*, vol. 28, no. 2, pp. 223–262, Mar. 2010.

[4] S. Buzaglo and T. Etzion. (Oct. 2013). "Perfect permutations codes with the Kendall's $\tau$-metric." [Online]. Available: http://arxiv.org/abs/1310.5515

[5] P. Cappelletti, C. Golla, P. Olivo, and E. Zanoni, *Flash Memories*. Norwell, MA, USA: Kluwer, 1999.

[6] T. M. Cover, "Enumerative source encoding," *IEEE Trans. Inf. Theory*, vol. 19, no. 1, pp. 73–77, Jan. 1973.

[7] F. Farnoud, V. Skachek, and O. Milenkovic, "Error-correction in flash memories via codes in the Ulam metric," *IEEE Trans. Inf. Theory*, vol. 59, no. 5, pp. 3003–3020, May 2013.

[8] E. Fujiwara, *Code Design for Dependable Systems: Theory and Practical Applications*. New York, NY, USA: Wiley, 2005.

[9] S. W. Golomb and L. R. Welch, "Perfect codes in the Lee metric and the packing of polyominoes," *SIAM J. Appl. Math.*, vol. 18, no. 2, pp. 302–317, Jan. 1970.

[10] G. H. Hardy and E. M. Wright, *An Introduction to the Theory of Numbers*, 6th ed. London, U.K.: Oxford Univ. Press, 2008.

[11] A. Jiang, H. Li, and Y. Wang, "Error scrubbing codes for flash memories," in *Proc. 11th Can. Workshop Inf. Theory (CWIT)*, Ottawa, ON, Canada, May 2009, pp. 32–35.

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

[13] A. Jiang, M. Schwartz, and J. Bruck, "Correcting charge-constrained errors in the rank-modulation scheme," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2112–2120, May 2010.

[14] M. Kendall and J. D. Gibbons, *Rank Correlation Methods*. New York, NY, USA: Oxford Univ. Press, 1990.

[15] T. Kløve, "Generating functions for the number of permutations with limited displacement," *Electron. J. Combinat.*, vol. 16, no. 1, pp. 1–11, 2009.

[16] T. Kløve, "Lower bounds on the size of spheres of permutations under the Chebychev distance," *Designs, Codes Cryptograph.*, vol. 59, nos. 1–3, pp. 183–191, 2011.

[17] T. Kløve, T.-T. Lin, S.-C. Tsai, and W.-G. Tzeng, "Permutation arrays under the Chebyshev distance," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2611–2617, Jun. 2010.

[18] D. E. Knuth, *The Art of Computer Programming: Fundemental Algorithms*, vol. 1, 3rd ed. Reading, MA, USA: Addison-Wesley, 1997.

[19] D. E. Knuth, *The Art of Computer Programming: Sorting and Searching*, vol. 3, 2nd ed. Reading, MA, USA: Addison-Wesley, 1998.

[20] J. Leech and N. J. A. Sloane, "Sphere packings and error-correcting codes," *Can. J. Math.*, vol. 23, no. 4, pp. 718–745, 1971.

[21] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*. Amsterdam, The Netherlands: North Holland, 1978.

[22] M. Mareš and M. Straka, "Linear-time ranking of permutations," in *Algorithms-ESA*. Berlin, Germany: Springer-Verlag, 2007, pp. 187–193.

[23] A. Mazumdar, A. Barg, and G. Zémor, "Constructions of rank modulation codes," *IEEE Trans. Inf. Theory*, vol. 59, no. 2, pp. 1018–1029, Feb. 2013.

[24] R. M. Roth and P. H. Siegel, "Lee-metric BCH codes and their application to constrained and partial-response channels," *IEEE Trans. Inf. Theory*, vol. 40, no. 4, pp. 1083–1096, Jul. 1994.

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

[26] M. Schwartz, "Efficiently computing the permanent and Hafnian of some banded Toeplitz matrices," *Linear Algebra Appl.*, vol. 430, no. 4, pp. 1364–1374, Feb. 2009.

[27] M. Schwartz and I. Tamo, "Optimal permutation anticodes with the infinity norm via permanents of (0, 1)-matrices," *J. Combinat. Theory, Ser. A*, vol. 118, no. 6, pp. 1761–1774, 2011.

