ABSTRACT

Title of dissertation: COMBINATORIAL METHODS IN CODING THEORY

Arya Mazumdar, Doctor of Philosophy, 2011

Dissertation directed by: Professor Alexander Barg
Department of Electrical and Computer Engineering
and Institute of Systems Research

This thesis is devoted to a range of questions in applied mathematics and signal processing motivated by applications in error correction, compressed sensing, and writing on non-volatile memories. The underlying thread of our results is the use of diverse combinatorial methods originating in coding theory and computer science.

The thesis addresses three groups of problems. The first of them is aimed at the construction and analysis of codes for error correction. Here we examine properties of codes that are constructed using random and structured graphs and hypergraphs, with the main purpose of devising new decoding algorithms as well as estimating the distribution of Hamming weights in the resulting codes. Some of the results obtained give the best known estimates of the number of correctable errors for codes whose decoding relies on local operations on the graph.

In the second part we address the question of constructing sampling operators for the compressed sensing problem. This topic has been the subject of a large body of works in the literature. We propose general constructions of sampling matrices based on ideas from coding theory that act as near-isometric maps on almost all sparse signal. This matrices can be used for dimensionality reduction and compressed sensing.

In the third part we study the problem of reliable storage of information in non-volatile memories such as flash drives. This problem gives rise to a writing scheme that relies on relative magnitudes of neighboring cells, known as rank modulation. We establish the exact asymptotic behavior of the size of codes for rank modulation and suggest a number of new general constructions of such codes based on properties of finite fields as well as combinatorial considerations.

COMBINATORIAL METHODS IN CODING THEORY

by

Arya Mazumdar

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2011

Advisory Committee:
Professor Alexander Barg, Chair/Advisor
Professor John J. Benedetto
Professor Prakash Narayan
Professor Adrian Papamarcou
Professor Aravind Srinivasan
Professor Sennur Ulukus

Dedication

To the time my wife and I spent in Maryland

Acknowledgments

Fast and foremost I would like to thank my PhD advisor, Professor Alexander Barg for his guidance throughout the past five years. I always found him literally a-knock-on-the-door away with plenty of time for me. I am indebted to him for the teachings, supports and careful directions he provided me. He taught me to look at the theory of codes, for error correction and beyond, with a number of interesting perspectives that really helped me to broaden my horizons. It was a pleasure to work and learn from such an extraordinary academic: I wish to carry his legacy forward.

The members of my dissertation committee were very kind to agree to be on the advisory board. I would like to especially acknowledge Professor Prakash Narayan, who served as a member in the committees of my PhD qualifier, my thesis proposal, and also my thesis defense. In the last few years, he has always been very gracious with his invaluable advises to me.

I am grateful to Mario Blaum, Navin Kashyap and Ronny Roth for being excellent coworkers and my mentors. I have learned a lot from all of them. The occasional visits of Gregory Kabatiansky and Gilles Zémor to Maryland were also very helpful to me. Olgica Milenkovic and Pascal Vontobel are two of my first acquaintances in the coding theory community. I have had useful discussions with both of them in different stages of my thesis.

I would like to acknowledge the contributions of some excellent teachers of University of Maryland. Apart from Professors Barg and Narayan, I owe my knowledge to the superb teaching of Professors Leonid Koralov, Steve Marcus, Adrian Papamarcou, James Schafer and Aravind Srinivasan.

In my colleagues Himanshu, Prasanth, Punarbasu and Sirin, I always found a set of keen and patient ears ready to hear my research problems. I would like to specially thank Punarbasu, for being one of my closest friends in the four years we overlapped here. He, along with Arijit, Biswadip, Kaushik, Mohan (and family), Rajibul, Snehaunshu and Tania made my stay here in College Park full of moments to cherish.

My parents have the most invaluable contributions to my life. My choice of a career in research can largely be attributed to the inclinations of my father, Amay Mazumdar. My mother, Juthika, to me is the sweetest person in the world. She sacrificed a lot for a proper upbringing of her children. My sister Eshita (who goes by the name Tuli) is my dearest and I am very proud of her. I am grateful to my extended family, my grandparents and others, for the love they have given me. I like to specially mention Dodo whom I wish for all the best.

Much of my accomplishments, including the completion of this dissertation, would not have been possible without the care and support of my beloved wife, Barna. After my undergraduate studies in engineering, she is the one who seeded the urge in me to explore the theoretical aspects of subjects that interest me. She

has direct contributions in this thesis: in few occasions she supplied me with crucial ideas and discussions with her have almost always been very helpful. But most importantly, during the course of this dissertation we found our first home together in Maryland, where we have spent every single moment sharing the ups and downs of life as one for the past few years. She has made me the happiest I ever was.

I gratefully acknowledge the supports of the NSF grants CCF0830699, CCF101-8012 and CCF065271 at different stages of my thesis research.

Contents

Part I

Codes

CHAPTER 1

Introduction

1.1 Introduction

This thesis is devoted to combinatorial aspects of the theory of *error-correcting codes*. In most communication or storage systems operating in noisy environments, codes are required to ensure that the transmitted or stored information is error-free. In a communication scenario, the information to be transmitted is encoded as one of the elements, or *codewords* of a code $\mathcal{C}$. The effect of the noise in discrete channels can often be quantified by the number of errors introduced in a codeword in transmission. Therefore, the number of errors that the code can correct is a natural measure of code's quality. The type of errors is related to the properties of the communication channel: the most common channel models of communication give rise to the Hamming metric, which is used as a figure of merit in a large part of coding theory. At the same time, other channels arising in applications can be related to other types of errors, calling for studies of coding in permutations, coding over matrices, spherical codes with the Euclidean metric, etc.

In this thesis we address three aspects of combinatorics for codes. In the first part, we discuss the parameters and performance of *codes on graphs*, an important area in classical coding theory that deals with *reliability* of information transmission over channels that introduces bit-flip or similar errors. In the second part of the thesis, an *exploration* of coding-theoretic ideas applied to *signal processing* problems stresses the fact that coding theory can be considered as a powerful tool in discrete mathematics and can be used to solve seemingly unrelated problems such as construction of *sampling matrices* for the compressed sensing problem. The third part of the thesis emphasizes present *innovations* in coding theory where its principles are applied to tackle reliability issues in non-conventional systems such as data protection in flash memory devices. A common thread of these works is application of combinatorial methods to asymptotic problems of coding theory with an emphasis on constructions, bounds on the parameters of codes and performance of decoding algorithms.

In the following sections of this introductory chapter, we briefly describe sev-

eral concepts used throughout the dissertation.

1.2 Error-correcting codes

Let $\mathcal{X}$ be a metric space equipped with a distance function $\mathrm{d} : \mathcal{X}^2 \to \mathbb{R}$. A *code* $\mathcal{C}$ is a subset of $\mathcal{X}$ with the property that any two elements or *codewords* of $\mathcal{C}$ are far apart. Formally, let d be the largest integer such that for all $\boldsymbol{c}_1 \neq \boldsymbol{c}_2 \in \mathcal{C}$:

$$\mathrm{d}(\boldsymbol{c}_1, \boldsymbol{c}_2) \geq d.$$

The number d is called the *minimum distance* or simply the *distance* of the code $\mathcal{C}$.

Given a positive integer d and a metric space $\mathcal{X}$, the main aim of coding theory is to find the largest set $\mathcal{C}$ such that the distance of $\mathcal{C}$ is at least d. Such a subset need not be unique. Let us consider next an example to see why such subsets will be of interest.

Let $\mathbb{F}_2^n$ be the n-dimensional vector space over the binary field $\mathbb{F}_2$. Let $\mathrm{wt}(\boldsymbol{z})$ be the *Hamming norm*, i.e., the number of nonzero coordinates in the vector $\boldsymbol{z} \in \mathbb{F}_2^n$, also called the Hamming weight. The *Hamming distance* between vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_2^n$ is defined as $\mathrm{wt}(\boldsymbol{x} - \boldsymbol{y})$.

Consider the following communication scenario with a code $\mathcal{C} \subseteq \mathbb{F}_2^n$ over an adversarial channel. The sender selects a vector $\boldsymbol{x}$ from $\mathcal{C}$ as a message to transmit over the channel. The channel flips a few bits of the transmitted vector and the received vector is $\boldsymbol{x}+\boldsymbol{e}$, where $\mathrm{wt}(\boldsymbol{e})$ is the number of errors the channel introduces. If the minimum distance of the code is d and the channel can introduce at most $t \triangleq \lfloor (d-1)/2 \rfloor$ errors, it is possible to identify the transmitted vector $\boldsymbol{x}$ uniquely from the received vector just by finding the element of $\mathcal{C}$ that is nearest to the received vector. The number t is called the error-correcting capability of the code $\mathcal{C}$. The size of the set $\mathcal{C}$ equals the number of possible messages that can be sent over the channel. Therefore, given a minimum distance (or error-correcting capability), one aims to construct a code of the largest possible size.

It turns out that this is a very difficult problem. The largest possible size of a code in $\mathbb{F}_2^n$ with minimum distance d is denoted by $A(n, d)$. Even the asymptotic behavior of this quantity is largely unknown. We will later discuss some simple techniques to bound this quantity from below and from above.

The rate of a code $\mathcal{C} \in \mathbb{F}_2^n$ is defined to be:

$$R(\mathcal{C}) = \frac{\log |\mathcal{C}|}{n}. \tag{1.1}$$

Here and in rest of the thesis the base of logarithms is 2 if not specified otherwise. The code maps 2^{Rn} messages into the space $\mathbb{F}_2^n$. We can say that, as a result of this mapping, Rn bits of information are encoded in a codeword of $\mathcal{C}$ of length n.

A straightforward way to estimate the transmitted vector $\boldsymbol{x}$ from the received vector $\boldsymbol{y} = \boldsymbol{x} + \boldsymbol{e}$ is to compare $\boldsymbol{y}$ with all the vectors in $\mathcal{C}$ and to state that $\boldsymbol{x} = \hat{\boldsymbol{x}}$,

where $\hat{\boldsymbol{x}}$ is the codeword closest to $\boldsymbol{x}$ by the Hamming distance. However, if the size of the code is large, such decoding becomes computationally prohibitive. To simplify the decoding and encoding procedures, coding theory often resorts to studying *linear codes*, i.e., codes that form linear subspaces of $\mathbb{F}_2^n$. A large amount of research in coding theory is devoted to linear codes that afford low-complexity, computationally feasible decoding algorithms. In Chapters 2, 3, 4 we discuss construction of codes with this issue in mind. We note that the minimum distance of a linear code equals the minimum weight among nonzero codewords.

More detailed information about the code (than is given by the minimum distance) is provided by the distribution of Hamming distances between the codewords. For a code $\mathcal{C} \in \mathbb{F}_2^n$ the *distance distribution* is the set of numbers $(A_0, A_1, \ldots, A_n)$, where

$$A_w = \frac{1}{|\mathcal{C}|}|\{(\boldsymbol{x}_1, \boldsymbol{x}_2) \in \mathcal{C}^2 : \mathrm{d}(\boldsymbol{x}_1, \boldsymbol{x}_2) = w\}|.$$

If $\mathcal{C}$ is a linear code then A_w is simply the number of vectors of weight w. In this case the distance distribution is called the *weight distribution* of the code $\mathcal{C}$.

To motivate the last definition, consider a channel that introduces errors in a probabilistic manner. Instead of the adversarial channel described above, consider transmission of binary data over a binary symmetric channel (BSC) which introduces errors (bit-flips) in the transmitted bits independently with some fixed probability $p \in (0, 1)$. The distance distribution of the code enables one to estimate the average probability of decoding error for the code in such scenario. This possibility was recognized and gainfully used by Gallager [52] and was amplified by the well-known work of Poltyrev [89]. This paper provides estimate of the error probability of complete decoding for binary linear codes with a known weight distribution. There is sizable literature on weight distributions of various ensembles of linear codes as well as on general estimates of the weight distribution (e.g., [5, 9, 16, 21, 22, 27, 77, 78, 85]).

A useful concept related to the distance distribution of the code $\mathcal{C}$ is that of the dual distance of $\mathcal{C}$. To define it, let us introduce the MacWilliams transform of the distance distribution of a code [80, p. 139]. This is the set of numbers $(A_0^\perp, A_1^\perp, \ldots, A_n^\perp)$, where for all w

$$A_w^\perp = \frac{1}{|\mathcal{C}|} \sum_{i=0}^{m} A_i \mathcal{K}_i(w), \tag{1.2}$$

where $\mathcal{K}_i(t)$ is a Krawtchouk polynomial of degree i. It is known that $A_0^\perp = 1, A_w^\perp \geq 0$ for all w. The number $d^\perp$ such that $A_1^\perp = \cdots = A_{d^\perp - 1}^\perp = 0, A_{d^\perp}^\perp > 0$ is called the dual distance of the code $\mathcal{C}$.

1.3 The sphere-volume bounds

Let $\mathcal{X}$ be a finite metric space. A metric ball centered at a point $\boldsymbol{x} \in \mathcal{X}$ and radius r is defined as follows:

$$\mathcal{B}_r(\boldsymbol{x}) = \{\boldsymbol{y} \in \mathcal{X} : \mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) \leq r\}.$$

In many examples of metric spaces, the volume of the ball does not depend on the center. For instance, this is the case for the Hamming space $\mathbb{F}_2^n$, which simplifies many proofs in coding theory. However, a number of applications that we consider gives rise to metric spaces in which this property does not hold.

Suppose, $|\mathcal{B}_r(\boldsymbol{x})| = B_r$ is independent of $\boldsymbol{x}$. Let $\mathcal{C} \in \mathcal{X}$ be a code of largest possible cardinality with minimum distance d. Then,

$$\frac{|\mathcal{X}|}{B_{d-1}} \leq |\mathcal{C}| \leq \frac{|\mathcal{X}|}{B_t}, \tag{1.3}$$

where $t = \lfloor (d-1)/2 \rfloor$. The lower bound is called the *Gilbert-Varshamov* (GV) bound and the upper bound is known as the *sphere-packing* (Hamming) bound [80, p.19, 33]. The upper bound follows from the fact that balls of radius t around the codewords must be disjoint: if they are not, there exist two points $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{C}$ such that $\mathrm{d}(\boldsymbol{x}_1, \boldsymbol{x}_2) \leq d-1$, in violation of the distance condition.

To prove the lower bound in (1.3), consider the following greedy construction of a code. Start with an empty set $\mathcal{C}$. Pick an element $\boldsymbol{x}_1$ from $\mathcal{X}$ arbitrarily and include it in $\mathcal{C}$. Next, discard all the points of $\mathcal{X}$ in $\mathcal{B}_{d-1}(\boldsymbol{x}_1)$, and from the remaining elements pick $\boldsymbol{x}_2$ arbitrarily, and include it in $\mathcal{C}$. Choose $\boldsymbol{x}_3$ arbitrarily from $\mathcal{X} \backslash (\mathcal{B}_{d-1}(\boldsymbol{x}_1) \cup \mathcal{B}_{d-1}(\boldsymbol{x}_2))$ and include it in $\mathcal{C}$. This procedure can continue until all elements of $\mathcal{X}$ are either discarded or included in $\mathcal{C}$. By construction this will produce a code $\mathcal{C}$ with distance d. The size of the code must be at least $|\mathcal{X}|/B_{d-1}$ otherwise the procedure would have continued.

When the metric space is such that $|\mathcal{B}_r(\boldsymbol{x})|$ depends on $\boldsymbol{x}$, the inequalities (1.3) remain true if in the upper bound B_t is replaced by $\min\{|\mathcal{B}_t(\boldsymbol{x})| : \boldsymbol{x} \in \mathcal{X}\}$ and in the lower bound B_{d-1} is replaced by $\max\{|\mathcal{B}_{d-1}(\boldsymbol{x})| : \boldsymbol{x} \in \mathcal{X}\}$. Surprisingly, the lower bound holds true when B_{d-1} is replaced by the average volume of the ball instead of the maximum volume. This follows from an application of the well-known Turán theorem of graph theory [99].

For the binary Hamming space $\mathbb{F}_2^n$, we have $B_r = \sum_{i=0}^{r} \binom{n}{i}$. Consequently,

$$\frac{2^n}{\sum_{i=0}^{d-1} \binom{n}{i}} \leq A(n,d) \leq \frac{2^n}{\sum_{i=0}^{t} \binom{n}{i}}. \tag{1.4}$$

This can be easily extended to the q-ary Hamming space for $q > 2$.

