Beating the probabilistic lower bound on perfect hashing[*]

Chaoping Xing[†] Chen Yuan[‡]

Abstract

For an integer $q \geqslant 2$, a perfect q-hash code C is a block code over $[q] := \{1, \ldots, q\}$ of length n in which every subset $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_q\}$ of q elements is separated, i.e., there exists $i \in [n]$ such that $\{\mathrm{proj}_i(\mathbf{c}_1), \ldots, \mathrm{proj}_i(\mathbf{c}_q)\} = [q]$, where $\mathrm{proj}_i(\mathbf{c}_j)$ denotes the ith position of $\mathbf{c}_j$. Finding the maximum size $M(n, q)$ of perfect q-hash codes of length n, for given q and n, is a fundamental problem in combinatorics, information theory, and computer science. In this paper, we are interested in asymptotical behavior of this problem. More precisely speaking, we will focus on the quantity $R_q := \limsup_{n\to\infty} \frac{\log_2 M(n,q)}{n}$.

A well-known probabilistic argument indicates $R_q \geqslant \frac{1}{q-1} \log_2 \left(\frac{1}{1-q!/q^q}\right)$ [10, 12]. This is still the best-known lower bound so far except for the case $q = 3$ for which Körner and Matron [13] found that the concatenation technique could lead to perfect 3-hash codes that could beat this probabilistic lower bound. This improved lower bound on R_3 was discovered in 1988 and there has been no progress of this lower bound on R_q for more than 30 years despite of some work on upper bounds on R_q. In this paper we show that this probabilistic lower bound can be improved for $q = 4, 8$ and all odd integers between 5 and 25,[1] and *all sufficiently large* q with $q \pmod 4 \neq 2$. Although we are not able to prove that our construction can beat the probabilistic method for all q with $q \pmod 4 \neq 2$, the fact that our construction beat the probabilistic method for both small and large q sheds light on that our new construction might beat the previous lower bound for all q with $q \pmod 4 \neq 2$. Our idea is based on a modified concatenation differing from the concatenation [10] where both the inner and outer codes are separated. In our concatenation, the inner code is not necessarily a perfect q-hash code. This gives a more flexible choice of inner codes and hence we are able to improve the lower bound on R_q.

1 Introduction

Probabilistic method is one of the most powerful tools to derive lower bounds in theoretical computer science and extremal combinatorics [1]. Roughly speaking, to prove the existence of an object of a given size satisfying certain conditions, one shows that a random object of this size (maybe after being slightly modified) has a positive probability to satisfy these conditions. In many problems the lower bound given by this method is conjectured exact, at least asymptotically, and sometimes one can prove it is indeed so. This means that optimal solutions to such problems are rather common. On the other hand, when the probabilistic lower bound is not asymptotically exact, optimal solutions tend to be rare and have some particular structure. So, from a theoretical point of view, it is of great importance to know whether a problem belongs to one or the other of these two classes. Some exceptional examples where the probabilistic lower bounds are not asymptotically exact include the Gilbert-Varshamov bound in coding theory [16] and the probabilistic lower bound on perfect hash codes [13]. In this paper, we study lower bounds on perfect hash codes and compare them with the probabilistic lower bound.

A q-ary code $C \subseteq [q]^n$ is said to be a perfect q-hash code if for every subset of C of q elements, there exists an coordinate where the q codewords in this subset have distinct values. The rate of a perfect q-hash code is defined as $R_C = \frac{\log_2 |C|}{n}$.

The existence of a perfect q-hash code gives rise to a perfect q-hash family. To see this, let C be the whole universe and the projection of each coordinate be a hash function. Then, for any q elements of this universe, there exists a hash function mapping them to distinct values. Another application of perfect q-hash code is the zero-error list decoding on certain channels. A channel can be thought of as a bipartite graph $(V; W; E)$, where V is the set of channel inputs, W is the set of channel outputs, and $(w, v) \in E$ if on input v, the channel can

[*]The research of C. Xing is in part supported by NSFC grant 12031011 and the Huawei-SJTU joint project. This work was done while C. Yuan was at Centrum Wiskunde & Informatica. C. Yuan has been supported by ERC ADG project No 74079 (ALGSTRONGCRYPTO).

[†]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China. E-mail:xingcp@sjtu.edu.cn.

[‡]School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China and Centrum Wiskunde & Informatica, Amsterdam, the Netherlands. Email:Chen.Yuan@cwi.nl

[1]The case $q = 3$ is resolved in [13].

output w. The $q/(q-1)$ channel then is the channel with $V = W = \{0, 1, \ldots, q-1\}$, and $(v, w) \in E$ if and only if $v \neq w$. If we want to ensure that the receiver can identify a subset of at most $q-1$ sequences that is guaranteed to contain the transmitted sequence, one can communicate via n repeated uses of the channel using the perfect q-hash code. See [9, 7] for more details.

In this paper, we only consider the asymptotic behavior of rates of perfect q-hash codes, namely, we focus on the quantity $R_q := \limsup_{n\to\infty} \frac{\log_2 M(n,q)}{n}$, where $M(n, q)$ stands for the maximum size of perfect q-hash codes of length n.

The study of R_q could be dated back to 80s. There are a few works dedicated to the upper bound on R_q. Fredman and Komlós [10] showed a general upper bound: $R_q \leqslant \frac{q!}{q^{q-1}}$ for all $q \geqslant 2$. Arikan [2] improved this bound for $q = 4$, and then Dalai, Guruswami and Radhakrishnan [7] further improved this upper bound. Recently, Guruswami and Riazanov [11] discovered a stronger upper bound for every $q \geq 4$. However, they are only able to explicitly compute this bound for $q = 5, 6$. Costa and Dalai [6] further show how to explicitly compute this improvement over the previous upper bound. Although there are some works towards tightening the upper bound on R_q, very few results about lower bounds on R_q are known. A plain probabilistic argument shows the existence of perfect q-hash code with rate $R_q \geqslant \frac{1}{q-1}\log_2\left(\frac{1}{1-q!/q^q}\right)$ [10, 12]. This is still the best-known lower bound till now except for the case $q = 3$ for which Körner and Matron [13] found that the concatenation technique could lead to perfect 3-hash codes beating this the probabilistic lower bound. The improvement on the lower bound on R_3 was discovered in 1988 and there has been no progress for more than 30 years. Körner and Matron's idea is to concatenate an outer code, an 9-ary 3-hash code with an inner code, a perfect 3-hash code with size 9. They further posed an open problem whether there exist perfect q-hash codes beating the random argument for every q. In this paper, we provide a partial and affirmative answer to this open problem. We show that there exist perfect q-hash codes beating the random argument for all sufficiently large q with $q \pmod 4 \neq 2$. To complement this result, we also prove the existence of perfect q-hash code that could beat random result for small q such as $q = 4, 8$ and all odd integers q between 3 and 25, as well as many other odd integers between 27 and 155 (see Remark 3.2). Our computer search result together with asymptotic result suggests that our construction might beat the the probabilistic lower bound for every integer q with $q \pmod 4 \neq 2$.