[28] J. Stirling, *Methodus Differentialis: Sive Tractatus de Summatione Et Interpolatione Serierum Infinitarum*. London, U.K.: Gul, 1730.

[29] I. Tamo and M. Schwartz, "Correcting limited-magnitude errors in the rank-modulation scheme," *IEEE Trans. Inf. Theory*, vol. 56, no. 6, pp. 2551–2560, Jun. 2010.

[30] I. Tamo and M. Schwartz, "On the labeling problem of permutation group codes under the infinity metric," *IEEE Trans. Inf. Theory*, vol. 58, no. 10, pp. 6595–6604, Oct. 2012.

**Hongchao Zhou** received the B.Sc. degree in physics and mathematics and M.Sc. degree in control science and engineering from Tsinghua University, Beijing, China, in 2006 and 2008, respectively, and the M.Sc. degree and Ph.D. degree in electrical engineering from California Institute of Technology, Pasadena, CA, in 2009 and 2012, respectively.

He is a postdoctoral researcher in the Research Laboratory of Electronics at Massachusetts Institute of Technology, Cambridge. His current interests include information theory and randomness, data storage, information security, and stochastic biological networks. He is a recipient of the 2013 Charles Wilts Prize for the best doctoral thesis in electrical engineering at the California Institute of Technology.

**Moshe Schwartz** (M'03–SM'10) received the B.A. (summa cum laude), M.Sc., and Ph.D. degrees from the Technion - Israel Institute of Technology, Haifa, Israel, in 1997, 1998, and 2004 respectively, all from the Computer Science Department.

He was a Fulbright post-doctoral researcher in the Department of Electrical and Computer Engineering, University of California San Diego, and a post-doctoral researcher in the Department of Electrical Engineering, California Institute of Technology. He now holds a position with the Department of Electrical and Computer Engineering, Ben-Gurion University of the Negev, Israel. While on sabbatical, he was a visiting scientist at the Massachusetts Institute of Technology.

He received the 2009 IEEE Communications Society Best Paper Award in Signal Processing and Coding for Data Storage, and the 2010 IEEE Communications Society Best Student Paper Award in Signal Processing and Coding for Data Storage. His research interests include algebraic coding, combinatorial structures, and digital sequences.

**Anxiao (Andrew) Jiang** (S'00–M'05–SM'12) received the B.Sc. degree in electronic engineering from Tsinghua University, Beijing, China in 1999, and the M.Sc. and Ph.D. degrees in electrical engineering from the California Institute of Technology, Pasadena, California in 2000 and 2004, respectively.

He is currently an Associate Professor in the Computer Science and Engineering Department and the Electrical and Computer Engineering Department at Texas A&M University in College Station, Texas. He has been a visiting professor at California Institute of Technology, University of California in San Diego, Ecole Polytechnique Federale de Lausanne (EPFL) and Singapore Data Storage Institute (DSI), and a consulting researcher at HP Labs, EMC and Microsoft Research. His research interests include information theory, data storage, networks and algorithm design.

Prof. Jiang is a recipient of the NSF CAREER Award in 2008 for his research on information theory for flash memories and a recipient of the 2009 IEEE Communications Society Best Paper Award in Signal Processing and Coding for Data Storage.

**Jehoshua Bruck** (S'86–M'89–SM'93–F'01) is the Gordon and Betty Moore Professor of computation and neural systems and electrical engineering at the California Institute of Technology (Caltech). His current research interests include information theory and systems and the theory of computation in nature.

Dr. Bruck received the B.Sc. and M.Sc. degrees in electrical engineering from the Technion-Israel Institute of Technology, in 1982 and 1985, respectively, and the Ph.D. degree in electrical engineering from Stanford University, in 1989. His industrial and entrepreneurial experiences include working with IBM Research where he participated in the design and implementation of the first IBM parallel computer; cofounding and serving as Chairman of Rainfinity (acquired in 2005 by EMC), a spin-off company from Caltech that created the first virtualization solution for Network Attached Storage; as well as cofounding and serving as Chairman of XtremIO (acquired in 2012 by EMC), a start-up company that created the first scalable all-flash enterprise storage system.

Dr. Bruck is a recipient of the Feynman Prize for Excellence in Teaching, the Sloan Research Fellowship, the National Science Foundation Young Investigator Award, the IBM Outstanding Innovation Award and the IBM Outstanding Technical Achievement Award.