If $\mathcal{C}$ is the largest possible code in the Hamming space with distance d then the rate of $\mathcal{C}$ satisfies

$$1 - \frac{1}{n} \log \sum_{i=0}^{d-1} \binom{n}{i} \leq R(\mathcal{C}) \leq 1 - \frac{1}{n} \log \sum_{i=0}^{t} \binom{n}{i}.$$

Using standard bounds for binomial coefficients, we can further simplify these inequalities. These bounds will be crucial in many parts of this thesis.

Lemma 1.3.1. [80, p.309ff] *Let* $0 < \alpha < 1$ *be such that* αn *is an integer. Then*

$$\frac{2^{n\,\mathrm{h}(\alpha)}}{\sqrt{8n\alpha(1-\alpha)}} \leq \binom{n}{\alpha n} \leq \frac{2^{n\,\mathrm{h}(\alpha)}}{\sqrt{2\pi n\alpha(1-\alpha)}}, \tag{1.5}$$

where $\mathrm{h}(z) = -z\log z - (1-z)\log(1-z)$ *is the* binary entropy function. *Moreover, when* $\alpha \leq 1/2$,

$$\frac{2^{n\,\mathrm{h}(\alpha)}}{\sqrt{8n\alpha(1-\alpha)}} \leq \sum_{i=0}^{\alpha n}\binom{n}{i} \leq 2^{n\,\mathrm{h}(\alpha)}. \tag{1.6}$$

The proof of the lemma uses Stirling's approximation of the factorial.

Using the lemma above, one deduces that

$$\liminf_{n\to\infty} \frac{\log A(n,\delta n)}{n} \geq 1 - \mathrm{h}(\delta) \tag{1.7}$$

and

$$\limsup_{n\to\infty} \frac{\log A(n,\delta n)}{n} \leq 1 - \mathrm{h}(\delta/2). \tag{1.8}$$

Here $0 \leq \delta \leq 1/2$ is called the *relative distance* of the code. The above equations represent the *asymptotic* GV bound and sphere-packing bound. The upper bound (1.8) can be improved for all $0 < \delta < 1$, and we refer the reader to [80, Ch. 17] for such improvements. At the same time, (1.7) is the best known asymptotic lower bound on the rate of a binary code. Given the rate, the achievable relative distance guaranteed by the GV bound is denoted by

$$\delta_{\mathrm{GV}}(R) \triangleq \mathrm{h}^{-1}(1-R).$$

A sequence of codes is called *asymptotically good* if as n increases both the rate and the relative distance stay bounded away from zero. The GV lower bound shows that there exist sequences of asymptotically good codes.

Let us end this section with an intuitive argument that relates the notion of *channel capacity* to the packing of spheres in the Hamming space. In a binary symmetric channel with crossover probability p, a 'typical' error vector will have weight approximately pn when n is large. Suppose that a code $\mathcal{C} \in \mathbb{F}_2^n$ is used for transmitting information over the channel. Transmitted vectors will have a low probability of being confused if for any two code vectors $\boldsymbol{x}_1, \boldsymbol{x}_2$, the probability $\Pr(\boldsymbol{x}_1 + \boldsymbol{e}_1 = \boldsymbol{x}_2 + \boldsymbol{e}_2)$ is small for any vectors $\boldsymbol{e}_1, \boldsymbol{e}_2$ of weight pn. In other words, spheres of radius pn about the codewords of $\mathcal{C}$ must be nearly disjoint. By (1.8) the rate of such code is at most $1 - \mathrm{h}(p)$. This informal argument can be made rigorous, and this upper bound on rate can be shown to be true. Moreover there exist code sequences that approach the transmission rate $1 - \mathrm{h}(p)$ with low probability of decoding error as n increases. The quantity $1 - \mathrm{h}(p)$ gives the value of Shannon *capacity* of the binary symmetric channel.

1.4 Linear codes

As mentioned above, a linear code is a linear subspace of the Hamming space $\mathbb{F}_2^n$. Without much effort all statements of this section generalize to the case of linear codes over a q-ary alphabet where q is a prime power other than 2. By definition a binary linear code $\mathcal{C}$ always has cardinality 2^k for some $0 \leq k \leq n$. We write $C[n, k, d]$ to refer to a linear code of length n, dimension k and distance d. A similar notation $\mathcal{C}(n, M, d)$ with respect to a code which is not necessarily linear replaces the dimension k with the cardinality $|\mathcal{C}| = M$.

A linear code can be viewed as a linear mapping $\mathcal{C} : \mathbb{F}_2^k \to \mathbb{F}_2^n$. A basis of this mapping forms a $k \times n$ matrix G called the generator matrix of the code $\mathcal{C}$. The code $\mathcal{C}$ can be defined by the generator matrix G in the following way:

$$\mathcal{C} = \{\boldsymbol{u}G : \boldsymbol{u} \in \mathbb{F}_2^k\}.$$

Here $\boldsymbol{u}$ denotes a row vector. A matrix H of rank $n - k$ such that $GH^T = 0$ is called the parity-check matrix of $\mathcal{C}$. The parity-check matrix H can also be used to define the code:

$$\mathcal{C} = \{\boldsymbol{x} \in \mathbb{F}_2^n : H\boldsymbol{x} = \mathbf{0}\};$$

here $\mathbf{0}$ is the all-zero vector of length $n - k$.

It is known that there exist sequences of linear codes that achieve the Gilbert-Varshamov bound. There are more than one way to prove this statement. We will show this using the so-called *probabilistic method* which will be another common thread of this dissertation.

One of the most important concepts related to linear codes is that of a *dual code*. For a linear code $\mathcal{C}$ with generator matrix G, the dual code $\mathcal{C}^\perp$ is defined to be the linear code whose parity-check matrix is G. The *dual distance* $d^\perp$ of the code $\mathcal{C}$ is the minimum distance of the dual code $\mathcal{C}^\perp$. Let $(A_w, w = 0, 1, \ldots, n)$ and $(A_w^\perp, w = 0, 1, \ldots, n)$ be the weight distributions of $\mathcal{C}$ and $\mathcal{C}^\perp$ respectively. These distributions are related by the MacWilliams identities (1.2).

The parity-check matrix of a linear code can be used to form another class of representations for codes, namely, graphical representations. A *hypergraph* $\mathrm{H}(V, E)$ consists of a set of vertices V and a set of (hyper)-edges E. The elements of E are subsets of the vertices V. In the case when the size of this subsets is restricted to be exactly 2, we obtain the notion of a *graph*. Given the parity-check matrix H of a linear code $\mathcal{C}$ one can form a hypergraph $\mathrm{H}(V, E)$ as follows. Every row of H can be identified with an element of V. That is $V = \{1, \ldots, n - k\}$. Every column of H can be identified with an element of E. Let us denote $E = \{e_i, i = 1, \ldots, n\}$. We set e_j to be a subset of V such that $i \in e_j$ if and only if $H(i, j) = 1$. This representation and its extensions will be the subject of study of the Chapters 2, 3 and 4.

1.5 Probabilistic methods

The term *probabilistic methods* loosely refer to the use of identities and inequalities from probability theory to prove combinatorial statements. Informally, if a random object picked from a collection of finite objects has a certain property with probability greater than zero then this proves the existence of at least one object in the collection that has the property. Another way to use the probabilistic method is by calculating the expected value of some random variable. It can often be claimed that the random variable can take a value less than or equal to the expected value.

As one would guess, there are no formal boundaries where such methods can be used. Let us illustrate the power of these methods via an example that will be useful later.

Proposition 1.5.1. *For any positive integer* $k < \log\left(\frac{2^n}{\sum_{i=1}^{d-1}\binom{n}{i}}\right)$, *there exists a linear code of length* n, *distance* d *and size* $\geq 2^k$.

Proof. We will prove the claim by considering the ensemble of linear codes defined by random parity-check matrices. Consider a $(n-k)\times n$ random binary matrix H whose entries are independent uniform Bernoulli random variables. Consider the code $\mathcal{C}$ given by $\mathcal{C} = \{\boldsymbol{x} \in \mathbb{F}_2^n : H\boldsymbol{x} = \mathbf{0}\}$. As the rank of H is at most $n-k$ the cardinality of the code $|\mathcal{C}| \geq 2^k$.

Now for any $\boldsymbol{x} \in \mathbb{F}_2^n \setminus \mathbf{0}$,

$$\Pr(\boldsymbol{x} \in \mathcal{C}) = \Pr(H\boldsymbol{x} = \mathbf{0}) = \frac{1}{2^{n-k}}.$$

Suppose that a random variable X denotes the number of non-zero codewords of weight at most $d-1$ that are in the code $\mathcal{C}$. Clearly,

$$\mathbb{E}X = \frac{1}{2^{n-k}}\Big[\sum_{i=1}^{d-1}\binom{n}{i}\Big].$$

If $\mathbb{E}X < 1$ then there must exist a code with distance d. However, $\mathbb{E}X < 1$ is true if

$$2^k < \frac{2^n}{\sum_{i=1}^{d-1}\binom{n}{i}}.$$

□

This shows that there exists a linear code that achieves the GV bound. Similar arguments can be used to prove the same fact considering a random generator matrix in place of the parity-check matrix.

From the above proof we also observe that the expected weight distribution of the code $\mathcal{C}$ is given by $A_0 = 1, A_w = \binom{n}{w}2^{k-n}, w = 1, \ldots, n$. Below we call this distribution the *binomial weight distribution*.

1.6 Isometric mappings

We will repeatedly use the concept of *near-isometric* mappings in this dissertation. Let $\mathcal{X}$ and $\mathcal{Y}$ be two metric spaces with distance functions d and d' respectively. A mapping $f : \mathcal{X} \to \mathcal{Y}$ is called *distance-preserving* if for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$,

$$\mathrm{d}'(f(\boldsymbol{x}), f(\boldsymbol{y})) \geq \mathrm{d}(\boldsymbol{x}, \boldsymbol{y}).$$

Distance-preserving mappings are useful in the context of showing existence of a code with certain distance in the metric space $\mathcal{Y}$, especially if it is easier to construct such a code in $\mathcal{X}$ and then map it to $\mathcal{Y}$ using f.

If the function $f : \mathcal{X} \to \mathcal{Y}$ is such that $f(\mathcal{X}) = \{f(\boldsymbol{x}) : \boldsymbol{x} \in \mathcal{X}\} \subset \mathcal{Y}$ then we call it an *embedding*. In Chapters 7 and 8 we will see many examples of embeddings where the function f^{-1} is well-defined and distance-preserving. In particular in Chapter 8 we use such embeddings to construct codes in the space of permutations.

If the mapping $f : \mathcal{X} \to \mathcal{Y}$ is such that for all $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$,

$$C_1 \mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) \leq \mathrm{d}'(f(\boldsymbol{x}), f(\boldsymbol{y})) \leq C_2 \mathrm{d}(\boldsymbol{x}, \boldsymbol{y}),$$

for some bounded constants C_1 and C_2, then the mapping is called *near-isometric*. Clearly near-isometric maps are distance preserving. Moreover if there exists a code with a certain distance guarantee in $\mathcal{X}$, there will exist a code with a similar guarantee in $\mathcal{Y}$, and thus the sphere-packing bound in $\mathcal{Y}$ will also be an upper bound on the size of the codes in $\mathcal{X}$. Also, it is possible to upper-bound the volume of spheres of radius d in $\mathcal{X}$ with the volume of spheres of radius $C_2 d$ in $\mathcal{Y}$, which could be easier to estimate. Then one can use Gilbert-Varshamov-type arguments to bound below the maximum size of a code in $\mathcal{X}$ even when computing the volume of the sphere in $\mathcal{X}$ is difficult.

When the metric spaces $\mathcal{X}$ and $\mathcal{Y}$ are ℓ_p-spaces, for instance, $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y} = \mathbb{R}^m$, $m < n$, near-isometric mappings are of special interest. Any homomorphism between $\mathbb{R}^n$ and $\mathbb{R}^m$ can be represented by a $m \times n$ matrix Φ. The well-known Johnson-Lindenstrauss lemma asserts that there exists an ensemble of $m \times n$ matrices where for most matrices in the ensemble the near-isometry property holds for any two fixed $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^n$. The rigorous statement of this lemma and a proof based on simple probabilistic methods can be found in [43].

In most applications one wants explicit matrices with the near-isometry property for special subsets of $\mathbb{R}^n$. Compressed sensing is one such application in the domain of signal processing where the requirement is to construct matrices that act as near-isometries on the set of *sparse* real signals (vectors with a small number of nonzero coordinates). It is interesting that one can construct such matrices from binary codes. This is the subject of discussion in Chapters 5, 6.

1.7 Organization

Apart from this introductory chapter, the dissertation is divided into three different parts to highlight different combinatorial applications. The first part is devoted to codes constructed by considering sparse *graphs* and *hypergraphs*. In Chapter 2 we introduce the concept of codes on graphs and then estimate the minimum distances and weight distributions for several ensembles of such codes. In Chapter 3 we discuss possible decoding algorithms for codes on graphs. An important result in this part is a new low-complexity decoding algorithm for codes on hypergraphs that has a good error-correcting guarantee for explicitly constructed codes. In Chapter 4 we consider a certain special ensemble of graph codes that are constructed by concatenating several copies of codes with small distance. We show that for this case, the proposed decoding algorithm corrects error patterns of weight that grows linearly with the length of the codes.

The second part of the thesis is devoted to construction of sampling matrices that find use in *compressed sensing* and *sparse recovery*, a highly active recent research area. In Chapter. 5 we show that sampling matrices that act as near-isometry on *sparse* signals can be constructed from binary linear codes. The construction discussed also gives the so-called dictionaries with *low coherence*. In the following chapter, a statistical near-isometry property is considered, and it is shown that matrices constructed from codes possess such property; thus, they are useful in the context of compressed sensing.

The third part highlights an application of coding theoretic ideas to a non-conventional error process motivated by the method of writing information onto flash memory devices. We consider the newly proposed model of *rank modulation* for coding in flash memories [67]. This scheme calls for construction of codes in the metric space of permutations with distance between them given by the minimum number of transpositions of adjacent symbols. We give bounds on the parameters of such codes in Chapter 7 and propose several classes of constructions for rank modulation codes in Chapter 8.

We would also like to mention a few other works by the present author that rely on the methods similar to those employed in this thesis. In the work on linear balancing sets [82] we explored a particular combinatorial question in the Hamming space. In [83], we considered a natural model for errors that arises in high-density magnetic media and estimated the capacity of information storage possible within such model. The recent paper [84] further extends these considerations. In all these works, the basic combinatorial ideas discussed in this thesis are highlighted as technical tools. These works are not included in the dissertation although they are within the scope of its methods.

Part II

Graphs

CHAPTER 2

Codes on Graphs: Weight Distribution

2.1 Introduction

In the description of linear codes in Chapter 1 we briefly mentioned graphical representation of codes. Although every linear code affords a graphical representation, codes that are constructed and analyzed primarily relying on it, are called *codes on graphs.* One useful approach to codes on graphs relies on the assumption that the maximum degree of the graph (the number of edges connected to a vertex) is kept constant while the number of vertices (and edges) increases. The resulting family of codes is known under the general name of *low density parity check codes* or *LDPC* codes.

Considerable attention in recent years was devoted to the study of error correction with graph codes and in particular, LDPC codes. Codes on graphs account for some of the best known code families in terms of their error correction under low-complexity decoding algorithms. They are also known to achieve a very good tradeoff between the rate and relative distance. The most well-studied case is that of codes defined on a bipartite graph. In this construction, a code of length $N = mn$ is obtained by "parallel concatenation" of $2m$ codes of a small length n which refers to the fact that each bit of the codeword is checked by two independent length-n codes. These length-n codes are referred as local codes or subcodes below. The arrangement of parity checks is specified by the edges of a bipartite graph which are in one-to-one correspondence with the codeword bits.

There exists codes on bipartite graphs that are known to be asymptotically good, i.e., to have nonvanishing rate R and relative distance δ as the code length N tends to infinity. Constructive families of bipartite-graph codes with the best known tradeoff between R and δ have been found in [16].