The main technique of this paper is a modified version of concatenation. Unlike Körner and Matron's concatenation where both inner and outer codes must be separated, we abandon this separateness of inner code at a cost of imposing a stronger constraint on the outer code. By relaxing the condition that the inner code is perfect q-hash code, we have more freedom to construct the inner code. As a result, we are able to improve the lower bound on R_q.

Before explaining our technique in detail, let us recall the concatenation technique introduced by Körner and Matron. A straightforward probabilistic argument can prove the existence of an m-ary outer code C_1 of length n_1 that is q-separated with $q \leqslant m$, i.e., for every q-element subset of C_1 (a q-element set is a set of size q), there exists $i \in \{1, 2, \ldots, n_1\}$ such that elements of this q-subset are pairwise distinct at position i. Then, they construct a perfect q-hash code C_2 of length n_2 as an inner code. By concatenating C_1 with C_2 (see Lemma 2.2 for detail), they obtain a perfect q-hash code of length $n_1 n_2$.

In our concatenation, we make a trade-off between inner code and outer code by relaxing the condition on the inner code and imposing a stronger condition on the outer code. We first take a set $\mathcal{A}$ consisting of some q-element subsets of $[m]$. We apply the probabilistic method to show the existence of an m-ary outer code C_1 such that, for every q-element subset $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_q\}$ of C_1, there exists i such that $\{\mathrm{proj}_i(\mathbf{c}_1), \mathrm{proj}_i(\mathbf{c}_2), \ldots, \mathrm{proj}_i(\mathbf{c}_q)\} \in \mathcal{A}$, where $\mathrm{proj}_i(\mathbf{c}_j)$ stands for the ith coordinate of $\mathbf{c}_j$. Note that Körner and Matron's concatenation only requires that there exists i such that $\{\mathrm{proj}_i(\mathbf{c}_1), \mathrm{proj}_i(\mathbf{c}_2), \ldots, \mathrm{proj}_i(\mathbf{c}_q)\}$ are pairwise distinct. In this sense, we extend their idea by confining $\{\mathrm{proj}_i(\mathbf{c}_1), \mathrm{proj}_i(\mathbf{c}_2), \ldots, \mathrm{proj}_i(\mathbf{c}_q)\}$ to be one of the subsets in $\mathcal{A}$. If we could find a q-ary inner code C_2 such that there are at least $|\mathcal{A}|$ q-element subsets of C_1 that are separated, concatenating these two codes leads a perfect q-hash code. Now, it remains to look for suitable inner code C_2. One good candidate for the inner code is the MDS codes. In this paper, we first choose an q-ary MDS code of length 3 and dimension 2 over an abelian group to be the inner code C_2. To compare our concatenation with the known lower bound, we have to estimate the number of separated q-element subsets of C_2. We then reduce determining the number of separated q-element subsets of C_2 to determining the number of q-element subsets of C_2 in which all three positions are separated. It turns out that the latter problem is equivalent to the following well-known combinatorial problem: determine the number s_q of pairs (π_1, π_2) of bijections $[q] \to \mathbb{Z}_q$ such that $\pi_1 + \pi_2$ is a bijection of $\mathbb{Z}_q$ as well. In literature, there is an asymptotic result on

s_q for odd number q [15] which can be used to estimate the number of separated q-element subsets of C_2. As a result, we are able to improve R_q for large odd q. Recently, this combinatorial problem is further extended to abelian group G with $\sum_{x\in G} x = 0$ [8]. Due to this result, we can also extend our result to improve R_q for every large q with $q \pmod 4 \neq 2$. When it comes to the case where q is small, the numerical result shows that this $[3,2]$-MDS code still leads to an improved lower bound on R_q.

We further extend this $[3,2]$-MDS code result to a $[4,2]$-MDS code. It turns out that an q-ary MDS code over an abelian group of length 4 and dimension 2 could lead to an even better lower bound on R_q. Our main result is summarized below.

THEOREM 1.1. *For every integer q with $q \pmod 4 \neq 2$, one has a lower bound*

$$R_q \geqslant -\frac{1}{4(q-1)} \log_2 \left((1-\frac{q!}{q^q})^4 - (\frac{3q}{\sqrt{e}} + o(q))(\frac{q!}{q^q})^3 \right).$$

This rate outperforms the probabilistic lower bound, $R_q \geqslant -\frac{1}{(q-1)} \log_2(1-\frac{q!}{q^q})$, for all sufficiently large q with $q \pmod 4 \neq 2$.

We note that the numerical results imply that the same construction also beat the probabilistic lower bound for small q. This leads to the following conjecture.

CONJECTURE 1.1. *For every integer q with $q \pmod 4 \neq 2$, there exists a perfect q-hash code beating the probabilistic lower bound. Moreover, such construction can be obtained via a concatenation code defined in Theorem 3.3 and Theorem 3.4.*

This paper is organized as follows. In Section 2, we propose a new concatenation technique and derive a lower bound on R_q in terms of the number of separated q-element subsets of the inner code. In Section 3, we provide several candidates for the inner code of our concatenation technique and estimate the number of separated q-element subsets for these candidates. By plugging this number into the lower bound in Section 2, we manage to prove that the probabilistic lower bound on R_q can be improved in many cases.

2 $\mathcal{A}$-friendly codes and concatenation

2.1 Hash code A set containing q elements is called a q-element set. Assume that $m \geqslant q$, then a q-element subset $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_q\}$ of $[m]^N$ is called separated if there exists $i \in [N]$ such that $\mathrm{proj}_i(\mathbf{c}_1), \ldots, \mathrm{proj}_i(\mathbf{c}_q)$ are pairwise distinct. If q is a prime power, we denote by $\mathbb{F}_q$ the finite field with q elements and let $\mathbb{Z}_m := \mathbb{Z}/m\mathbb{Z}$ be a congruence class of integers modulo m.