Moving from constructive families to existence results obtained by averaging over ensembles of bipartite-graph codes, it is possible to derive even better rate-distance tradeoffs. In particular, bipartite-graph codes with random local codes and random bipartite graphs attain the asymptotic GV bound (1.7) for relatively small code rates and are only slightly below it for higher rates [16].

A natural way to generalize codes on bipartite graphs is to consider concatenations governed by regular l-partite hypergraphs, $l \geq 2$. This code family was studied by Bilu and Hoory in [19]. While constructive families of bipartite-graph codes rely on the expansion property of the underlying graph, expansion is not well defined for hypergraphs. Instead, Bilu and Hoory put forward a property of hypergraphs, called ε-homogeneity, which replaces expansion in the analysis of hypergraph codes. They showed that there exist explicit, easily constructible families of ε-homogeneous hypergraphs, and estimated the number of errors corrected by their codes under a decoding algorithm suggested in their paper.

In this chapter we study hypergraph codes from the perspective of weight distributions. In Theorem 2.4.2 and its corollary we prove that the code ensemble defined by random regular l-partite hypergraphs and random local linear codes contains codes that meet the GV bound if the rate of codes satisfies a certain condition. This condition becomes less restrictive as l increases from the value $l = 2$, and covers all values of the rate R except a small neighborhood of $R = 1$ for large l. We also show (Theorem 2.4.7, Cor. 2.4.8) that the ensemble of hypergraph codes contains codes that attain the GV bound even if random hypergraphs are replaced with a *fixed* ε-homogeneous hypergraph. Specializing the last result for $l = 2$, we establish that expander codes of Sipser and Spielman [93] constructed from a fixed graph with a large spectral gap[1] and random local codes attain the GV bound with high probability. Finally, we derive an estimate of the average weight distribution for the ensemble of hypergraph codes with a fixed local code (see Theorem 2.4.5) that refines substantially a corresponding result in [16] and generalizes it from $l = 2$ to arbitrary l.

The material presented in this chapter is published in [14].

2.1.1 Codes on bipartite graphs

Let $\mathrm{G}(V, E)$ be a balanced, n-regular bipartite graph with the vertex set $V = V_1 \cup V_2, |V_1| = |V_2| = m$ and $|E| = N = nm$ edges. Let us choose an arbitrary ordering of edges in E. For a given vertex $v \in V$ this defines an ordering of edges $v(1), v(2), \ldots, v(n)$ incident to it. We denote this subset of edges by $E(v)$. Given a binary vector $\boldsymbol{x} \in \{0,1\}^N$, let us establish a one-to-one correspondence between the coordinates of $\boldsymbol{x}$ and the edges in E. For a given vertex v let $\boldsymbol{x}(v) = (x_e, e \in E(v))$ be the subvector that corresponds to the edges in $E(v)$. Denote by λ the second largest in the absolute value eigenvalue of the graph G.

Consider a set of binary linear codes $A_v[n, R_0 n]$ of length n and rate $R_0 \triangleq$

[1] The spectrum of a graph is defined as the spectrum of its adjacency matrix. If the graph is regular of degree n, then the largest eigenvalue is n. The spectral gap is defined as the difference between n and the second largest eigenvalue (by the absolute value). The spectral gap is known to control the expansion property of the graph [63]

$\dim(A_v)/n$, where $v \in V$. Define a *bipartite-graph code* as follows:

$$\mathcal{C}(\mathrm{G}, \{A_v, v \in V\}) = \{\boldsymbol{x} \in \{0,1\}^N : \forall_{v \in V_1 \cup V_2} \boldsymbol{x}(v) \in A_v\}.$$

The rate of the code $\mathcal{C}$ is easily seen to satisfy

$$R(\mathcal{C}) \geq 2R_0 - 1. \tag{2.1}$$

If we assume that all the local codes are the same, i.e., $A_v = A$, where $A[n, R_0 n, d_0 = \delta_0 n]$ is some linear code, then the distance of the code $\mathcal{C}$ can be estimated as follows [104]:

$$d/N \geq \delta_0^2 \Big(1 - \frac{\lambda}{d_0}\Big)^2$$

(we will write $\mathcal{C}(\mathrm{G}, A)$ instead of $\mathcal{C}(\mathrm{G}, \{A\})$ in this case). In particular, if the spectral gap of G is large, i.e., λ is small compared to d_0, then the relative distance d/N is close to the value δ_0^2, similarly to the case of the direct product code $A \otimes A$.

The weight distribution of bipartite-graph codes constructed from random regular bipartite graphs and a fixed local code A with a known weight distribution was analyzed in [25, 74]. In particular, it was shown that if A is the Hamming code then the ensemble $\mathscr{C} = (\mathcal{C}(\mathrm{G}, A))$ contains asymptotically good codes. Paper [16] also studied the weight distribution of bipartite-graph codes with random regular bipartite graphs. It was shown that for $N \to \infty$ the ensemble of codes constructed from random regular bipartite graphs and a fixed code A with distance $d_0 \geq 3$ contains asymptotically good codes. It has also been shown [16] that if the local codes are chosen randomly, then the code ensemble $\mathscr{C}$ contains codes that meet the GV bound in the interval of code rates $R(\mathcal{C}) \leq 0.202$.

2.1.2 Codes on hypergraphs

Generalizing the above construction, let $\mathrm{H} = (V, E)$ be a l-uniform l-partite n-regular hypergraph. This means that the set of vertices $V = V_1 \cup \cdots \cup V_l$ of H consists of l disjoint parts of equal size, say, $|V_i| = m, 1 \leq i \leq l$. Every hyperedge $\{v_{i_1}, v_{i_2}, \ldots, v_{i_l}\}$ contains exactly l vertices, one from each part, and each vertex is incident to n hyperedges. Below for brevity we say edges instead of hyperedges. The number of edges of H equals $N = mn$ which will also be the length of our hypergraph codes. As above, assume that the edges are ordered in an arbitrary fixed way and denote by $E(v)$ the set of edges incident to a vertex v. For definiteness, let us assume that edges $e_{(i-1)n+j}, j = 1, \ldots, n$ are incident to the vertex $v_i \in V_1, 1 \leq i \leq m$. This fixes the order of edges in part V_1, while the order in the other parts of H is established by the connections in the hypergraph.

Given a binary vector $\boldsymbol{x} \in \{0,1\}^N$ whose coordinates are in a one-to-one correspondence with the edges of H denote by $\boldsymbol{x}(v)$ its subvector that corresponds to the edges in $E(v)$.

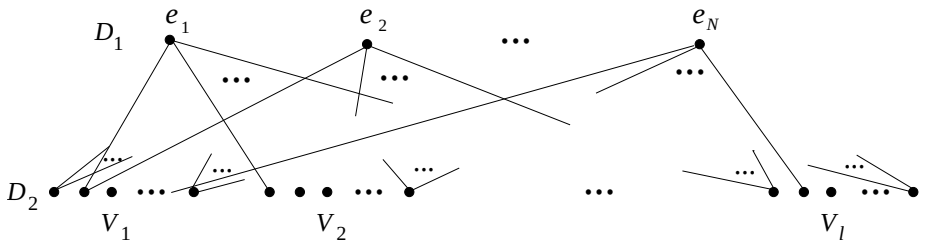


Figure 2.1: Alternate construction of the hypergraph code: The set $D_1 = \{e_1, \ldots, e_N\}$, where $\deg(e_i) = l$ for all i, represents the coordinates of the code (hyperedges of H); the sets $V_1, \ldots, V_l$, where $|V_j| = m$ for all j, represent the vertices of the hypergraph H. Each vertex $v_{i,j}, 1 \leq i \leq l, 1 \leq j \leq m$ carries a codeword of the local code A of length n.

Define a *hypergraph code* as follows:

$$\mathcal{C}(\mathrm{H}, \{A_v, v \in V\}) = \{\boldsymbol{x} \in \{0,1\}^N : \forall_{v \in V} \boldsymbol{x}(v) \in A_v\},$$

where $\{A_v, v \in V\}$ is a set of binary linear codes of length n. As above, if all the codes are the same, we write $\mathcal{C}(\mathrm{H}, A)$. Assume that all the codes A_v have the same rate R_0, then the rate of the code $\mathcal{C}$ satisfies

$$R(\mathcal{C}) \geq lR_0 - (l-1). \tag{2.2}$$

Remark 1. An equivalent description of the bipartite-graph code ensemble is obtained by considering an edge-vertex incidence graph of the graph $\mathrm{G}(V, E)$, i.e., a bipartite graph $(D_1 \cup D_2, \bar{E})$ where $D_1 = E, D_2 = V_1 \cup V_2$, each vertex in D_1 is connected to one vertex in V_1 and to one vertex in V_2, and there are no other edges in $\bar{E}$. Thus, for all $v \in D_1, \deg(v) = 2$, and for all $v \in D_2$, $\deg(v) = n$. The local code constraints are imposed on the vertices in D_2. By increasing the number of parts in D_2 from two to l, we then obtain the hypergraph codes defined above. This gives an alternate description of the hypergraph code presented in Fig. 2.1.

The ensemble of hypergraph codes with local constraints given by single parity-check codes was introduced by Gallager [52, p.12]. The proportion of errors correctable with these codes using the so-called "flipping" algorithm was estimated in [107]. Several generalizations of this ensemble were studied in [14, 19].

Definition 2.1.1. *[19] A hypergraph* H *is called ε-homogeneous if for every l sets $D_1, D_2, \ldots, D_l$ with $D_i \subseteq V_i$ and $|D_i| = \alpha_i m$,*

$$\frac{|E(D_1, D_2, \ldots, D_l)|}{N} \leq \prod_{i=1}^{l} \alpha_i + \varepsilon \min_{1 \leq i < j \leq l} \sqrt{\alpha_i \alpha_j}, \tag{2.3}$$

where $E(D_1, D_2, \ldots, D_l)$ denotes the set of edges that intersect all the sets D_i.

This definition quantifies the deviation of the hypergraph H from the expected behavior of a random hypergraph. For $l = 2$ the well-known "expander mixing lemma" (e.g., [63]) asserts that

$$\Big|\frac{|E(D_1, D_2)|}{N} - \alpha_1\alpha_2\Big| \le \frac{\lambda}{n}\sqrt{\alpha_1\alpha_2},$$

showing that regular bipartite graphs are λ/n-homogeneous. This inequality is frequently used in the analysis of bipartite-graph codes [93, 104].

Let $A[n, R_0 n, d_0 = \delta_0 n]$ be a binary linear code. The distance of a code $\mathcal{C}(\mathrm{H}, A)$ where H is ε-homogeneous satisfies [19]

$$d/N \ge \delta_0^{\frac{l}{l-1}} - c_1(\varepsilon, \delta_0, l) \tag{2.4}$$

where $c_1 \to 0$ as $\varepsilon \to 0$.

One of the main results in [19] gives an explicit construction of ε-homogeneous hypergraphs H starting with a regular graph $\mathrm{G}(U, E)$ with degree Δ and second eigenvalue λ. Putting $V_i = U, i = 1, 2, \ldots, l$ and introducing a hyperedge whenever the l vertices in the graph G are connected by a path of length $l - 1$, that paper shows that the resulting hypergraph is n-regular and ε-homogeneous with $n = \Delta^{l-1}, \varepsilon = 2(l-1)\lambda/\Delta$. Therefore, starting with a family of Δ-regular bipartite graphs with a large spectral gap, one can construct a family of regular homogeneous hypergraphs with a small value of ε. Paper [19] also shows that random n-regular hypergraphs with high probability are $O(1/\sqrt{n})$-homogeneous.

2.2 Ensembles of graph codes

Below we consider ensembles of random codes on graphs and hypergraphs. There have been many studies in coding theory that consider different ensembles of graph codes (see, [16, 25, 26, 52, 74, 77, 78, 87, 106, 107]). There are two components in these codes that can be randomized; namely, the graph and the local codes. Below we consider all three possible randomization scenarios: the graph is random but the local codes are fixed, the graph is fixed but the local codes are random, and both the graph and local codes are random. We proceed to define three general ensembles of codes on graphs that includes all previous considerations as special cases.

Let us consider the scenarios when the (hyper)graph will be selected randomly. In the case of bipartite graphs this is done as follows. Connect the edges $e_{(i-1)n+j}$, $j = 1, \ldots, n$ to the vertex $v_i \in V_1$, $i = 1, \ldots, m$. Next choose a permutation on the set E with a uniform distribution and connect the remaining half-edges to the vertices in V_2 using this permutation. Similarly, to construct an ensemble of random hypergraphs, we choose $l - 1$ permutations independently with uniform distribution and use them to connect the parts of H.

Random linear codes are selected from the standard ensemble of length-n codes defined by $n(1-R_0) \times n$ random binary parity-check matrices whose entries are chosen independently with a uniform distribution.

Definition 2.2.1. *We consider the following three ensembles of hypergraph codes.*

Ensemble $\mathscr{C}_1(l)$. *A code* $\mathcal{C}(\mathrm{H}, \{A_1, \ldots, A_l\}) \in \mathscr{C}_1(l)$ *is constructed by choosing a random* l*-partite hypergraph* H *and choosing random local linear codes* A_i *of length* n *independently for each part* $V_i \in V$.

Ensemble $\mathscr{C}_2(l, A)$. *A code* $\mathcal{C}(\mathrm{H}, A) \in \mathscr{C}_2$ *is constructed by choosing a random* l*-partite hypergraph* H *and using the same fixed local code* $A[n, R_0 n, d_0]$ *as a local code at every vertex.*

Ensemble $\mathscr{C}_3(l, \mathrm{H})$. *A code* $\mathcal{C}(\mathrm{H}, \{A_v\})$ *from this ensemble is formed by choosing a fixed, nonrandom hypergraph* H *and taking random local linear codes* A_v *independently for each vertex* $v \in V$.

Our purpose is to compute ensemble-average asymptotic weight distributions for codes in these ensembles and to estimate the average minimum distance assuming that $m \to \infty$ and n is a constant. The case $l = 2$ corresponds to ensembles of bipartite-graph codes, some of which were studied in [16, 25, 74]. Below we will cover the remaining cases for the code ensembles $\mathscr{C}_i(l)$, $i = 1, 2, 3$ and any $l \geq 2$. The analysis for the ensemble $\mathscr{C}_1(l)$ extends the results of [16] from graphs to hypergraphs while the results for the remaining two ensembles have no direct precursors in the literature.

2.3 Prior work on concatenated codes

Calculations of the weight distributions in this chapter reveal some parallels to earlier results for concatenated code ensembles [9, 21, 98]. These similarities are to some extent expected because graph codes can be interpreted as a version of code concatenation. This fact has been discussed in detail in [17]. At the same time, calculations for graph codes are rather different from those for concatenated codes. We define concatenated codes and quote earlier results to underscore these links.

Code concatenation is a method of obtaining long codes from short codes. The first construction of this kind was put forward by Elias [49] under the name of product codes. Given two binary linear codes $A[n_1, k_1, d_1]$ and $B[n_2, k_2, d_2]$ we can construct a code $\mathcal{C} = A \otimes B$ by taking the tensor product. The code $\mathcal{C}$ has parameters $[n_1 n_2, k_1 k_2, d_1 d_2]$. Decoding of product codes poses a set of interesting and as yet unresolved questions (see e.g., [55, 92]). As noted above, codes on bipartite graphs form an extension of Elias's construction.

Another extension was suggested by Forney under the name concatenated codes [51]. Given a binary linear code $B[n_2, k_2, d_2]$ and a linear code $A[n_1, k_1, d_1]$ over

the field of $q = 2^{k_2}$ elements, we define a concatenated code via the composite map

$$\mathbb{F}_2^{k_1 k_2} \to \mathbb{F}_q^{k_1} \xrightarrow{A} (\mathbb{F}_q)^{n_1} \to (\mathbb{F}_2^{k_2})^{n_1} \xrightarrow{B} (\mathbb{F}_2)^{n_2 n_1}.$$

In words: the binary message of $k_1 k_2$ bits is first mapped to a q-ary message vector for the code A, and the resulting length n_1 vector over $\mathbb{F}_q$ is mapped on n_1 binary vectors of the code B. The image of this map is a concatenated code $\mathcal{C}$ with the parameters $[n_1 n_2, k_1 k_2, \geq d_1 d_2]$. Unlike product codes, the value $d_1 d_2$ is only a coarse lower bound on the distance of the concatenated code $\mathcal{C}$. This fact manifests itself in early works [21, 98] showing that ensembles of concatenated codes (with various assumptions on the ensembles of constituent codes) attain the GV bound, which is a much higher distance than the product bound. If one of the constituent codes A, B is fixed rather than random, the resulting distance estimates are below the GV bound, but still better than the product bound. We quote a result from [21, 98].

Theorem 2.3.1. *Suppose that the code B is drawn from the ensemble of binary codes defined by random parity-check matrices of dimensions $(n_2 - k_2) \times n_2$ and the code A is a q-ary Reed-Solomon code [80, p. 294] with the parameters $[n_1, k_1]$, where $q = 2^{k_2}$. The average number of vectors of weight $w = \omega N$ over the ensemble of resulting concatenated codes equals $2^{n_1 n_2 (F(\omega, R, R_0) + o(1))}$, where*

$$F(\omega, R, R_0) = \begin{cases} R - R_0 - \omega \log(2^{1-R_0} - 1) & 0 < \omega \leq 1 - 2^{R_0 - 1} \\ \mathrm{h}(\omega) + R - 1 & \omega > 1 - 2^{R_0 - 1} \end{cases}$$

where $R_0 = {}^{k_2}/_{n_2}, R = {}^{k_1 k_2}/_{n_1 n_2}$. The ensemble-average relative distance as a function of the code rate R is given by

$$\delta(R) = \begin{cases} \delta_{\mathrm{GV}}(R) & R_0 \geq \log(2(1 - \delta_{\mathrm{GV}}(R))) \\ \frac{R - R_0}{\log(2^{1-R_0} - 1)} & 0 \leq R_0 < \log(2(1 - \delta_{\mathrm{GV}}(R))) \end{cases}$$

where $\delta_{\mathrm{GV}}(x) = \mathrm{h}^{-1}(1 - x)$ as defined in Sec. 1.3.

In particular, this theorem exhibits the range of code rates for the attainment of the GV bound. This happens when the weight distribution of the code matches the weight distribution of random codes from the parity-check ensemble (the so-called binomial weight distribution). The theorem also quantifies the gap between the weight distribution of concatenated codes and the binomial weight distribution in the case that the GV bound is not attained. The latter result is important for estimating the error probability of decoding as shown for instance in [9].

This result has set stage for a number of later studies on concatenated codes and codes on graphs [12, 16, 61]. Similarities of this result with some theorems in this chapter will become apparent below; yet we note that the proofs involve new ideas compared to the earlier works.

2.4 Weight distribution

Below $A_w = A_w(\mathcal{C})$ denotes the number of codewords of weight w and $\mathrm{wt}(\boldsymbol{x})$ denotes the Hamming weight of the vector $\boldsymbol{x}$ as before. Before proceeding, we note that upper bounds on the ensemble-average weight distribution in many cases also give a lower bound on the code's distance.

Lemma 2.4.1. *Suppose that for an ensemble of codes $\mathscr{C}$ of length N there exists an $\omega_0 > 0$ such that*

$$\lim_{N\to\infty} \sum_{w\le\omega_0 N} \mathbb{E}A_w = 0.$$

Then for large N the ensemble contains codes whose relative distance satisfies $d/N \ge \omega_0$.

Proof. The proof is almost obvious because for a randomly chosen code $\mathcal{C} \in \mathscr{C}$,

$$\Pr[d(\mathcal{C}) \le \omega_0 N] \le \sum_{w\le\omega_0 N} \Pr[A_w(\mathcal{C}) \ge 1] \le \sum_{w\le\omega_0 N} \mathbb{E}A_w.$$

□

2.4.1 Ensemble $\mathscr{C}_1(l)$

Theorem 2.4.2. *For $m \to \infty$ the average weight distribution over the ensemble of linear codes $\mathscr{C}_1(l)$ of length $N = mn$ and rate (2.2) satisfies $\mathbb{E}A_{\omega N} \le 2^{N(F+\gamma)}$, where*

$$F = \begin{cases} \omega l \log_2(2^{(1-R)/l} - 1) - (l-1)\,\mathrm{h}(\omega) & \text{if} \quad 0 \le \omega \le 1 - 2^{(R-1)/l}, \\ \mathrm{h}(\omega) + R - 1 & \text{if} \quad \omega \ge 1 - 2^{(R-1)/l}, \end{cases} \tag{2.5}$$

and

$$\gamma \le (l/n)(1 + \log_2 n) + (l/2N)\log_2(2N).$$

Proof. Let $C_i, i = 1, \ldots, l$ be the set of vectors $\boldsymbol{x} \in \{0,1\}^N$ that satisfy the linear constraints of part V_i of the hypergraph H so that the code $\mathcal{C} = \cap_i C_i$. Let $P_i = \Pr[\boldsymbol{x} \in C_i]$. The events $\boldsymbol{x} \in C_i$ for different i are independent, and therefore

$$\Pr[\boldsymbol{x} \in \mathcal{C}] = P_i^l$$

(for any $i = 1, \ldots, l$). Let $B_w(C_i)$ be the random number of vectors of weight w in the code C_i. Then, $\mathbb{E}B_w(C_i) = \binom{N}{w} P_i$ and

$$\mathbb{E}A_w(\mathcal{C}) = \binom{N}{w} \Pr[\boldsymbol{x} \in \mathcal{C}] = \binom{N}{w} \prod_{i=1}^{l} \frac{\mathbb{E}B_w(C_i)}{\binom{N}{w}}.$$

Let $\mathcal{X}_{s,w}$ be the set of vectors of weight $w = \omega N$ whose nonzero coordinates are incident to some vertices $v_{i_1}, \dots, v_{i_s} \in V_1$, $s \geq w/n$. Let $w_j = \mathrm{wt}(x(v_{i_j})), j = 1, \dots, s$ and let $\omega_j = w_j/n$. We have

$$|\mathcal{X}_{s,w}| = \binom{m}{s} \sum_{\substack{w_1,\dots,w_s \\ \sum w_j = w}} \prod_{j=1}^{s} \binom{n}{w_j} \leq \binom{m}{s} \sum_{\substack{w_1,\dots,w_s \\ \sum w_j = w}} 2^{n \sum_j \mathrm{h}(\omega_j)}.$$

By convexity of the entropy function, the maximum of the last expression on $\omega_1, \dots, \omega_s$ under the constraint $n \sum_j \omega_j = \omega N$ is attained for $\omega_j = \omega m/s, j = 1, \dots, s$. Since the sum contains no more than n^s terms, we obtain

$$|\mathcal{X}_{s,w}| \leq 2^{m\,\mathrm{h}(x) + s \log n + sn\,\mathrm{h}(\omega m/s)} \leq 2^{N(x\,\mathrm{h}(\omega/x) + \varepsilon)}$$

where $x = s/m$ and $\varepsilon = (1 + \log n)/n$. A vector $\boldsymbol{x} \in \mathcal{X}_{s,w}$ is contained in C_1 with probability $2^{sn(R_0 - 1)}$. Thus,

$$\mathbb{E}B_w(C_1) = |\mathcal{X}_{s,w}| 2^{sn(R_0 - 1)},$$

and the same expression is true for $\mathbb{E}B_w(C_i), i = 2, \dots, l$. Therefore,

$$\mathbb{E}A_w(\mathcal{C}) \leq \binom{N}{w}^{-(l-1)} 2^{lN(\max_{\omega \leq x \leq 1}(x(\mathrm{h}(\omega/x) + R_0 - 1)) + \varepsilon)}.$$

Since $l(R_0 - 1) \leq R - 1$, we obtain $\mathbb{E}A_w(\mathcal{C}) \leq 2^{N(F(\omega) + \gamma)}$, where

$$\begin{aligned} F(\omega) &\leq -(l-1)\,\mathrm{h}(\omega) + l \max_{\omega \leq x \leq 1}(x(R_0 - 1 + \mathrm{h}(\omega/x))) \\ &\leq -(l-1)\,\mathrm{h}(\omega) + \max_{\omega \leq x \leq 1}(x(R - 1 + l\,\mathrm{h}(\omega/x))). \end{aligned}$$

The maximum on x of $x(R - 1 + l\,\mathrm{h}(\omega/x))$ is attained for $x = x_0 = \omega/(1 - z)$ where $l \log_2 z = R - 1$. The two cases in the theorem are obtained depending on whether $x_0 < 1$ or not. If $x_0 < 1$, we substitute x_0 in the expression for $F(\omega)$ and obtain

$$F(\omega) \leq -(l-1)\,\mathrm{h}(\omega) + \omega l \log_2 \frac{z}{1 - z}$$

which implies the first part of (2.5) on account of the identity $R - 1 + l\,\mathrm{h}(z) = l(1 - z)\log_2(z/(1 - z))$. If $x_0 \geq 1$, we substitute the value $x = 1$ to obtain the second part of (2.5). □

Corollary 2.4.3. *Let ω^* be the only nonzero root of the equation*

$$\omega\Big(R - 1 - l \log_2\Big(1 - 2^{(R-1)/l}\Big)\Big) = (l-1)\,\mathrm{h}(\omega).$$

Then the average relative distance over ensemble $\mathscr{C}_1(l)$ behaves as

$$\delta(R) \geq \begin{cases} \omega^*, & \text{if } R \leq \log_2(2(1 - \delta_{\mathrm{GV}}(R))^l) \\ \delta_{\mathrm{GV}}(R), & \text{if } R > \log_2(2(1 - \delta_{\mathrm{GV}}(R))^l). \end{cases}$$

The proof is analogous to the proof of Corollary 4 in [16] and will be omitted.

For $l = 2$ it was proved in [16] that ensemble $\mathscr{C}_1$ contains codes that reach the GV bound if the code rate satisfies $0 \le R \le 0.202$. This result forms a particular case of the above corollary. Increasing l, we find that the ensemble contains codes that reach the GV bound for the values of the rate as shown below:

$$\begin{array}{lll} l = 3 & 4 & 10 \\ R \le 0.507 & 0.737 & 0.998. \end{array}$$

Thus already for $l = 10$ almost all codes in the ensemble $\mathscr{C}_1$ attain the GV bound for all but very high rates.

2.4.2 Ensemble $\mathscr{C}_2(l, A)$.

In this case the results depend on the amount of information available for the local codes. Specifically, [16] shows that for $l = 2$ the ensemble contains asymptotically good codes provided that the distance of the local code A is at least 3. In the case when the weight distribution of the code A is known, a better estimate is known from [25, 74].

Theorem 2.4.4. *Let A be a linear code of length n with weight enumerator $a(x) = \sum_{i=0}^{n} a_i x^i$, i.e., the number of codewords of weight i in A is a_i. Let A_w be the random number of codewords of weight w of a code $\mathcal{C}(\mathrm{H}, A) \in \mathscr{C}_2(l, A)$. Then its average value over the ensemble satisfies*

$$\lim_{N\to\infty} \frac{1}{N} \log_2 \mathbb{E} A_{\omega N} \le -(l-1)\,\mathrm{h}(\omega) + \frac{l}{\ln 2}\Big(\frac{1}{n} \ln a(e^{s^*}) - s^*\omega\Big),$$

where s^* is the root of $(\ln a(e^s))'_s = n\omega$.

This theorem enables us to estimate the asymptotics of the mean relative distance $\delta = \lim\limits_{m\to\infty} \frac{\mathbb{E}d(\mathcal{C})}{N}$ for the ensemble $\mathscr{C}_2$. Let us consider several examples.

1. Let $l = 3$ and let A be the Hamming code of length $n = 15$ and rate $R_0 = 11/15$. Then the rate $R(\mathscr{C}_2) \ge 0.2$ and the distance $\delta = 0.2307$. The relative GV distance for this rate is $\delta_{\mathrm{GV}}(0.2) = 0.2430$.

2. Let $l = 3$ and let A be the Hamming code of length $n = 31$. Then $R(\mathscr{C}_2) \ge 16/31$ and $\delta \approx 0.0798$. Using the same code with $l = 4$ gives $R(\mathscr{C}_2) \ge 11/31$ and $\delta \approx 0.1607$ while $\delta_{\mathrm{GV}}(11/31) \approx 0.1646$.

3. Let $l = 3$ and let A be the 2-error-correcting primitive BCH code of length $n = 31$ and rate $R_0 = 21/31$. Then the rate $R(\mathscr{C}_2) \ge 1/31$ and the value of δ is ≈ 0.3946608. The relative GV distance for this rate is $\delta_{\mathrm{GV}}(1/31) \approx 0.3946614$.

Let us turn to the case when only the minimum distance d_0 of the code A is available. In [16] the case $l = 2$ was addressed, proving that as long as $d_0 \ge 3$,

there exists an $\varepsilon > 0$ such that the ensemble-average relative distance $\delta > \varepsilon$ as $m \to \infty$. In the next theorem this result is extended to arbitrary $l \geq 2$. We also prove a related result which gives an upper bound on the average weight spectrum and provides a way of estimating the value of ω_0.

Theorem 2.4.5. *(a) Let A be the local code of length n and distance d_0 used to construct the ensemble $\mathscr{C}_2(l, A)$ of hypergraph codes. Let $x_0 = x_0(\omega)$ be the positive solution of the equation*

$$\omega n + \sum_{i=d_0}^{n} \binom{n}{i} (\omega n - i) x^i = 0. \tag{2.6}$$

The ensemble-average weight distribution satisfies

$$\lim_{N\to\infty} \frac{1}{N} \log \mathbb{E} A_{\omega N} \leq \frac{l}{n} \log \frac{1 + \sum_{i=d_0}^{n} \binom{n}{i} x_0^i}{x_0^{\omega n}} - (l-1)\,\mathrm{h}(\omega).$$

(b) The inequality $d_0 > l/(l-1)$ gives a sufficient condition for the ensemble to contain asymptotically good codes.

Proof. (a) Let H be a random hypergraph and $\mathcal{C}(\mathrm{H}, A)$ be the corresponding code. Recall that $\mathcal{C} = \cap_i C_i$, where C_i is the set of vectors that satisfy the constraints of part i of the graph. Let $U_i(w, d_0)$ be the set of vectors $\boldsymbol{x} \in \{0,1\}^N$ such that $\mathrm{wt}(\boldsymbol{x}) = w$ and $\mathrm{wt}(\boldsymbol{x}(v)) = 0$ or $\mathrm{wt}(\boldsymbol{x}(v)) \geq d_0$ for all $v \in V_i$. Since the number of such vectors is the same for all i, below we write $|U(w, d_0)|$ omitting the subscript. Let us choose a vector $\boldsymbol{x} \in \{0,1\}^N$ randomly with a uniform distribution. Then

$$\Pr[\boldsymbol{x} \in C_1 |\, \mathrm{wt}(\boldsymbol{x}) = w] \leq \frac{|U(w, d_0)|}{\binom{N}{w}}$$

and for $i \geq 2$,

$$\Pr[\boldsymbol{x} \in C_i |\, \mathrm{wt}(\boldsymbol{x}) = w, \boldsymbol{x} \in C_1] = \Pr[\boldsymbol{x} \in C_i |\, \mathrm{wt}(\boldsymbol{x}) = w].$$

Then

$$\begin{aligned} \mathbb{E} A_w(\mathcal{C}) = \binom{N}{w} \Pr[\boldsymbol{x} \in \mathcal{C} |\, \mathrm{wt}(\boldsymbol{x}) = w] &= \binom{N}{w} (\Pr[\boldsymbol{x} \in C_1 |\, \mathrm{wt}(\boldsymbol{x}) = w])^l \\ &\leq \frac{|U(w, d_0)|^l}{\binom{N}{w}^{l-1}}. \end{aligned} \tag{2.7}$$

Given a vector $\boldsymbol{x}$ denote by $j_\imath$ the number of vertices $v \in V_i$ such that $\mathrm{wt}(\boldsymbol{x}(v)) = \imath$. Clearly,

$$|U(w, d_0)| = \sum_{\substack{j_0, j_{d_0}, j_{d_0+1}, \ldots, j_n \\ \sum \imath j_\imath = w,\, j_0 + \sum\limits_{\imath \geq d_0} j_\imath = m}} \binom{m}{j_0, j_{d_0}, \ldots, j_n} \prod_{\imath=d_0}^{n} \binom{n}{\imath}^{j_\imath}.$$

This sum contains no more than $(m+1)^n = O(N^n)$ terms, so for $N \to \infty$ its exponent is determined by the maximum term (which has exponential growth).