A subset C of $[m]^N$ is called an m-ary code of length N. For an integer $q \leqslant m$, an m-ary code C of length N is called an m-ary q-hash code if every q-element subset of C is separated. In particular, we say that C is a perfect q-hash code if $m = q$.

Let us generalize the notion of an m-ary q-hash code. Let $\binom{[m]}{q}$ denote the collection of all q-element subsets of $[m]$. Let $\mathcal{A}$ be a subset of $\binom{[m]}{q}$ and let C be a code in $[m]^N$. We say that a q-element subset $\{\mathbf{c}_1, \ldots, \mathbf{c}_q\}$ of $[m]^N$ is $\mathcal{A}$-friendly if there exists $i \in [N]$ such that $\{\mathrm{proj}_i(\mathbf{c}_1), \mathrm{proj}_i(\mathbf{c}_2), \ldots, \mathrm{proj}_i(\mathbf{c}_q)\} \in \mathcal{A}$. Otherwise, we call $\{\mathbf{c}_1, \ldots, \mathbf{c}_q\}$ an $\mathcal{A}$-unfriendly subset. If every q-element subset of C is $\mathcal{A}$-friendly, we say that C is an $\mathcal{A}$-friendly code. In particular, this definition coincides with an m-ary q-hash code when $\mathcal{A} = \binom{[m]}{q}$.

2.2 Random $\mathcal{A}$-friendly codes In this subsection, by applying a probabilistic argument, we prove the existence of $\mathcal{A}$-friendly codes.

LEMMA 2.1. *Let $\mathcal{A}$ be a nonempty subset of $\binom{[m]}{q}$. Then there exists an m-ary $\mathcal{A}$-friendly code C of length N and size at least $\lceil \frac{M}{3} \rceil$ as long as*

$$\binom{M}{q}\left(1 - \frac{q!|\mathcal{A}|}{m^q}\right)^N \leq \frac{M}{2q} \tag{2.1}$$

Proof. We sample M codewords $\mathbf{c}_1, \ldots, \mathbf{c}_M$ uniformly at random in $[m]^N$ with replacement. The number of collisions is negligible compared to M. To see this, let $X_{i,j}$ be the 0,1-random variable such that $X_{i,j} = 1$ if $\mathbf{c}_i = \mathbf{c}_j$ and $X_{i,j} = 0$ otherwise. It is clear $P[X_{i,j} = 1] = m^{-N}$. It follows that $E[\sum_{1\leq i<j\leq M} X_{i,j}] = \binom{M}{2} m^{-N} = o(M)$ due to the fact that $M = o(\sqrt{m^N})$. Next, we bound the number of $\mathcal{A}$-friendly q-element sets from these M codewords. Let us fix a q-element set $\{\mathbf{c}_1, \ldots, \mathbf{c}_q\}$ with $\mathbf{c}_i = (c_{i,1}, \ldots, c_{i,N})$. For any $j \in [n]$, the probability that $\{c_{1,j}, \ldots, c_{q,j}\} \in \mathcal{A}$ is $\frac{q!|\mathcal{A}|}{m^q}$ as $c_{i,j}$ is picked uniformly at random in $[m]$. It follows that the probability that $\{\mathbf{c}_1, \ldots, \mathbf{c}_q\}$ is not $\mathcal{A}$-friendly is $(1 - \frac{q!|\mathcal{A}|}{m^q})^N$. There are at most $\binom{M}{q}$ q-element sets from $\{\mathbf{c}_1, \ldots, \mathbf{c}_M\}$. By linearity of expectation, the expected number of $\mathcal{A}$-unfriendly q-element sets is at most $\binom{M}{q}\left(1 - \frac{q!|\mathcal{A}|}{m^q}\right)^N \leq \frac{M}{2q}$. Remove all the codewords that lie in any of these $\mathcal{A}$-unfriendly q-element sets. Then, we remove at most $q \times \frac{M}{2q} = \frac{M}{2}$ codewords. According to our previous argument, there are $o(M)$ collisions among these M codewords. Remove these $o(M)$ codewords and we obtain an $\mathcal{A}$-friendly code of size at least $\frac{M}{3}$. The desired result follows. □

REMARK 2.1. *Note that in [13], the set $\mathcal{A}$ is the collection of all q-element subsets of $[m]$. Thus, our random* *argument can be viewed as a generalization of the argument in [13]. This generalization allows us to relax the constraint on our inner code C_1, i.e., only $|\mathcal{A}|/\binom{m}{q}$ fraction of q-element sets of C_1 are separated.*

If we choose $m = q$ in Lemma 2.1, then $|\mathcal{A}| = 1$. We obtain a random construction of perfect q-hash codes.

COROLLARY 2.1. *Let $q \geqslant 2$. Then there exists perfect q-hash code of length N and size at least $\lceil \frac{M}{3} \rceil$ as long as*

$$\binom{M}{q}\left(1-\frac{q!}{q^q}\right)^N \le \frac{M}{2q}. \tag{2.2}$$

In particular, this perfect q-hash code has rate

$$\frac{\log_2 M}{N} = -\frac{1}{q-1}\log_2\left(1-\frac{q!}{q^q}\right)+\frac{O(1)}{N}. \tag{2.3}$$

Hence, we obtain a probabilistic lower bound

$$R_q \geqslant \frac{1}{q-1}\log_2\left(\frac{1}{1-q!/q^q}\right). \tag{2.4}$$

Proof. As $\binom{M}{q} \leqslant \frac{M^q}{q!}$, the following inequality

$$\frac{M^q}{q!}\left(1-\frac{q!}{q^q}\right)^N \le \frac{M}{2q} \tag{2.5}$$

implies the inequality (2.2). Choose M to be the largest integer satisfying the inequality (2.5) and consider the limit $\lim_{N\to\infty}\frac{\log_2 M}{N}$. The desired equality (2.3) follows. ∎

2.3 A concatenation technique Let C be a q-ary code of length n and size m. Denote by $\mathcal{S}(C)$ the collection of all q-element subsets of C that are separated.

LEMMA 2.2. *Let C be a q-ary code of length n and size m. Then one has*

$$R_q \geqslant -\frac{1}{(q-1)n}\log_2\left(1-\frac{q!|\mathcal{S}(C)|}{m^q}\right). \tag{2.6}$$

Proof. Let π be any bijection from C to $[m]$. Define $\mathcal{A} := \bigcup_{\{\mathbf{c}_1,\ldots,\mathbf{c}_q\}\in\mathcal{S}(C)}\{\pi(\mathbf{c}_1),\ldots,\pi(\mathbf{c}_q)\}$. It is clear that $\mathcal{A} \subseteq \binom{[m]}{q}$ and $|\mathcal{A}| = |\mathcal{S}(C)|$. Lemma 2.1 tells us that there exists an m-ary $\mathcal{A}$-friendly code C_1 of length n_1 with rate

$$R = -\frac{1}{(q-1)}\log_2\left(1-\frac{q!|\mathcal{A}|}{m^q}\right)+\frac{O(1)}{n_1}.$$

Let C_2 be the concatenation of C_1 with C, i.e., $C_2 := \{\pi^{-1}(\mathbf{c}) = (\pi^{-1}(c_1),\pi^{-1}(c_2),\ldots,\pi^{-1}(c_{n_1})) : \mathbf{c} = (c_1,c_2,\ldots,c_{n_1}) \in C_1\}$. Clearly, the rate of C_2 is $R = -\frac{1}{n(q-1)}\log_2(1-\frac{q!|\mathcal{A}|}{m^q})+\frac{O(1)}{n_1 n}$. It remains to show that C_2 is a perfect q-hash code.

Choose any q-element subset $\{\pi^{-1}(\mathbf{c}_1),\pi^{-1}(\mathbf{c}_2),\ldots,\pi^{-1}(\mathbf{c}_q)\}$ from C_2 with $\{\mathbf{c}_1,\mathbf{c}_2,\ldots,\mathbf{c}_q\}$ being a q-element subset of C_1. Since C_1 is $\mathcal{A}$-friendly, there exists $i \in [N]$ such that $\{\mathrm{proj}_i(\mathbf{c}_1), \mathrm{proj}_i(\mathbf{c}_2),\ldots,\mathrm{proj}_i(\mathbf{c}_q)\} \in \mathcal{A}$. This implies that $\{\pi^{-1}(\mathrm{proj}_i(\mathbf{c}_1)),\ldots,\pi^{-1}(\mathrm{proj}_i(\mathbf{c}_q))\} \in \mathcal{S}(C)$ and thus $\{\pi^{-1}(\mathbf{c}_1),\pi^{-1}(\mathbf{c}_2),\ldots,\pi^{-1}(\mathbf{c}_q)\}$ is separated. The desired result follows from the definition of perfect q-hash codes. ∎

REMARK 2.2. *Given a q-ary code C of length n, Lemma 2.2 tells us there must exist an outer code whose concatenation with C yields a perfect q-hash code with rate $-\frac{1}{n(q-1)}\log_2(1-\frac{q!|\mathcal{S}(C_2)|}{m^q})$. That means we only need to focus on finding good inner codes C with large subset $\mathcal{S}(C)$. In what follows, when we talk about concatenation, we only specify the inner code. The outer code is always given by Lemma 2.2.*

3 Beating probabilistic lower bound

By Lemma 2.2, to have a good lower bound on R_q, one needs to find a q-ary inner code C of length n such that $\mathcal{S}(C)$ has large size for fixed q, n and size $|C|$. However, determining (or even estimating) the size of $\mathcal{S}(C)$ for a given inner code C with dimension at least 2 seems very difficult. In this section, we estimate the size of $\mathcal{S}(C)$ for some classes of codes and show that these inner codes give lower bounds on R_q better than the probabilistic lower bound (2.4)

3.1 Lower bounds from linear codes To overcome the problem of estimating the size of $\mathcal{S}(C)$, we resort to linearity and dual distance of linear codes and narrow our target to linear codes with simple structure. In this subsection, we investigate a promising candidate for the inner code.

Let us recall some facts about linear codes. Let q be a prime power and let C be a q-ary $[n,k]$-linear code. A subset I of $[n]$ of size k is called an information set of C if every codeword $\mathbf{c} \in C$ is uniquely determined by $\mathbf{c}_I$, where $\mathbf{c}_I$ is the projection of $\mathbf{c}$ at I. In other words, let G be a generator of C, then a subset I of $[n]$ of size k is an information set of C if and only if G_I is a $k \times k$ invertible matrix, where G_I is the submatrix of G consisting of those columns of G indexed by $i \in I$.

LEMMA 3.1. *Let C be a q-ary $[n,k]$-linear code with dual distance $d^{\perp}$. Then for any subset J of $[n]$ with $|J| \leqslant d^{\perp}-1$, there exists an information set I such that $J \subseteq I$.*

Proof. Let G be a generator of C. As C has dual distance $d^\perp$, any $d^\perp - 1$ columns of G are linearly independent. Thus, the submatrix G_J has rank $|J|$. Hence, one can find a subset I of $[n]$ of size k such that $J \subseteq I$ and G_I has rank k. The proof is completed. ∎

Let F be a finite set of q elements. Let C be a q-ary code over F of length n. For each $i \in [n]$, define the set $\mathcal{A}_i = \{\{\mathbf{c}_1, \ldots, \mathbf{c}_q\} \subseteq C : \{\mathrm{proj}_i(\mathbf{c}_1)), \ldots, (\mathrm{proj}_i(\mathbf{c}_q))\} = F\}$.

Thus, we have $\mathcal{S}(C) = \cup_{i=1}^n \mathcal{A}_i$. For any subset T of $[n]$, we denote by $\mathcal{A}_T$ the set $\cap_{i\in T}\mathcal{A}_i$. Let A_i denote the number

$$A_i = \sum_{T\subseteq[n],|T|=i} |\mathcal{A}_T|. \tag{3.7}$$

LEMMA 3.2. *Let C be a q-ary $[n,k]$-linear code with dual distance $d^\perp$. Then $|\mathcal{S}(C)|$ equals*

$$\sum_{i=1}^{d^\perp-1}(-1)^{i-1}\binom{n}{i}q^{q(k-i)}(q!)^{i-1} + \sum_{i=d^\perp}^{n}(-1)^{i-1}A_i. \tag{3.8}$$

Proof. First we claim that for any $j \in [d^\perp - 1]$ and subset J of $[n]$ with $|J| = j$, we have $|\mathcal{A}_J| = q^{q(k-j)}(q!)^{j-1}$.