We obtain

$$\frac{1}{N}\log|U(\omega N, d_0)|^l \;\le\; \frac{l}{n}\max_{\substack{\nu_0,\nu_{d_0},\ldots,\nu_n \\ \sum \imath\nu_\imath=\omega n,\,\sum\nu_\imath=1}} \Big\{ \mathrm{h}(\nu_0,\nu_{d_0},\nu_{d_0+1},\ldots,\nu_n) + \sum_{\imath=d_0}^{n}\nu_\imath \log\binom{n}{\imath}\Big\} + \frac{\log N}{m}, \tag{2.8}$$

where $\nu_\imath = j_\imath/m, \imath = 0, d_0, d_0+1, \ldots, n$, and $\mathrm{h}(\boldsymbol{z})$ denotes the entropy of the probability vector $\boldsymbol{z} \in \mathbb{R}^{n+1}$. The objective function is concave, so the point of extremum is found from the system of equations

$$\binom{n}{i}(1-\sum_{\imath=d}^{n}\nu_\imath) = \nu_i\mu^{-i}, \quad i = d_0, d_0+1, \ldots, n$$
$$\sum_{\imath=d_0}^{n}\imath\nu_\imath = \omega n.$$

Its solution is given by

$$\nu_i = \frac{\binom{n}{i}\mu^i}{1+\sum_{\imath=d}^{n}\binom{n}{\imath}\mu^\imath}, \quad i = d_0, d_0+1, \ldots, n,$$

where μ is chosen so as to satisfy the last equation of the system. Evaluating $\sum_i i\nu_i$ and writing x instead of μ, we observe that it should satisfy Eq. (2.6). This equation has a unique root $x_0 > 0$ because putting $x = p/(1-p)$, we can write it as

$$\omega n\Big(\frac{\Pr[X=0]}{\Pr[X \ge d_0]} + 1\Big) = \mathbb{E}[X | X \ge d_0],$$

where X is a binomial $(p, 1-p)$ random variable. As p changes from 0 to 1, the left-hand side of the last equation decreases monotonically from $+\infty$ to ωn while the right-hand side increases monotonically from d_0 to n.

Finally, computing the entropy and simplifying, we obtain the estimate

$$\lim_{N\to\infty}\frac{1}{N}\log|U(\omega N, d_0)|^l \le \log\frac{1+\sum_{i=d_0}^{n}\binom{n}{i}x_0^i}{x_0^{\omega n}}.$$

(b) The proof of the second part is analogous to the case of $l = 2$ in [16]. Let

$w, 1 \le w \le N$ be the weight and let $p = w/d_0$. We have

$$\begin{aligned} |U(w,d_0)| &\le \sum_{i=w/n}^{p} \binom{n}{i}\binom{n}{d_0}^i \binom{in}{(p-i)d_0} \\ &\le \binom{m}{p}\binom{n}{d_0}^p \sum_{i=w/n}^{p} \binom{pn}{(p-i)d_0} \\ &\le \binom{m}{p}\binom{n}{d_0}^p 2^{pn}. \end{aligned}$$

Then

$$\mathbb{E}A_w(\mathcal{C}) \le \Big(\binom{m}{p}\binom{n}{d_0}^p 2^{pn}\Big)^l \binom{N}{w}^{1-l}.$$

Using the estimates $(\frac{n}{k})^k \le \binom{n}{k} \le (\frac{en}{k})^k$, we compute

$$\begin{aligned} \mathbb{E}A_w(\mathcal{C}) &\le \Big(\frac{em}{p}\Big)^{pl} n^{d_0 pl} 2^{lpn} \Big(\frac{w}{N}\Big)^{w(l-1)} \\ &= (sm/w)^{\frac{w}{d_0}(l-d_0(l-1))}, \end{aligned}$$

where $s = ((ed_0 2^n)^l n^{d_0})^{\frac{1}{l-d_0(l-1)}}$. Thus, for any ω satisfying $\omega < s/m$, the average number of vectors of weight ωN tends to 0 as $m \to \infty$ as long as $d_0(l-1) > l$. This proves that under this condition the ensemble contains asymptotically good codes. □

Example: Let A be the $[7,4,3]$ Hamming code and let $l = 2$. Theorem 2.4.5(a) implies a lower bound $\delta \ge 0.01024$ on the average relative distance for the ensemble $\mathscr{C}_2(2, A)$. This improves upon previous results ([25, 74]; also Part (b) of this theorem) which assert only that the ensemble contains asymptotically good codes. Of course, in this case we can use the entire weight distribution of the code A to find the estimate $\delta \ge 0.186$ from Theorem 2.4.4; however, in cases when the weight distribution is difficult to find, the last theorem provides new information for the ensemble of graph codes.

Similarly, for $A[23, 12, 7]$ from Theorem 2.4.5(a) we obtain the estimate $\delta \ge 0.0234$. Again, using the entire weight distribution, it is possible to obtain a better estimate.

Part (a) of the last theorem implies the following corollary which shows what happens to the average weight spectrum of the ensemble for long local codes.

Corollary 2.4.6. *Let $d_0 = \delta_0 n$. Then*

$$\frac{1}{N}\log \mathbb{E}A_{\omega N}(\mathcal{C}) \le \frac{l\omega}{\delta_0}\,\mathrm{h}(\delta_0) - (l-1)\,\mathrm{h}(\omega) + \gamma$$

where $\gamma \le (\log N)/m + (\log n)/n$.

Proof. In (2.8) let us bound above $\mathrm{h}(\cdot)$ by $\log n$. Then

$$\frac{1}{N}\log|U(w,d_0)|^l \le \frac{l}{n}\max_{\substack{\nu_{d_0},\dots,\nu_n \\ \sum \imath\nu_\imath=\omega n,}} \sum_{\imath=d_0}^{n} \nu_\imath \log\binom{n}{\imath} + \gamma.$$

Computing the maximum amounts to solving a linear programming problem whose dual is

$$\omega n z \to \min$$
$$\imath z \ge \log\binom{n}{\imath}, \imath = d_0, d_0+1, \dots, n;\ z \ge 0.$$

Its solution is given by $z^* = \omega n \max_{d_0\le\imath\le n} \log\binom{n}{\imath}/\imath$. We obtain

$$\frac{1}{N}\log|U(w,d_0)|^l \le l\omega \max_{\delta_0\le x\le 1} \frac{\mathrm{h}(x)}{x} + \gamma \le l\omega\,\mathrm{h}(\delta_0)/\delta_0 + \gamma.$$

Employing (2.7) now completes the proof. □

2.4.3 Ensemble $\mathscr{C}_3(l,\mathrm{H})$

Theorem 2.4.7. *Assume that* H *is ε-homogeneous. For $m\to\infty$ the average weight distribution over the ensemble of linear codes $\mathscr{C}_3(l,\mathrm{H})$ satisfies $\mathbb{E}A_{\omega N} \le 2^{N(F+\gamma)}$ where*

$$F = \begin{cases} -x_0(1-R) + x_0^l\,\mathrm{h}\left(\frac{\omega}{x_0^l}\right) & \text{if } x_0 < 1, \\ \mathrm{h}(\omega) + R - 1 & \text{if } x_0 \ge 1, \end{cases}$$

where x_0 is the unique positive root of the equation

$$l x^{l-1}\log(x^l/(x^l-\omega)) = 1-R, \tag{2.9}$$

$\gamma = l(n+\log m)/N + \varepsilon$.

Proof. Let $\mathcal{C}\in\mathscr{C}_3(l,\mathrm{H})$ and let $\boldsymbol{x}\in\{0,1\}^N$ be a nonzero vector. Denote by B_i the set of nonzero vertices of $\boldsymbol{x}$ in the part $V_i, i=1,\dots,l$. Let $E=|E(B_1,B_2,\dots,B_l)|$. Let $b_i=|B_i|, \beta_i = b_i/m$, then the probability that $\boldsymbol{x}\in\mathcal{C}$ equals $2^{-(1-R_0)N\sum_i\beta_i}$. Assume w.l.o.g. that $\beta_1<\beta_2<\dots<\beta_l$. The average number of vectors of weight $w=\omega N$ in the code $\mathcal{C}$ can be bounded above as

$$\mathbb{E}A_w \le \sum_{\omega m\le b_1,b_2,\dots,b_l\le m} \binom{N\prod_{i=1}^l\beta_i + \varepsilon\sqrt{b_1 b_2}}{w} \prod_{i=1}^{l}\binom{n}{b_i} 2^{-(1-R_0)N\sum_i\beta_i}.$$

Then

$$\frac{1}{N}\log\mathbb{E}A_{\omega N} \le \max_{\substack{\omega\le\beta_i\le 1 \\ \prod_i\beta_i\ge\omega}} \Big\{\prod_i \beta_i\,\mathrm{h}\Big(\frac{\omega}{\prod_i\beta_i}\Big) - (1-R_0)\sum_i\beta_i\Big\} + \gamma.$$

Let $\phi(\beta_1, \dots, \beta_l)$ be the function in the brackets in the last expression. Let us prove that ϕ is concave in the domain $\mathcal{D} = \prod_i [\omega, 1] \cap \{(\beta_1, \dots, \beta_l) : \prod_i \beta_i \geq \omega\}$. Computing its Hessian matrix, we obtain

$$H_\phi = -\log e \begin{bmatrix} \frac{s_1}{\beta_1^2} & \frac{s_2}{\beta_1\beta_2} & \cdots & \frac{s_2}{\beta_1\beta_l} \\ \frac{s_2}{\beta_2\beta_1} & \frac{s_1}{\beta_2^2} & \cdots & \frac{s_2}{\beta_2\beta_l} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{s_2}{\beta_l\beta_1} & \frac{s_2}{\beta_l\beta_2} & \cdots & \frac{s_1}{\beta_l^2} \end{bmatrix},$$

where

$$s_1 = \frac{\omega \prod_i \beta_i}{\prod_i \beta_i - \omega}$$

$$s_2 = s_1 + \prod_i \beta_i \ln\Big(1 - \frac{\omega}{\prod_i \beta_i}\Big).$$

The matrix H_ϕ can be written as

$$H_\phi = -\log e(s_2 z z^T + (s_1 - s_2)\text{diag}(\beta_1^{-2}, \dots, \beta_l^{-2})),$$

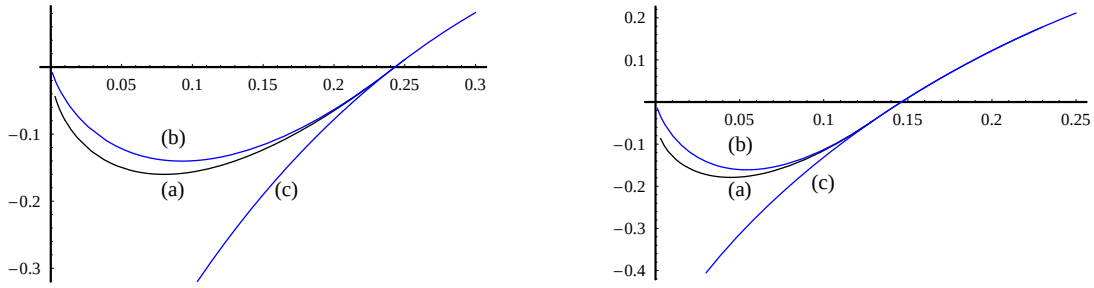
where $z = (1/\beta_1, \dots, 1/\beta_l)^T$ and diag$(\cdot)$ denotes a diagonal matrix. We wish to prove that H_ϕ is negative definite for $\beta_i > 0, 0 < \omega < \prod_i \beta_i$. Clearly, $s_1 > s_2$, and therefore the claim will follow if we show that $s_2 > 0$. This is indeed true because letting $Q = \prod_i \beta_i$ and using the inequality $x > \ln(1+x)$ valid for $x > -1, x \neq 0$, we have

$$s_2 = Q\Big(\frac{\omega}{Q-\omega} + \ln\frac{Q-\omega}{Q}\Big) > Q\Big(\ln\Big(1 + \frac{\omega}{Q-\omega}\Big) + \ln\frac{Q-\omega}{Q}\Big) = 0.$$

We will now show that the maximum of ϕ in $\mathcal{D}$ is attained on the line ℓ given by $\beta_1 = \beta_2 = \cdots = \beta_l$. Note that $\mathcal{D}$ is an intersection of convex domains and therefore itself convex. Moreover, the domain $\mathcal{D}$ is also symmetric in the sense that together with any point $p = (\beta_1, \dots, \beta_l)$ it also contains all the points obtained from p by permuting its coordinates, and the value of ϕ at each of these points is the same and equal to $\phi(p)$. Because ϕ is strictly concave, for any point $p \in \mathcal{D}, p \notin \ell$ it is possible to find a point q such that $\phi(q) > \phi(p)$ (any point q on the segment between p and one of its symmetric points will do). This shows that the global maximum of ϕ in $\mathcal{D}$ is attained on ℓ including possibly the point $\beta_1 = \cdots = \beta_l = 1$. Thus, we obtain

$$\frac{1}{N}\log \mathbb{E}A_w \leq \max_{\omega^{1/l} \leq x \leq 1} \{-(1-R)x + x^l \,\mathrm{h}\Big(\frac{\omega}{x^l}\Big)\} + \gamma.$$

The maximum of this expression on x is attained for x determined from (2.9). This equation has a unique positive root x_0 because the left-hand side is a falling function of x that takes all positive values for $x \in (\omega^{1/l}, \infty)$. This concludes the proof. $\square$



(I) (II)

Figure 2.2: Average weight spectra for ensembles of graph codes: (I) $l = 2, R = 0.2$, (II) $l = 3, R = 0.4$; (a) ensemble $\mathscr{C}_3(2, \mathrm{H})$, (b) ensemble $\mathscr{C}_1(2)$, (c) ensemble of random linear codes.

This theorem implies the following result.

Corollary 2.4.8. *For all values of the code rate satisfying $R \geq \log(2(1-\delta_{\mathrm{GV}}(R))^l)$, almost all codes in the ensemble $\mathscr{C}_3(l)$ approach the GV bound as $N \to \infty$.*

Proof. From the previous theorem, the GV bound is met for the first time when x_0 becomes 1. Substituting 1 in (2.9), we obtain a condition on ω in the form $\omega = 1 - 2^{(R-1)/l}$. As long as this value is less than $\delta_{\mathrm{GV}}(R)$, the ensemble-average relative distance approaches $\delta_{\mathrm{GV}}(R)$ as $N \to \infty$. □

We note that the condition for the attainment of the GV bound turns out to be the same as for the ensemble $\mathscr{C}_1(l)$ constructed from random graphs. The ε-homogeneity condition, and in particular, the expander mixing lemma for bipartite graphs are known to approximate the behavior of random graphs. This approximation turns out to be good enough to ensure that both ensembles contain GV codes in the same interval of code rates. Moreover, for small weights the average number of codewords for the ensemble $\mathscr{C}_3(l, \mathrm{H})$ turns out to be smaller than for the ensemble $\mathscr{C}_1(l)$. This is illustrated in the two examples in Fig. 2.2.

For $l = 2$ codes in the ensembles $\mathscr{C}_3$ and $\mathscr{C}_1$ reach the GV bound for code rates $R \leq 0.202$. For $R > 0.202$ the codes are still asymptotically good, although slightly below the GV bound. For these values of the rate, the average relative distance for the ensemble $\mathscr{C}_3$ is greater than for the ensemble $\mathscr{C}_1$ as shown by the following numerical examples.

R	0.3	0.5	0.7	0.9
$\mathscr{C}_1(2)$	0.18558	0.09276	0.03211	0.00337
$\mathscr{C}_3(2, \mathrm{H})$	0.18605	0.09492	0.03242	0.00380

Similar relations between the weight spectra and distances of the ensembles $\mathscr{C}_1(l), \mathscr{C}_3(l, H)$ hold also for larger values of l.