By Lemma 3.1, we can choose an information set $I \subseteq [n]$ that includes J. For any matrix M in $\mathbb{F}_q^{q\times k}$, by the definition of the information set, it suffices to determine M_I so as to fix M. Since I is an information set, there is a unique q-tuple $(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_q)$ such that

$$M = \begin{pmatrix} \mathbf{c}_1 \\ \mathbf{c}_2 \\ \vdots \\ \mathbf{c}_q \end{pmatrix}_I . \tag{3.9}$$

It is clear that $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_q\} \in \mathcal{A}_J$ if and only if every column of M_J is a permutation of $(0, \ldots, q-1)$. There are $(q!)^{|J|} = (q!)^j$ ways to pick M_J and $q^{q(|I|-|J|)} = q^{q(k-j)}$ ways to pick M_{I-J}. This gives $(q!)^j q^{q(k-j)}$ different q-tuples $(\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_q)$ with $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_q\} \in \mathcal{A}_J$. It follows that the number of q-element sets in $\mathcal{A}_J$ is $(q!)^{j-1}q^{q(k-j)}$.

By the inclusion-exclusion principle, we have

$$|\mathcal{S}(C)| = |\bigcup_{i=1}^n \mathcal{A}_i| = \sum_{i=1}^{d^\perp-1}(-1)^{i-1}\binom{n}{i}q^{q(k-i)}(q!)^{i-1} + \sum_{i=d^\perp}^{n}(-1)^{i-1}A_i.$$

This completes the proof. ∎

By the equality (3.8), we have

$$\begin{aligned} |\mathcal{S}(C)| &= \sum_{i=1}^{n}(-1)^{i-1}\binom{n}{i}q^{q(k-i)}(q!)^{i-1} \\ &\quad - \sum_{i=d^\perp}^{n}(-1)^{i-1}\binom{n}{i}q^{q(k-i)}(q!)^{i-1} + \sum_{i=d^\perp}^{n}(-1)^{i-1}A_i \\ &= \frac{q^{qk}}{q!}\left(1 - \left(1 - \frac{q!}{q^q}\right)^n\right) \\ &\quad - \sum_{i=d^\perp}^{n}(-1)^{i-1}\binom{n}{i}q^{q(k-i)}(q!)^{i-1} + \sum_{i=d^\perp}^{n}(-1)^{i-1}A_i. \end{aligned}$$

This implies

$$1 - \frac{q!|\mathcal{S}(C)|}{q^{qk}} = \left(1 - \frac{q!}{q^q}\right)^n + \sum_{i=d^\perp}^{n}(-1)^{i-1}\binom{n}{i}\left(\frac{q!}{q^q}\right)^i - \frac{q!}{q^{qk}}\sum_{i=d^\perp}^{n}(-1)^{i-1}A_i.$$

Hence, in order to beat the probabilistic lower bound, we need to verify the following inequality for an inner code $C = [n,k]_q$,

$$\sum_{i=d^\perp}^{n}(-1)^{i-1}\binom{n}{i}\left(\frac{q!}{q^q}\right)^i < \frac{q!}{q^{qk}}\sum_{i=d^\perp}^{n}(-1)^{i-1}A_i \tag{3.10}$$

Lemma 3.2 shows that computing $|\mathcal{S}(C)|$ is reduced to computing A_i for $i = d^\perp, d^\perp+1, \ldots, n$. However, if $d^\perp$ is too far from n, we have to compute many A_i. The simplest case is $d^\perp = n$ where we need to compute only A_n. In this case the dimension k is at least $n-1$. Therefore, let us consider $[n, n-1]$ MDS codes. On the other hand, when C has dimension $n-1$, we do not require that q is a prime power. Precisely speaking, we have the following result.

LEMMA 3.3. *Let $q \geqslant 2$ be an integer and let F be an abelian group of order q. Define the q-ary code $C = \{(x_1, \ldots, x_{n-1}, \sum_{i=1}^{n-1} x_i) : x_1, \ldots, x_{n-1} \in F\}$. Let A_n denote the cardinality of the set $\{\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_q\} \subseteq C : \{\mathrm{proj}_i(\mathbf{c}_1), \ldots, \mathrm{proj}_i(\mathbf{c}_q)\} = F \quad \forall i \in [n]\}$. Then $|\mathcal{S}(C)| = \frac{q^{q(n-1)}}{q!}\left(1 - \left(1 - \frac{q!}{q^q}\right)^n\right) - (-1)^{n-1}q^{-q}(q!)^{n-1} + (-1)^{n-1}A_n$.*

Proof. One can show that in this case, we have $A_i = q^{q(n-i)}(q!)^{i-1}$ for any $1 \leqslant i \leqslant n-1$. The desired result follows from the same arguments as in Lemma 3.2. ∎

COROLLARY 3.1. *Let $q \geqslant 2$ be an integer and let F be an abelian group of order q. Let A_n be the number given in Lemma 3.3. If*

$$(-1)^{n-1}A_n > (-1)^{n-1}\frac{(q!)^{n-1}}{q^q}, \tag{3.11}$$

Then there exist families of perfect q-hash codes over $\mathbb{F}_q$ with rate better than the probabilistic lower bound (2.4)*.*

Proof. Let C be the q-ary code defined in Lemma 3.3. Then

$$1 - \frac{q!|\mathcal{S}(C)|}{q^{q(n-1)}} = \left(1 - \frac{q!}{q^q}\right)^n + (-1)^{n-1}\left(\frac{q!}{q^q}\right)^n$$
$$-(-1)^{n-1}\frac{q!}{q^{q(n-1)}}A_n < \left(1 - \frac{q!}{q^q}\right)^n.$$

The desired result follows. □

If C is the code of length 3 over $\mathbb{Z}_q$ given in Lemma 3.3, i.e, $C = \{(x, y, x+y) : x, y \in \mathbb{Z}_q\}$, then determining A_3 given in Lemma 3.3 is actually reduced to the following well-known combinatorial problem: determining the number s_q of pairs (π_1, π_2) of bijections $[q] \to \mathbb{Z}_q$ such that $\pi_1 + \pi_2$ is a bijection as well. The relation between A_3 and s_q is $A_3 = \frac{s_q}{q!}$.