CHAPTER 3

Codes on Graphs: Decoding

3.1 Introduction

The principal appeal of graph codes lies in their performance under iterative local decoding procedures. This idea is prominent in the study of theoretical properties and applications of LDPC codes [91]. It has also given rise to a number of interesting results in the area of generalized LDPC codes, i.e., general linear codes on graphs.

Error correction with graph codes has been studied along two lines, namely, by examining explicit code families whose construction involves graphs with a large spectral gap, or by computing the average number of errors correctable with some decoding algorithm by codes from a certain random ensemble of graph codes. The focus of this chapter will be on the first direction. The second line of work will be the subject of Chapter 4.

The research topic of this chapter, initiated in Tanner's paper [96] and in Sipser and Spielman's [93], pursues estimates of error correction with codes on regular graphs with a small second eigenvalue and ensuing expansion properties. Presently it is known that such codes under iterative decoding can correct the number of errors equal to a half of the designed distance of graph codes [17, 94]. This estimate fits in a series of analogous results for various "concatenated" coding schemes and has prompted a view of graph codes as parallel concatenations of the local codes [17].

The focus of this chapter is on decoding of hypergraph codes. The only known algorithm for their decoding [19] stops short of exploiting the full power of these codes as indicated in particular by its parameter estimates. This shortcoming shows most prominently for the case of small relative distances when the proportion of errors corrected by this algorithm vanishes compared to the value of the distance. At the same time, the tradeoff between the rate and relative distance of hypergraph codes shows an improvement over bipartite graph codes for small values of the distance. Motivated by this, we propose a new decoding algorithm of hypergraph codes and estimate its error-correcting capability. We show that it corrects the number of errors which constitutes a fixed proportion of the code's distance. The

material presented in this chapter is published in [14].

3.2 Decoding of bipartite graph codes

We set stage by presenting a by now standard iterative decoding algorithm for bipartite graph codes of Zémor [104]. Consider a code $\mathcal{C}(\mathrm{G}, A)$ on a bipartite graph $\mathrm{G}(V, E)$, $V = V_1 \cup V_2$. Its decoding can be performed by a natural algorithm [104] that alternates between parallel decoding of local codes in the parts V_1 and V_2 until, hopefully, it converges to a fixed point. In this algorithm, the most current value of each edge (bit) is stored at the vertex in the part decoded in the most recent iteration.

For the ease of analysis we assume that the local codes are decoded to correct up to t errors, where $t \geq 0$ is an integer that satisfies $2t + 1 \leq d_0$ and d_0 is the distance of the code A. Formally, define a mapping $\psi_{A,t} : \{0,1\}^n \to \{0,1\}^n$ such that $\psi_{A,t}(\boldsymbol{z}) = \boldsymbol{x} \in A$ if $\boldsymbol{x}$ is the unique codeword that satisfies $d(\boldsymbol{z}, \boldsymbol{x}) \leq t$ and $\psi_{A,t}(\boldsymbol{z}) = \boldsymbol{z}$ otherwise. Let $\boldsymbol{y}^{(i)}$ be the estimate of the transmitted vector before the ith iteration, $i \geq 1$, where $\boldsymbol{y} = \boldsymbol{y}^{(1)}$ is the received vector. The next steps are repeated for a certain number of iterations.

Algorithm I ($\boldsymbol{y}^{(1)}$)

- i odd: for all $v \in V_1$ put $\boldsymbol{y}^{(i+1)}(v) = \psi_{A,t}(\boldsymbol{y}^{(i)}(v))$;
- i even: for all $v \in V_2$ put $\boldsymbol{y}^{(i+1)}(v) = \psi_{A,t}(\boldsymbol{y}^{(i)}(v))$.

We say more on this algorithm in the next chapter; however, upon some reflection it becomes clear this algorithm as well as other "edge-oriented" procedures do not easily generalize to the case of hypergraphs when one edge is checked by more than two vertices.

3.3 Decoding of hypergraph codes

In [19] the following alternative to Algorithm I is suggested: starting from the values of the bits stored on the edges of H decode in parallel all local codes in *all* parts of H and for each $v \in V$ form an independent decision about the codeword of A that corresponds to the edges $E(v)$. Next, the values of the bits at every vertex are updated, so that now every vertex stores an independent opinion of its bits' values. For the update, the value of the bit $x_e(v)$ is set to the majority value of the decoded versions of this bit at all the vertices $v' \in e \backslash v$, where $e \ni v$ is an edge (for this to be well-defined, the values of l are assumed to be even). The decoding then iterates, repeating this parallel decoding round until all the vertices agree on all bits.

In [19] this algorithm is shown to correct all patterns of errors provided that

their proportion, as a fraction of the blocklength N, is less than

$$\binom{l-1}{l/2}^{-2/l}\left(\frac{\delta_0}{2}\right)^{(l+2)/l} - c_2(\varepsilon,\delta_0,l) \tag{3.1}$$

where $c_2(\varepsilon,\delta_0,l)\to 0$ as $\varepsilon\to 0$. This algorithm consists of $\log N$ iterations, each of which has serial running time linear in the blocklength N. Its analysis relies on the ε-homogeneous property of H.

This result should be contrasted with the distance estimate of (2.4). For fixed values of $l > 2$, if one thinks of δ_0 as a variable quantity, then the number of correctable errors in (3.1) is not a constant fraction of the designed distance (2.4). For example, for $l = 4$, (3.1) gives a decoding radius equal to N times the fraction

$$\frac{\delta_0^{3/2}}{2\sqrt{6}}.$$

For small δ_0 this is a much smaller quantity than the relative designed distance $\delta_0^{4/3}$. This consideration is reinforced by the fact that advantages of hypergraph codes are most pronounced for small values of the distance δ.

Our objective is to propose an alternative decoding strategy that decodes a constant fraction of the designed distance.

For every $i = 1, 2\ldots, l$, we shall define a *i-th subprocedure* that decodes the local code A on every vertex belonging to the vertex set V_i. We shall claim that if the initial number of errors is less than a bound that we shall introduce, then *for at least one* i, the i-th subprocedure applied to the initial error pattern produces a pattern with a smaller number of errors.

Let us now describe the decoding procedure in more detail. For every vertex v, and the associated subspace $\{0,1\}^n$ where coordinates are indexed by the edges incident to v, we will use the following *threshold decoding* procedure T_κ of the constituent code A. This means that we introduce a number $\kappa \geq 2$, to be optimized later, and that we decode a vertex subcode *only if* its Hamming distance to the nearest codeword is less or equal to $\theta = d_0/\kappa$. If every codeword of A is at distance more than d_0/κ we leave the subvector untouched. Let $V_i = (v_{i,1},\ldots,v_{i,m})$ be the ith component of H. Given an N-vector $\boldsymbol{z} = (\boldsymbol{z}(v_{i,1}),\ldots,\boldsymbol{z}(v_{i,m}))$, we can decode each of the m of its subvectors with T_κ, obtaining an N-vector w. Abusing notation, we will write $w = T_\kappa(\boldsymbol{z})$. The *$i$-th subprocedure* now consists of applying T_κ to the component V_i.

As mentioned above, we shall claim that one among l of the i-th subprocedures lowers the total number of errors. However the decoding algorithm will not be able to discern which of the i-th subprocedures is successful. So the decoder will apply all l subprocedures in parallel to the received vector, yielding l output vectors. The next decoding iteration will have to be applied to every output of the preceding iteration, so that s iterations of the algorithm will yield l^s output vectors. We will only

apply the algorithm for a constant number of iterations however, until we are guaranteed that the number of remaining error for at least one of the l^s outputs has fallen below the error-correcting capability of Bilu and Hoory's decoding procedure. We then let the latter decoder take over and decode all l^s candidates. At least one of them is guaranteed to be the closest codeword, and it can be singled out simply by computing the Hamming distance of every candidate to the initial received vector.

To give a more formal description of the algorithm, suppose that $\boldsymbol{y} \in \{0,1\}^N$ is the vector received from the channel. In each iteration the processing is done in parallel in all the vertices of H. Let $\mathscr{Y}_i^j = \{\boldsymbol{y}_{i,l}^{(j)}\}$ be the set of N-vectors stored at the vertices of the component V_i before the jth iteration. By the discussion above, $|\mathscr{Y}_i^j| \leq l^{j-1}$.

We begin by setting $\mathscr{Y}_i^1 = \{\boldsymbol{y}\}$ for all i. Iteration $j, j = 1, 2, \ldots, s$ consists of running l parallel subprocedures. The ith subprocedure applies decoder T_κ to every vector $\boldsymbol{y}_{i,l}^{(j)}$ in the set $\mathscr{Y}_i^j$, replacing it with the vector $T_\kappa(\boldsymbol{y}_{i,l}^{(j)}), l = 1, \ldots, |\mathscr{Y}_i^j|$. The outcome of this step creates l potentially different decodings of every vector $\boldsymbol{y}_{i,l}^{(j)} \in \mathscr{Y}_i^j, i = 1, \ldots, l$. In the second part of the iteration we form the sets $\mathscr{Y}_i^{j+1}, i = 1, \ldots, l$ by replacing each vector $\boldsymbol{y}_{i,l}^{(j)} \in \mathscr{Y}_i^j$ with its decodings obtained in all the l subprocedures.

Next, we prove that one of the l subprocedures will actually diminish the number of errors. This analysis also relies on ε-homogeneity, although in a way different from [19]. Let $\mathcal{E}$ be the set of coordinates, i.e. the set of edges, that are in error. For every $i = 1 \ldots l$, let us partition the set of vertices in V_i that are incident to $\mathcal{E}$ into three subsets, G_i, N_i, B_i. The set G_i is the subset of vertices that will be correctly decoded, N_i is the subset of vertices that are left untouched by the threshold decoder, and B_i is the set of those vertices that are wrongly decoded to a parasite codeword of A. The situation is summarized in Figure 3.1. From now on by the *$\mathcal{E}$-degree* of a vertex we shall mean the degree of this vertex in the subhypergraph induced by the edge set $\mathcal{E}$. It should be clear that every vertex of G_i has $\mathcal{E}$-degree not more than d_0/κ, every vertex in N_i has $\mathcal{E}$-degree at least d_0/κ, and every vertex in B_i has $\mathcal{E}$-degree at least $(\kappa - 1)d_0/\kappa$.

We use a shorthand notation $\mathcal{E}(G_i)$ to mean the set of edges that has one of its endpoints in G_i. Similarly we shall write $\mathcal{E}(N_i)$ and $\mathcal{E}(B_i)$.

Lemma 3.3.1. *If the i-th decoding subprocedure introduces more errors than it removes, then $|\mathcal{E}(G_i)| \leq |\mathcal{E}|/\kappa$. Moreover, if*

$$\mu_i = \frac{|\mathcal{E}(N_i)|}{|\mathcal{E}(N_i) \cup \mathcal{E}(B_i)|}, \quad i = 1, \ldots, l$$

then

$$|\mathcal{E}(G_i)| \leq \frac{1 - \mu_i}{\kappa - \mu_i}|\mathcal{E}|.$$

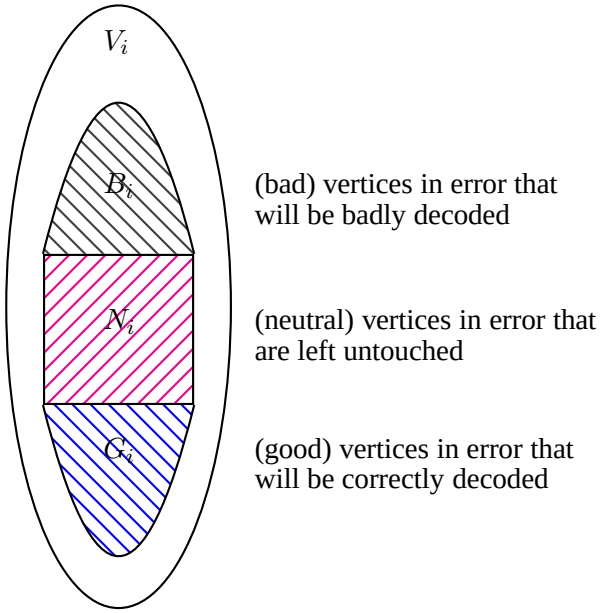


Figure 3.1: Details of the set of vertices incident to edges in error. The max $\mathcal{E}$-degree in G_i is less than d_0/κ, the min $\mathcal{E}$-degree in B_i is at least $(\kappa - 1)d_0/\kappa$, the min $\mathcal{E}$-degree in N_i is at least d_0/κ.

Proof. The first part of the lemma follows from the second part, which is proved as follows. We bound from above $|\mathcal{E}(G_i)|$, the set of edges removed, by the set of edges added, $|\mathcal{E}(B_i)|$: we get

$$\begin{aligned} |\mathcal{E}(G_i)| &\leq |B_i|\frac{d_0}{\kappa} = |B_i| d_0 \left(1 - \frac{1}{\kappa}\right) \frac{1}{\kappa - 1} \\ &\leq |\mathcal{E}(B_i)| \frac{1}{\kappa - 1}. \end{aligned}$$

The first inequality comes from the definition of κ and the threshold decoder. The second inequality states that $(1-1/\kappa)d_0$ is a lower bound on the minimum $\mathcal{E}$-degree in B_i. We now have

$$|\mathcal{E}| = |\mathcal{E}(G_i)| + |\mathcal{E}(N_i)| + |\mathcal{E}(B_i)| = |\mathcal{E}(G_i)| + |\mathcal{E}(B_i)|/(1-\mu_i) \tag{3.2}$$

$$\geq \frac{\kappa - \mu_i}{1 - \mu_i} |\mathcal{E}(G_i)|$$

which proves the lemma. □

Theorem 3.3.2. *For any $\alpha > 0$, if the number of errors eN is such that*

$$e \leq (1-\alpha) \frac{\delta_0^{l/(l-1)}}{(l+1)^{(l+1)/(l-1)}} \tag{3.3}$$

they can be corrected in time $O(N \log N)$.

Proof. The theorem will follow if we show that at least one subprocedure reduces the error count by a constant fraction. Indeed, in this case a constant number of rounds of the above algorithm will reduce the error count to any positive proportion of the designed distance whereupon the remaining errors will be removed in $O(\log N)$ steps of Bilu-Hoory's algorithm.

Assume toward a contradiction that *all* the i-th decoding subprocedures, $i = 1, \ldots, l$, introduce more errors than they remove. Let us introduce the following notation: $|\mathcal{E}| = eN, S_i = B_i \cup N_i$, $|S_i| = \sigma_i m$. Note that since the minimum $\mathcal{E}$-degree in S_i is at least d_0/κ, we have

$$\sigma_i \leq \kappa e / \delta_0. \tag{3.4}$$

Consider the subset of edges obtained from $\mathcal{E}$ by removing all edges incident to "good" vertices G_i for all i. We are left with a subhypergraph $\mathrm{H}_{\mathcal{E}}$ with vertex set S_i, $i = 1 \ldots l$. Use Lemma 3.3.1 (the first part) for all i to argue that the total fraction of edges in $\mathrm{H}_{\mathcal{E}}$ is at least $e(1-l/\kappa)$. Applying the ε-homogeneous property (2.3) gives

$$e\left(1 - \frac{l}{\kappa}\right) \leq \sigma_1 \cdots \sigma_l + \varepsilon \min_{1 \leq i < j \leq l} (\sigma_i \sigma_j)^{1/2}.$$

Applying (3.4) we obtain

$$e\left(1-\frac{l}{\kappa}\right) \le \left(\frac{\kappa e}{\delta_0}\right)^l + \varepsilon\frac{\kappa e}{\delta_0}.$$

This inequality does not hold (and therefore our assumption is false) if

$$e < \delta_0^{l/(l-1)} \left(\frac{1-l/\kappa-\varepsilon\kappa/\delta_0}{\kappa^l}\right)^{1/(l-1)}. \tag{3.5}$$

Taking $\kappa = l+1$, rewrite the expression in the brackets on the right as

$$\Big(\frac{1}{l+1}\Big)^{(l+1)/(l-1)}\Big(1-\frac{(l+1)^2\varepsilon}{\delta_0}\Big)^{\frac{1}{l+1}}.$$

By taking sufficiently large n it is possible to make ε small enough so that for any given $\alpha' > 0$ there holds

$$(1-(l+1)^2\varepsilon/\delta_0)^{1/l+1} > 1-\alpha'.$$

This means that (3.5) is satisfied for all

$$e < (1-\alpha')\frac{\delta_0^{l/(l-1)}}{(l+1)^{(l+1)/(l-1)}}.$$

Finally, choosing $\alpha' < \alpha$ guarantees that at least one subprocedure reduces the error count by a constant fraction. □

We see that the upper bound on the number of correctable errors given by Theorem 3.3.2 is a constant proportion γ of the designed distance δN (2.4), where $\gamma = 1/(l+1)^{(l+1)/(l-1)}$. For example, for $l = 3, 4$ we get $\gamma = 1/16$ and $1/14.2$, respectively.