The number s_q has been studied somewhat extensively, but under a different guise [3, 5, 4, 17, 15]. It is in general very difficult to determine the exact value of s_q unless q is an even number for which $s_q = 0$. It has been conjectured in [17] that there exists two positive constant c_1 and c_2 such that $c_1^q(q!)^2 < s_q < c_2^q(q!)^2$ for all odd q. Various upper bounds are given [15]. To beat the probabilistic lower bound on R_q, we want to show $s_q > (\frac{q!^2}{q^q})$. That means, we are only interested in the lower bounds on s_q. A generic lower bound is $s_q \geqslant 3.246^q \times q!$ for all odd q. However, there is still a very big gap between this lower bound and the aforementioned conjecture. Fortunately, for sufficiently large q, one can pin down the exact value of c_1 and c_2. We postpone this discussion to the next subsection. On the other hands, there are various algorithms to numerically approximate s_q [14].

By taking exact value of s_q for all odd q between 3 and 25 from [14], we obtain the following result.

Corollary 3.2. *There exists a family of perfect q-hash codes over $\mathbb{Z}_q$ with rate better than the probabilistic lower bound* (2.4) *for all odd q between* 3 *and* 25.

Proof. By Corollary 3.1, it is sufficient to verify the inequality

$$\frac{s_q}{q!} > \frac{(q!)^2}{q^q} \tag{3.12}$$

for all odd q between 3 and 25. Taking the values of s_q from Table I of [14] gives the desired claim. □

Remark 3.1. *For completeness, we list the values of $A_3 = \frac{s_q}{q!}$ and $\frac{(q!)^2}{q^q}$ for odd $q \in [3, 25]$ in Table 1. We observe that the ratio A_3 over $\frac{(q!)^2}{q^q}$ grows slowly but monotonically. In fact, we will see that this ratio is asymptotically equal to $\frac{q}{\sqrt{e}}$ in our following discussion.*

$\mathbb{Z}_q$	$\mathbb{Z}_5$	$\mathbb{Z}_7$	$\mathbb{Z}_9$	$\mathbb{Z}_{11}$
A_3	15	133	2025	37851
$\frac{(q!)^2}{q^q}$	4.6	30.8	339.9	5584.6
Ratio	3.26	4.32	5.96	6.78
$\mathbb{Z}_q$	$\mathbb{Z}_{13}$	$\mathbb{Z}_{15}$	$\mathbb{Z}_{17}$	$\mathbb{Z}_{19}$
A_3	1.03×10^6	3.63×10^7	1.60×10^9	8.76×10^{10}
$\frac{(q!)^2}{q^q}$	1.28×10^5	3.90×10^6	$1.52 * 10^8$	7.47×10^9
Ratio	8.04	9.30	10.53	11.71

Table 1: The comparison between A_3 and $\frac{(q!)^2}{q^q}$ for small odd q.

Remark 3.2. *In literature, various algorithms were proposed to compute s_q for large odd q. Using these algorithms, for many odd q in the interval* $[27, 155]$*, estimation on s_q is given in [14]. One can verify from these estimation that the probabilistic lower bound* (2.4) *is improved for all odd integers q for which available values of s_q are given in [14].*

For even q, we have $s_q = 0$. Therefore, we cannot use the codes defined in Lemma 3.3. Instead, we can replace $\mathbb{Z}_q$ by $\mathbb{F}_q$ if q is a prime power.

Corollary 3.3. *There exists a family of perfect q-hash code over $\mathbb{F}_q$ with rate better than the probabilistic lower bound* (2.4) *for $q = 4, 8, 9$. Furthermore, the lower bound on R_9 given here is better than that in Corollary 3.2 and the probabilistic lower bound.*

Proof. Let C be a code with the form

$$C = \{(x, y, x+y) : x, y \in \mathbb{F}_q\}.$$

Let A_3 be defined in (3.7). With the help of computer search, we get the values A_3 of C: 8 for code over $\mathbb{F}_4$, 384 for code over $\mathbb{F}_8$ and 2241 for code over $\mathbb{F}_9$, respectively. We note the fact that A_3 from the code over $\mathbb{F}_9$ is 2241, while A_3 from the code over $\mathbb{Z}_9$ is 2025. It is straightforward to verify that the inequality $A_3 > \frac{(q!)^2}{q^q}$ is satisfied for $q = 4, 8$ and 9. □

Remark 3.3. *The lower bound on R_3 given in [13] is $R_3 \geqslant \frac{1}{4}\log_2\frac{9}{5}$. Let C be a ternary* $[4, 2]$*-MDS code. The computer search shows that $|\mathcal{S}(C)| = 84$. By Lemma 2.2, we also obtain the same lower bound $R_3 \geqslant \frac{1}{4}\log_2\frac{9}{5}$.*

This remark indicates that q-ary MDS codes of larger length sometimes leads to a better lower bound on R_q than q-ary MDS codes of length 3 and dimension 2. This is further confirmed by the following example for $q = 4$.

Corollary 3.4. *There exists a family of perfect 4-hash code over $\mathbb{F}_4$ with rate at least 0.049586. This is better than both the lower bound given in Corollary 3.3 and the probabilistic lower bound.*

Proof. Assume $\mathbb{F}_4 = \{0, 1, \alpha, \alpha + 1\}$. Consider a $[5, 2]$-MDS code:

$$C = \{(a, b, a + b, a\alpha + b, a(\alpha + 1) + b) : a, b \in \mathbb{F}_4\}.$$

By computer search, we find that there are 1100 out of $\binom{32}{4}$ 4-element subsets of C that are separated. Plugging it into Lemma 2.2, we obtain perfect 4-hash code with rate 0.049586. $\square$

3.2 Beating the probabilistic lower bound for large q For some odd integers q large than 25, there are also some known lower bounds on s_q [14]. By these lower bounds, we can verify that the probabilistic lower bound on R_q are improved for odd integers between 27 and 155. The computer search can only help for small values of q. To lower bound s_q for large q, we have to look for a lower bound with rigorous mathematical proof. Fortunately, a recent progress on asymptotic behavior of s_q is given in [15]. Recall that there is a conjecture saying that, for all odd q, the number s_q lies in between $c_1^q q!^2$ and $c_2^q q!^2$ for some constants c_1, c_2. This conjecture is recently confirmed in [15]. Moreover, they even close the gap by showing $c_1 = c_2 = \frac{1}{\sqrt{e}} + o(1)$.