The next theorem provides a better estimate of γ by refining the above analysis. The way this is done is to rely on the full power of Lemma 3.3.1 instead of its first part as above.

Theorem 3.3.3. *For any $\alpha > 0$, if the number of errors eN is such that*

$$e \le (1-\alpha)\delta_0^{l/(l-1)} \max_{\kappa\ge 2}\min_{0<\mu<1} f(\mu,\kappa)$$

with

$$f(\mu,\kappa) = \frac{[1-l(1-\mu)/(\kappa-\mu)]^{1/(l-1)}}{\kappa^{l/(l-1)}[\mu+(1-\mu)/(\kappa-1)]^{l/(l-1)}}$$

they can be corrected in time $O(N\log N)$.

Proof. We proceed as in the previous theorem, assuming toward a contradiction that each subprocedure increases the error count. Using the definition of μ_i given above,

$$|\mathcal{E}(S_i)| = \frac{|\mathcal{E}(B_i))|}{1-\mu_i} = \frac{|\mathcal{E}(N_i)|}{\mu_i}.$$

Recall that the subhypergraph $\mathrm{H}_{\mathcal{E}}$ is formed of the edges all of whose vertices are in S_i. To count the total fraction of edges $\beta(\mathrm{H}_{\mathcal{E}})$ in the subhypergraph $\mathrm{H}_{\mathcal{E}}$ we employ Lemma 3.3.1:

$$\beta(\mathrm{H}_{\mathcal{E}}) \geq e\Big(1 - \sum_{i=1}^{l} \frac{1-\mu_i}{\kappa-\mu_i}\Big).$$

The $\mathcal{E}$-degree of a vertex in S_i (resp., B_i) is at least d_0/κ (resp., $d_0(\kappa-1)/\kappa$). Hence

$$\begin{aligned} |S_i| = |B_i| + |N_i| &\leq \mathcal{E}(N_i)\frac{\kappa}{d_0} + \mathcal{E}(B_i)\frac{\kappa(1-\mu_i)}{d_0(\kappa-1)} \\ &\leq \frac{\kappa e}{d_0}\Big(\frac{1-\mu_i}{\kappa-1} + \mu_i\Big)N. \end{aligned}$$

Using the last two inequalities in (2.3), we obtain

$$e\Big(1 - \sum_{i=1}^{l} \frac{1-\mu_i}{\kappa-\mu_i}\Big) \leq \Big(\frac{\kappa e}{\delta_0}\Big)^l \prod_{i=1}^{l} \Big(\frac{1-\mu_i}{\kappa-1} + \mu_i\Big) + \varepsilon\frac{\kappa e}{\delta_0}.$$

To contradict this, let

$$e < \Big(\frac{\delta_0}{\kappa}\Big)^{l/(l-1)} \left\{ \frac{1 - \sum_{i=1}^{l} \frac{1-\mu_i}{\kappa-\mu_i} - \varepsilon\kappa/\delta_0}{\prod_{i=1}^{l}(\frac{1-\mu_i}{\kappa-1} + \mu_i)} \right\}^{1/(l-1)}.$$

We again bound the terms that involve ε from below by a multiplicative term $1-\alpha'$. Optimizing on all possible values of μ_i gives $\mu_i = \mu$ for all $i = 1 \ldots l$, whereupon the expression on the right can be replaced by $(1-\alpha)\delta_0^{l/(l-1)} f(\mu,\kappa)$. The proof is thus complete. □

Numerically, the first values of the decoding radius ρ given by Theorem 3.3.3 are

$$\rho \geq \frac{\delta_0^{3/2}}{5.94} \text{ for } l=3 \qquad \rho \geq \frac{\delta_0^{4/3}}{6.46} \text{ for } l=4$$

attained for κ satisfying $(\kappa-1)^{-l} = 1 - l/\kappa$ and $\mu = 0$ or 1. These results should be compared with the estimate of the designed distance of codes given by (2.4): one notices that we are now correcting a constant proportion of the designed distance which was our goal. This is also an advance over the earlier result of [19] given in (3.1).

Can one obtain better bounds for the decoding radius? In principle, it is possible to obtain further improvements by introducing *multiple* thresholds instead of the

single decoding threshold $\theta = d_0/\kappa$, and approach $\rho = \delta/2$ by increasing their number. However we shall only be able to claim that using one of the multiple thresholds reduces the number of errors for one of the subprocedures, but we shall not be able to discern which decoding threshold achieves that. This will result in yet another layer of parallelism, further increasing the value of the constant in the decoding complexity. We will not pursue this line of research further here. A remaining challenge is to decode up to half the designed distance with an iterative decoding procedure of reasonable complexity.

CHAPTER 4

Codes on Graphs: Correctable Errors

4.1 Introduction

In this chapter we are interested in estimating the average number of errors correctable with the ensemble of codes on graphs. The work in this direction originates in the works of Gallager [52] and Zyablov and Pinsker [107] who showed that random LDPC codes of growing length can correct a nonvanishing fraction of errors. Recently the decoding algorithm of [107] was studied by Burshtein [26] who derived an improved estimate of the number of correctable errors compared to [107] and by Zyablov et al. [106] who provided estimates of the number of errors under the assumption of local single error-correcting (Hamming) codes.

As is well known (e.g., [63]), graphs with high expansion and random graphs share many properties that can be used to prove estimates of error correction. Regarding the proportion of errors corrected by graph codes under iterative decoding, we note one difference between (generalized) LDPC codes on random graphs and explicit constructions based on the graph spectrum. The explicit constructions based on regular graphs depend on the difference between the largest and the second largest eigenvalue of the graph (the "spectral gap"). For this reason, one is forced to rely on local codes with rather large minimum distance d_0, for instance, d_0 greater than the square root of the degree n of the graph. Even though in the construction of [93] and later works, n is kept constant, this effectively rules out of consideration local codes with small minimum distance such as the Hamming codes and the like. The square root restriction is implied by the spectral gap of regular bipartite graphs, and is the best possible owing to the Alon-Boppana bound for graph spectra [86]. The purpose of the present chapter is to lift this limitation on the distance d_0 by switching from graphs with a large spectral gap to random graphs.

In this chapter we obtain new estimates of the number of correctable errors for random ensembles of bipartite-graph and hypergraph codes under iterative decoding. The first part of the chapter is devoted to codes on regular bipartite graphs. To construct long graph codes, we assume that the degree of the graph is fixed and the

number of vertices in both parts approaches infinity. Assuming that local constraint codes are used to correct 2 or more errors, we show that almost all codes in the ensemble of graph codes are capable of correcting all error patterns of weight that forms a constant fraction of the code length. This is a much less restrictive assumption on the local codes than the one taken in earlier works on the decoding of graph codes [17, 104].

We then observe that if the degree of the graph is allowed to increase then graph codes with local codes of constant distance do not correct a linearly growing number of errors under the proposed iterative decoding. This motivates us to study graph codes with long local codes correcting a growing number of errors that forms a fixed proportion of the degree. The results obtained in this case parallel earlier theorems for product codes and graph codes based on the spectral gap.

In the second part of the chapter we establish similar results for codes on hypergraphs, showing that a constant proportion of errors is corrected by an iterative decoding algorithm. Constructing the code ensemble based on regular hypergraphs of a fixed degree, we show that they contain codes capable of correcting a constant proportion of errors. The proof involves no assumptions on the distance of the local codes; in particular, we show that networks of Hamming codes correct a fixed proportion of errors under iterative decoding. This fact was previously proved by Tanner [96] under the assumption that the underlying graph is a tree. This assumption is not needed in our results. As in the case of the graph ensemble, we also perform the analysis of the decoding algorithm for the case of growing degree, finding the proportion of errors correctable with hypergraph codes based on long local codes.

The material presented in this chapter is published in [12].

4.1.1 Code ensembles

For the bipartite graph codes we consider the ensemble of codes $\mathscr{C}_2(2, A)$ described in Definition 2.2.1 of Chapter 2. Here $A[n, R_0 n, d_0]$ is the fixed linear binary local code. Suppose that the $[N, RN]$ code $\mathcal{C}(\mathrm{G})$ is constructed by associating it with a graph $\mathrm{G}(V_1 \cup V_2, E)$, $|V_1| = |V_2| = m$, $|E| = mn = N$, sampled from the set of graphs defined by a random permutation on N elements which establishes how the edges originating in V_1 are connected to the vertices in V_2.

Generalizing, we consider for hypergraph codes the ensemble of codes $\mathscr{C}_2(l, A)$ from Definition 2.2.1. A code $\mathcal{C}$ in this ensemble is constructed on an l-partite n-regular uniform hypergraph $\mathrm{H} = (V, E)$, $V = V_1 \cup \cdots \cup V_l$, $|V_1| = \cdots = |V_l| = m$, $|E| = mn = N$, which is constructed by sampling a random hypergraph from the set of hypergraphs defined by $l - 1$ independent random permutations on N elements. For $i = 1, 2, \ldots, l-1$, the ith permutation accounts for the placement of edges between parts V_1 and V_{i+1} of H. As earlier $A[n, R_0 n, d_0]$ is the fixed linear binary local code.

Recall that the rate R of the codes $\mathcal{C} \in \mathscr{C}_2(l, A)$ satisfies $R \geq lR_0 - (l-1), l =$

$2, 3, \ldots$. Denote by $d_{\mathscr{C}} = d(\mathscr{C}_2(l, A))$ the average value of the minimum distance of codes in the hypergraph ensemble and let

$$\delta = \delta(\mathscr{C}_2) \triangleq \liminf_{N\to\infty} \frac{d_{\mathscr{C}}}{N}. \tag{4.1}$$

In Chapter 2 we discussed ways to bound the value of δ from below using the distribution of distances in the local code A (suggested in [25, 74]). In particular, we showed that $\delta(\mathscr{C}_2) > 0$ if the local distance d_0 satisfies $d_0 > l/(l-1)$. For the bipartite graph ensemble $\mathscr{C}_2(2, A)$ this implies that $d_0 \geq 3$; i.e., there exist codes in the ensemble that are asymptotically good (have non-vanishing rate and relative distance) when the local codes correct one or more errors. For hypergraphs with $l = 3$ or more parts any local codes (without repeated vectors) account for an asymptotically good ensemble. An explicit lower bound for $\delta(\mathscr{C}_2)$ that depends only on l and d_0 is also discussed in Chapter 2, see Theorem 2.4.5 in there and rephrased in 4.3.8 below. For the case when n is large and $d_0 = \delta_0 n$, a lower estimate of $\delta(\mathscr{C}_2)$ is given by the solution for x of the following equation from Corollary 2.4.6:

$$\frac{\mathrm{h}(x)}{x} = \frac{l}{l-1}\frac{\mathrm{h}(\delta_0)}{\delta_0}. \tag{4.2}$$

4.2 Decoding algorithms for graph codes

4.2.1 Decoding for the ensemble $\mathscr{C}_2(2, A)$

In our estimates of the number of correctable errors for the ensemble we rely upon the Algorithm I described in the Chapter 3.

4.2.2 Decoding for the ensemble $\mathscr{C}_2(l, A)$

For the hypergraph ensemble $\mathscr{C}_2(l, A)$ we use the decoding algorithm proposed in Chapter 3. Although the main steps remain same, we modify the algorithm at certain points to accommodate the special setting of ensemble $\mathscr{C}_2(l, A)$. It is described below.

Let $\mathcal{C} \in \mathscr{C}_2(l, A)$ be a code and let $\mathrm{H}(V, E), V = V_1 \cup \cdots \cup V_l$ be the graph associated with it. For every $i = 1, 2 \ldots, l$ we will define an i-th subprocedure that decodes the local code A on every vertex in the part V_i. Suppose that a vector $\boldsymbol{u} \in \{0, 1\}^N$ is associated with the edges $e \in E$. Let $v_{i,1}, \ldots, v_{i,m}$ be the vertices in the part V_i of H and let $\boldsymbol{u}_{i,1} = \boldsymbol{u}(v_{i,1}), \ldots, \boldsymbol{u}_{i,m} = \boldsymbol{u}(v_{i,m})$ be the m subvectors obtained from $\boldsymbol{u}$ upon permuting its coordinates according to the order of edges in V_i and projecting it on the vertices $v_{i,1}, \ldots, v_{i,m}$. In other words, the vector $(\boldsymbol{u}_{i,1}, \ldots, \boldsymbol{u}_{i,m})$ is obtained from $\boldsymbol{u}$ using the permutation that establishes edge connections between parts V_1 and V_i. The ith subprocedure replaces the vector

$(\boldsymbol{u}_{i,1}, \ldots, \boldsymbol{u}_{i,m})$ with the vector $(\psi_{A,t}(\boldsymbol{u}_{i,1}), \ldots, \psi_{A,t}(\boldsymbol{u}_{i,m}))$. $\psi_{A,t}$ is the bounded-distance decoder of the local code defined in section 3.2.

The algorithm proceeds in iterations. Let $\boldsymbol{y} \in \{0,1\}^N$ be the received vector. Denote by $Y^{(j)}$ the set of estimates of the transmitted codeword (i.e., the set of N-vectors) stored at the vertices of H before the jth iteration $j = 1, 2, \ldots$. After each iteration, this set is formed as the union of the vectors obtained upon decoding of the vertices in the ith part, $i = 1, \ldots, l$. Decoding begins with setting $Y^{(1)} = \{\boldsymbol{y}\}$. After the first iteration we obtain l potentially different vectors (one for each subprocedure) which form the current estimates of the transmitted vector. These vectors form the sets $Y_i^{(2)}, i = 1, \ldots, l$. In the next iteration each subprocedure will have to be applied to each of the l outcomes of the preceding iteration. Proceeding in this way, we observe that $|Y_i^{(j)}| \leq l^{j-1}$.

This algorithm, called Algorithm II below, will only be applied for a constant number s of iterations until we can guarantee that at least one subprocedure has reduced the number of errors to a specified proportion, say from $\gamma_0 N$ to some $\gamma_1 N, \gamma_1 < \gamma_0$. We then let another algorithm take over and decode all the l^s candidates. Any low-complexity decoder of graph codes that removes an arbitrarily small positive fraction of errors γ_1 will do at this stage. This is because taking the proportion of errors from γ_0 to $\gamma_1 > 0$ can be accomplished in a constant number s of steps, so the number of candidates that this decoder has to handle is at most l^s and does not depend on N.

For the case of local codes correcting $t \geq 2$ errors we let this algorithm to be the decoding algorithm of bipartite-graph codes (Algorithm I), making sure that γ_1 is below the proportion of errors that are necessarily corrected by this algorithm for the ensemble $\mathscr{C}_2(2, A)$. This is possible because, leaving any two parts of the original hypergraph H to form a bipartite graph G, we obtain a random code from the ensemble $\mathscr{C}_2(2, A)$ which with high probability (over the ensemble) will remove all the residual errors from at least one candidate estimate. For $t = 1$ this approach fails for the reasons discussed in the next section, so we resort to a procedure in [106] that corrects a small linear fraction of errors for single-error-correcting Hamming codes.

Upon performing the described procedure we obtain a list of at most l^s candidate codewords of the code C. The final decoding result is found by choosing the codeword from this list closest to $\boldsymbol{y}$ by the Hamming distance.

4.3 Number of correctable errors

4.3.1 The ensemble $\mathscr{C}_2(2, A)$

Let $\mathcal{C} \in \mathscr{C}_2(2, A)$ be a code and let $\mathrm{G}(V, E)$ be the graph associated with it. For a given subset of vertices $S \subset V_i, i = 1, 2$ and a vertex v denote by $\deg_S(v)$ the number of edges between v and S. Let $T_r(S) = \{v \in V : \deg_S(v) \geq r + 1\}$,

where $r \in \{0, \dots, n-1\}$ is an integer.