Proposition 3.1. ([15]) *Let q be an odd integer. Then, the number s_q is $(\frac{1}{\sqrt{e}} + o(1))\frac{q!^3}{q^{q-1}}$, and hence A_3 defined in Lemma 3.3 is $(\frac{1}{\sqrt{e}} + o(1))\frac{q!^2}{q^{q-1}}$.*

Plugging A_3 in Proposition 3.1 into (3.8) and (2.6) gives the following theorem.

Theorem 3.1. *For every odd integer q, R_q is at least*

$$-\frac{1}{3(q-1)}\log_2\left(1-3\frac{q!}{q^q}+3\frac{(q!)^2}{q^{2q}}-\left(\frac{1}{\sqrt{e}}+o(1)\right)\frac{(q!)^3}{q^{3q-1}}\right).$$

Moreover, for every sufficiently large odd q this rate is bigger than that given by the probabilistic lower bound.

Proof. It remains to compare this rate with (2.3). It suffices to show that $A_3 > \frac{(q!)^2}{q^q}$. For large odd q, this inequality is reduced to prove $\left(\frac{1}{\sqrt{e}} + o(1)\right)\frac{(q!)^3}{q^{3q-1}} > \frac{(q!)^3}{q^{3q}}$. This holds as $\frac{1}{\sqrt{e}}+o(1) > \frac{1}{q}$ for sufficiently large q. $\square$

As $s_q = 0$ for even q, we have to replace group $\mathbb{Z}_q$ by other abelian groups of order q. Recently, Eberhard [8] extends Proposition 3.1 to any abelian group F with $\sum_{x\in F} x = 0$ and size q, i.e., determine the number s_F of pairs (π_1, π_2) of bijections $[q] \to F$ such that $\pi_1 + \pi_2$ is a bijection as well.

Proposition 3.2. ([8]) *Let F be an abelian group with $\sum_{x\in F} x = 0$ and size q. Then, the number s_F is $(\frac{1}{\sqrt{e}} + o(1))\frac{q!^3}{q^{q-1}}$.*

Theorem 3.2. *For every integer q with $q \pmod 4 = 0$, R_q is at least*

$$-\frac{1}{3(q-1)}\log_2\left(1-3\frac{q!}{q^q}+3\frac{(q!)^2}{q^{2q}}-\left(\frac{1}{\sqrt{e}}+o(1)\right)\frac{(q!)^3}{q^{3q-1}}\right).$$

Moreover, for every sufficiently large q with $q \pmod 4 = 0$, this rate is bigger than that given by the probabilistic lower bound.

Proof. Since $q \pmod 4 = 0$, let $q = 2^r p$ with an odd integer p and $r \geq 2$. Let $F = \mathbb{F}_{2^r} \times \mathbb{Z}_p$. It is clear that F is an abelian group and $\sum_{x\in F} x = 0$. Define the code $C := \{(x, y, x + y) : x, y \in F\}$. Then, C is an MDS code with dimension 2 and length 3. It remains to bound A_3. This is equivalent to counting the pair of bijections (π_1, π_2) $[q] \to F$ such that $\pi_1 + \pi_2$ is a bijection as well. Proposition 3.2 says that the number A_3 of C is $\frac{s_F}{q!} = (\frac{1}{\sqrt{e}}+o(1))\frac{q!^2}{q^{q-1}}$. Plugging A_3 into (3.8) and (2.6) gives the desired result. $\square$

3.3 A better lower bound The lower bounds given in Theorems 3.1 and 3.2 make use of linear codes over an abelian group of length 3 and dimension 2. As we have seen, there exists linear codes that yield better lower bound. In this section, we will see that this is indeed the case when we replace the $[3, 2]$ MDS code in 3.1 and 3.2 with the $[4, 2]$ MDS code.

Lemma 3.4. *Let $q \geqslant 3$ be an odd integer. Consider the code*

$$C = \{(x, y, x + y, x - y) : \ x, y \in \mathbb{Z}_q\}.$$

Then one has

$$|\mathcal{S}(C)| \geqslant \binom{4}{1}q^q - \binom{4}{2}q! + 3\frac{s_q}{q!}.$$

Proof. Similar to the arguments in Lemma 3.2, we have

$$|\mathcal{S}(C)| = \binom{4}{1}q^q - \binom{4}{2}q! + A_3 - A_4,$$

where A_i is the number defined in (3.7). For any subset T of $[4]$ with $|T| = 3$, we claim that $|\mathcal{A}_T| = \frac{s_q}{q!}$. To prove this claim, let us only consider the case where $T = \{1,3,4\}$. Note that C can be rewritten as $C = (2^{-1}(w+z), 2^{-1}(w-z), w, z) : \ w, z \in \mathbb{Z}_q\}$. If the third and fourth positions of $\mathbb{Z}_q$ are associated with two permutations π_1 and π_2, respectively, then the first position forms a permutation of $\mathbb{Z}_q$ if and only if $2^{-1}(\pi_1+\pi_2)$ is a permutation of $\mathbb{Z}_q$. This is equivalent to that $\pi_1+\pi_2$ is a permutation of $\mathbb{Z}_q$. Hence, we have $|\mathcal{A}_T| = \frac{s_q}{q!}$. We can similarly prove the claim for other three cases.

Hence, we have $A_3 = 4\frac{s_q}{q!}$. As we have $A_4 \leqslant |\mathcal{A}_{[3]}| = \frac{s_q}{q!}$, the desired result follows. ∎

Theorem 3.3. *For any odd integer $q \geqslant 3$, R_q is at least*

$$-\frac{1}{4(q-1)}\log_2\left((1-\frac{q!}{q^q})^4 - (\frac{3q}{\sqrt{e}}+o(q))(\frac{q!}{q^q})^3\right).$$

Moreover, for every sufficiently large odd q, this rate is bigger than that given in Theorem 3.1.

Proof. Let C be the q-ary code given in Lemma 3.4. Then we have

$$\begin{aligned} 1-\frac{q!|\mathcal{S}(C)|}{|C|^q} &\leqslant 1-\binom{4}{1}\frac{q!}{q^q}+\binom{4}{2}\left(\frac{q!}{q^q}\right)^2-3\frac{s_q}{q^{2q}} \\ &= \left(1-\frac{q!}{q^q}\right)^4-\left(\frac{3q}{\sqrt{e}}+o(q)\right)\left(\frac{q!}{q^q}\right)^3. \end{aligned}$$

The first claim is proved. To prove the second claim, it is sufficient to show that

$$\begin{aligned} &\left((1-\frac{q!}{q^q})^4-(\frac{3q}{\sqrt{e}}+o(q))(\frac{q!}{q^q})^3\right)^3 \\ &< \left((1-\frac{q!}{q^q})^3-(\frac{q}{\sqrt{e}}+o(q))(\frac{q!}{q^q})^3\right)^4. \end{aligned}$$

The left-hand side of the inequality is

$$\begin{aligned} &\left(1-\frac{q!}{q^q}\right)^{12}-\left(\frac{9q}{\sqrt{e}}+o(q)\right)\left(\frac{q!}{q^q}\right)^3(1+o(1)) \\ &= \left(1-\frac{q!}{q^q}\right)^{12}-\frac{9q}{\sqrt{e}}\left(\frac{q!}{q^q}\right)^3(1+o(1)). \end{aligned} \tag{3.13}$$

Similarly, the right-hand side of the inequality is

$$\left(1-\frac{q!}{q^q}\right)^{12}-\frac{4q}{\sqrt{e}}\left(\frac{q!}{q^q}\right)^3(1+o(1)). \tag{3.14}$$

As the number of (3.13) is less than the number of (3.14), the second claim follows. ∎

Similar to the case where q is odd, we can also improve the lower bound given in Theorem 3.2 if q is divisible by 4.

Theorem 3.4. *For any integer q with $q \equiv 0 \pmod 4$, R_q is at least*

$$-\frac{1}{4(q-1)}\log_2\left((1-\frac{q!}{q^q})^4-(\frac{3q}{\sqrt{e}}+o(q))(\frac{q!}{q^q})^3\right).$$

Moreover, for every sufficiently large q with $q \equiv 0 \pmod 4$, this rate is bigger than that given in Theorem 3.2.

Proof. Case 1: $q = 2^r$ for some integer $r \geqslant 2$. Choose an element $\alpha \in \mathbb{F}_q - \mathbb{F}_2$ and consider the code $C = \{(x, y, x+y, x+\alpha y) : \ x, y \in \mathbb{F}_q\}$. Then as in the proof of Lemma 3.4, one can show that $|\mathcal{S}(C)| \geqslant \binom{4}{1}q^q - \binom{4}{2}q! + 3\frac{s_q}{q!}$. By the same arguments in the proof of Theorem 3.3, we obtain the desired result.

Case 2: $q = 2^r p$ for some integer $r \geqslant 2$ and an odd prime $p \geqslant 3$. Choose an element $\alpha \in \mathbb{F}_{2^r} - \mathbb{F}_2$ and consider the ring $\mathbb{F}_{2^r} \times \mathbb{Z}_p$. Define the code $C = \{(x, y, x+y, x+(\alpha,-1)y) : \ x, y \in \mathbb{F}_{2^r} \times \mathbb{Z}_p\}$. C is a $[4,2]$-MDS code by observing that both $(\alpha,-1)$ and $(\alpha,-1)-(1,1) = (\alpha-1,-2)$ are invertible elements in $\mathbb{F}_{2^r} \times \mathbb{Z}_p$. The desired result then follows from the similar arguments in the proofs of Lemma 3.4 and Theorem 3.3. ∎

Acknowledgements

We are grateful to Venkat Guruswami who brought this topic to us. He gave a talk on his paper [11] when he was visiting Nanyang Technological University in 2018.

References

[1] Noga Alon and Joel H. Spencer. *The Probabilistic Method, Third Edition.* Wiley-Interscience series in discrete mathematics and optimization. Wiley, 2008.

[2] Erdal Arikan. Upper bound on the zero-error list-coding capacity. *Information Theory, IEEE Transactions on*, 40:1237 – 1240, 08 1994.

[3] Miklos Bona. *Handbook of enumerative combinatorics.* Discrete Mathematics and Its Applications. CRC Press, Hoboken, NJ, 2015.

[4] C. Cooper. A lower bound for the number of good permutations. *Nat. Acad. Sci. Ukraine*, 213:15–25, 2000.

[5] C. Cooper, R. Gilchrist, I. N. Kovalenko, and D. Novakovic. Estimation of the number of “good” permutatio with applications to cryptography. *Cybernetics and Systems Analysis*, 35(5):688–693, Sep 1999.

[6] Simone Costa and Marco Dalai. New bounds for perfect k-hashing. *CoRR*, abs/2002.11025, 2020.

[7] M. Dalai, V. G. Carnegie, and J. Radhakrishnan. An improved bound on the zero-error list-decoding capacity of the 4/3 channel. In *2017 IEEE International Symposium on Information Theory (ISIT)*, pages 1658–1662, June 2017.

[8] Sean Eberhard. More on additive triples of bijections. *CoRR*, abs/1704.02407, 2017.

[9] P. Elias. Zero error capacity under list decoding. *IEEE Transactions on Information Theory*, 34(5):1070–1074, Sep. 1988.

[10] M. Fredman and J. Komlós. On the size of separating systems and families of perfect hash functions. *SIAM Journal on Algebraic Discrete Methods*, 5(1):61–68, 1984.

[11] Venkatesan Guruswami and Andrii Riazanov. Beating Fredman-Komlós for Perfect k-Hashing. In *46th International Colloquium on Automata, Languages, and Programming (ICALP 2019)*, volume 132, pages 92:1–92:14, Dagstuhl, Germany, 2019.

[12] J. Körner. Fredman-komlós bounds and information theory. *SIAM Journal on Algebraic Discrete Methods*, pages 560–570, 1986.

[13] J. Körner and K. Marton. New bounds for perfect hashing via information theory. *European Journal of Combinatorics*, 9(6):523–530, 1988.

[14] N Kuznetsov. Applying fast simulation to find the number of good permutations. *Cybernetics and Systems Analysis - CYBERN SYST ANAL-ENGL TR*, 43:830–837, 11 2007.

[15] F. Manners S. Eberhard and R. Mrazović. Additive triples of bijections, or the toroidal semiqueens problem. *Journal of the European Mathematical Society*, 21(2):441–463, 2019.

[16] M. A. Tsfasman, S. G. Vlădutx, and Th. Zink. Modular curves, shimura curves, and goppa codes, better than varshamov-gilbert bound. *Mathematische Nachrichten*, 109(1):21–28, 1982.

[17] Ilan Vardi. *Computational recreations in Mathematica*. Addison Wesley, 1991.