Below $\mathrm{h}(\boldsymbol{z})$ denotes the entropy of the probability vector $\boldsymbol{z} = (z_0, \dots, z_n) \in \mathbb{R}^{n+1}$, i.e., $\mathrm{h}(\boldsymbol{z}) = -\sum_{i=0}^{n} z_i \log z_i$. As before, in the particular case of $n = 1$ we write $\mathrm{h}(z)$ instead of $\mathrm{h}(z, 1-z)$.

Let $t \geq 0$ be any integer such that $2t + 1 \leq d_0$. The calculation in this section is based on the following simple observation.

Proposition 4.3.1. *Suppose that for all $S \subset V_i, i = 1, 2, |S| \leq \sigma m, \sigma \in (0, 1)$, there exists $\varepsilon > 0$ such that $|T_t(S)| \leq |S| - \varepsilon m$. Then any $\sigma t m = \sigma t(N/n)$ errors will be corrected by Algorithm I in $O(\log m)$ iterations.*

Proof. Suppose that no more than $\sigma t m$ errors occurred in the channel. Let S_i be the set of vertices that are decoded incorrectly in iteration i of Algorithm I. The assumption of the proposition implies that $|S_{i+1}| \leq |S_i|(1 - \varepsilon/\sigma)$, so $O(\log m)$ iterations suffice to remove all the errors. □

Define

$$F_{n,t}(\sigma) = \mathrm{h}(\sigma) - \sigma n \log x + \sigma \log \sum_{i=t+1}^{n} \binom{n}{i} x^i + (1-\sigma) \log \sum_{i=0}^{t} \binom{n}{i} x^i, \tag{4.3}$$

where $x > 0$ is found from the equation

$$\sum_{i=0}^{t} \sum_{j=t+1}^{n} \binom{n}{i} \binom{n}{j} (\sigma(n-j) - i(1-\sigma)) x^{i+j-t-1} = 0. \tag{4.4}$$

Let $\mathcal{Z}_n = \{\boldsymbol{z} \in [0,1]^{n+1} : \sum_{i=0}^{n} z_i = 1\}$ be the $(n+1)$-dimensional probability simplex.

The main result of this section is given by the next theorem.

Theorem 4.3.2. *Let $A[n, R_0 n, d_0]$ be the local code, let $m \to \infty$, and let $2 \leq t < d_0/2$. All codes in the ensemble $\mathscr{C}_2(2, A)$ except for an exponentially small (in N) proportion of them correct any combination of errors of weight $\sigma t m$ in $O(\log m)$ iterations of Algorithm I, where $0 < \sigma < \sigma_0$ and σ_0 is the smallest positive root of the equation*

$$F_{n,t}(\sigma) = (n-1)\,\mathrm{h}(\sigma).$$

Remark 2. The case of local codes with $t = 1$ is excluded from this theorem because G with high probability contains a large number of 4-cycles, which means that correcting single error at every vertex does not ensure overall convergence of the decoding. Indeed, if two vertices are affected by two errors each, and the corresponding 4 edges form a cycle, then the decoder will loop indefinitely without approaching the correct decision. The theorem is still valid in this case, but gives $\sigma_0 = 0$.

Proof. We need to verify the assumption of Proposition 4.3.1. Let $S \subset V_1, |S| = \sigma m$ and let $m_i = |\{v \in V_2 : \deg_S(v) = i\}|, i = 1, \dots, n$. Clearly,

$$\sum_{i=1}^{n} m_i \le m, \ \sum_{i=t+1}^{n} m_i = |T_t(S)|, \ \sum_{i=1}^{n} i m_i = |S| n.$$

Let us compute the probability (over the choice of G) that $|T_t(S)| \ge (\sigma - \varepsilon)m$. Let $\boldsymbol{\mu} = (m_1, \dots, m_n)$ be a vector with nonnegative integer components, let

$$M_\varepsilon(t, \sigma) \ = \ \{\boldsymbol{\mu} \ : \ \sum_{i=1}^{n} m_i \ \le \ m, \sum_{i=1}^{n} i m_i \ = \ \sigma N, \sum_{i=t+1}^{n} m_i \ \ge \ (\sigma - \varepsilon)m\},$$

and let $\binom{m}{\boldsymbol{\mu}}$ denote the number of choices of subsets of size $m_1, \dots, m_n$ out of a set of size m. We have

$$\Pr(|T_t(S)| \ge |S| - \varepsilon m) = \frac{1}{\binom{N}{\sigma N}} \sum_{\boldsymbol{\mu} \in M_\varepsilon(t,\sigma)} \binom{m}{\boldsymbol{\mu}} \prod_{i=1}^{n} \binom{n}{i}^{m_i}. \qquad (4.5)$$

Let $\mathscr{L}_1(s)$ denote the event that V_1 contains a subset $S, |S| = s$ for which $|T_t(S)| \ge |S| - \varepsilon m$. We have

$$\Pr(\mathscr{L}_1(\sigma m)) \le \binom{m}{\sigma m} \Pr(|T_t(S)| \ge |S| - \varepsilon m)$$

and

$$\Pr\Big(\bigcup_{i=1}^{\sigma m} \mathscr{L}_1(i)\Big) \le m \Pr(\mathscr{L}_1(\sigma m)).$$

Denote by $\mathscr{L}_2(\sigma)$ an analogous event with respect to V_2. Then

$$\Pr\Big(\bigcup_{i=1}^{\sigma m} (\mathscr{L}_1(i) \cup \mathscr{L}_2(i))\Big) \le \frac{2m\binom{m}{\sigma m}}{\binom{N}{\sigma N}} \sum_{\boldsymbol{\mu} \in M_\varepsilon(t,\sigma)} \binom{m}{\boldsymbol{\mu}} \prod_{i=1}^{n} \binom{n}{i}^{m_i}. \qquad (4.6)$$

Letting L to be the logarithm of the left-hand side divided by m and omitting $o_m(1)$ terms, we obtain the estimate $L \le n^{-1} \bar{F}_{n,t}(\sigma)$, where

$$\bar{F}_{n,t}(\sigma) = -(n-1)\,\mathrm{h}(\sigma) + \max_{\boldsymbol{z} \in \mathscr{M}'_\varepsilon(t,\sigma)} \Big(\mathrm{h}(\boldsymbol{z}) + \sum_{i=1}^{n} z_i \log \binom{n}{i}\Big),$$

where

$$\mathscr{M}'_\varepsilon(t, \sigma) = \Big\{\boldsymbol{z} \in \mathcal{Z}_n : \sum_{i=1}^{n} i z_i = \sigma n, \ \sum_{i=t+1}^{n} z_i \ge \sigma - \varepsilon\Big\}$$

and $z_i = m_i/m, z_0 = (m - \sum m_i)/m$.

The rest of the proof is concerned with the evaluation of the above maximum. Define

$$g(\boldsymbol{z}) = \mathrm{h}(\boldsymbol{z}) + \sum_{i=1}^{n} z_i \log \binom{n}{i} \tag{4.7}$$

$$\bar{\sigma} = \sup\{\sigma > 0 : \bar{F}_{n,t}(y) < 0 \text{ for all } 0 \le y < \sigma\}.$$

As long as $\sigma < \bar{\sigma}$, the probability of not being able to correct $\sigma t m$ errors with a random code from the considered ensemble approaches zero. Thus, we need to find the maximum $\max_{\boldsymbol{z} \in \mathscr{M}'_\varepsilon(t,\sigma)} g(\boldsymbol{z})$ for all $\sigma \in [0, \bar{\sigma})$. The proof will be accomplished in the next three steps. Since ε will be assumed arbitrarily small, we will omit it from our considerations and write $\mathscr{M}'$ instead of $\mathscr{M}'_\varepsilon$.

1. We find the point $\boldsymbol{z}^*$ that gives the maximum of $g(\boldsymbol{z})$ without the constraint $\sum_{i=t+1}^{n} z_i \ge \sigma$.

2. Next we show that for $0 \le \sigma < \bar{\sigma}$, the point $\boldsymbol{z}^* \notin \mathscr{M}'$, and therefore the maximum over $\mathscr{M}'$ is attained on the boundary, i.e., we can replace $\mathscr{M}'$ with

$$\mathscr{M}(t,\sigma) = \Big\{\boldsymbol{z} \in \mathcal{Z}_n : \sum_{i=1}^{n} i z_i = \sigma n,\ \sum_{i=t+1}^{n} z_i = \sigma\Big\}.$$

3. Finally we compute the value of the maximum.

Step 1. Without the constraint $\sum_{i=t+1}^{n} z_i \ge \sigma$ the maximum is easily computed. Indeed, the proportion of edges incident to the vertices in S out of the N edges of G is σ, so the fraction of vertices with S-degree i should be close to $z_i^*(\sigma) = \binom{n}{i}\sigma^i(1-\sigma)^{n-i}$. Thus, the coordinates of the maximizing point $\boldsymbol{z}^* = \boldsymbol{z}^*(\sigma)$ are $z_i^*, i = 1, \ldots, n; z_0 = 1 - \sum_i z_i^*$, and

$$g(\boldsymbol{z}^*) = n\,\mathrm{h}(\sigma).$$

Slightly more formally, note that $\boldsymbol{z}^*$ is the unique stationary point of the function $g(\boldsymbol{z})$, and that this function is strictly concave in $\boldsymbol{z}$. Therefore, $\boldsymbol{z}^*$ is a unique maximum of $g(\boldsymbol{z})$ on $\mathcal{Z}_n$, and the function $g(\boldsymbol{z})$ grows in the direction $\boldsymbol{z}^* - \boldsymbol{z}$ for any $\boldsymbol{z} \in \mathcal{Z}_n$.

Step 2. Suppose that $0 \le \sigma \le \bar{\sigma}$. Observe that $p(\sigma) \triangleq \sum_{i=t+1}^{n} z_i^* = \Pr(X \ge t+1)$, where X is a $(\sigma, 1-\sigma)$ binomial random variable. This probability is monotone increasing on σ for $\sigma \in [0,1]$, and $p(0) = p'(0) = 0$. Thus for $\sigma \in [0, \alpha)$ where α is the smallest positive root of $\sum_{i=t+1}^{n} z_i^*(\sigma) = \sigma$, we have

$$\sum_{i=t+1}^{n} z_i^* = \sum_{i=t+1}^{n} \binom{n}{i}\sigma^i(1-\sigma)^{n-i} < \sigma,$$

and so the point $\boldsymbol{z}^*(\sigma) \notin \mathscr{M}'(t,\sigma)$. Our claim will follow if we show that $\bar{\sigma} < \alpha$. This is indeed the case because for $0 \le \sigma < \bar{\sigma}$,

$$\max_{\boldsymbol{z} \in \mathscr{M}'(t,\sigma)} g(\boldsymbol{z}^*(\sigma)) < (n-1)\,\mathrm{h}(\sigma).$$

On the other hand, $g(\boldsymbol{z}^*(\alpha)) = n\,\mathrm{h}(\alpha)$. This establishes that the maximum of $g(\boldsymbol{z})$ on $\boldsymbol{z} \in \mathscr{M}'$ is attained on the hyperplane $\sum_{i=t+1}^{n} z_i = \sigma$.

Step 3. To compute the maximum of $g(\boldsymbol{z})$ on $\boldsymbol{z}$, let us form the Lagrangian

$$U(\boldsymbol{z}, \tau_1, \tau_2) = \mathrm{h}(\boldsymbol{z}) + \sum_{i=1}^{n} z_i \log \binom{n}{i} + \tau_1 \Big(\sum_{i=1}^{n} i z_i - \sigma n \Big) + \tau_2 \Big(\sum_{i=t+1}^{n} z_i - \sigma \Big).$$

Setting $\nabla U = 0$ and $\tau_1 = \log x, \tau_2 = \log y$, we find that

$$z_i = \begin{cases} \binom{n}{i} x^i D & \text{if } 0 \le i \le t \\ \binom{n}{i} y x^i D & \text{if } t < i \le n, \end{cases}$$

where we have denoted

$$D = \Big[\sum_{i=0}^{t} \binom{n}{i} x^i + y \sum_{i=t+1}^{n} \binom{n}{i} x^i \Big]^{-1}.$$

Adding these equations together, we find conditions for x and y:

$$\sigma = Dy \sum_{i=t+1}^{n} \binom{n}{i} x^i$$

$$\sigma n = D \Big(\sum_{i=0}^{t} i \binom{n}{i} x^i + y \sum_{i=t+1}^{n} i \binom{n}{i} x^i \Big).$$

Once y is eliminated from the last two equations, we obtain the condition (4.4) for x. Finally, substituting the found values of $z_i, i = 1, \ldots, n$ into $g(\boldsymbol{z})$, we find that the maximum evaluates to the expression $F_{n,t}(\sigma)$ given in (4.3) (and therefore, $\bar{\sigma} = \sigma_0$). Since we seek to obtain a value $L < 0$, the boundary condition for the proportion of correctable errors is obtained by setting $L = 0$. This concludes the proof. □

Example: Using Theorem 4.3.2 together with (4.3) we can compute the proportion of errors corrected by codes in the ensemble $\mathscr{C}_2(2, A), m \to \infty$ for several choices of the local code A. For instance, taking A to be the binary Golay code of length $n = 23$ we find $\sigma_0 \approx 0.0048586$ and therefore, the proportion of correctable errors is $\frac{\sigma_0 t}{n} \approx 0.00063$. Similarly, for the 2-error-correcting $[n = 31, k = 21]$ BCH code we find $\sigma_0 \approx 0.000035$ and $\frac{\sigma_0 t}{n} \approx 0.0000023$.

To underscore similarities with the results obtained for product codes and their later variations including graph codes (e.g., [104]) we compute the proportion of errors correctable with codes from the ensemble $\mathscr{C}_2(2, A)$ in the case of large n.

Proposition 4.3.3. *Let $t = \tau n$. Then the ensemble $\mathscr{C}_2(2, A)$ contains codes that correct $\sigma\tau N$ errors for any $\sigma \le \sigma_0$, where σ_0 is given by*

$$\sigma_0 = \sup\Big\{\sigma > 0 : \forall_{0<x<\sigma}\ (1-x)\,\mathrm{h}\Big(\frac{x(1-\tau)}{1-x}\Big) + x\,\mathrm{h}(\tau) + \varepsilon_n < \mathrm{h}(x)\Big\}$$

where $\varepsilon_n = (1+\log n)/n$.

Proof. Referring to the notation of the previous proof, let us evaluate the asymptotic behavior of the exponent L of the probability in (4.6). Since $\mathrm{h}(\boldsymbol{z}) \le \log n$, we have

$$n^{-1}\bar{F}_{n,t}(\sigma) \ \le\ -\,\mathrm{h}(\sigma) + n^{-1} \max_{\boldsymbol{z}\in\mathscr{M}(\tau n,\sigma)} \sum_{i=0}^{n} z_i \log \binom{n}{i} + n^{-1}(1+\log n).$$

Next,

$$\begin{aligned}\frac{1}{n}\sum_{i=0}^{n} z_i \log\binom{n}{i} &\le \sum_i z_i\,\mathrm{h}\Big(\frac{i}{n}\Big)\\ &= (1-\sigma)\sum_{i=0}^{t}\frac{z_i}{1-\sigma}\,\mathrm{h}\Big(\frac{i}{n}\Big) + \sigma\sum_{i=t+1}^{n}\frac{z_i}{\sigma}\,\mathrm{h}\Big(\frac{i}{n}\Big)\\ &\le (1-\sigma)\,\mathrm{h}\Big(\frac{\sum_{i=1}^{t} i z_i}{(1-\sigma)n}\Big) + \sigma\,\mathrm{h}\Big(\frac{\sum_{i=t+1}^{n} i z_i}{\sigma n}\Big).\end{aligned}$$

Let $y = n^{-1}\sum_{i=t+1}^{n} i z_i$, then for any $\boldsymbol{z} \in \mathscr{M}(\tau n, \sigma)$ we have

$$\frac{1}{n}\sum_{i=0}^{n} z_i \log\binom{n}{i} \le \max_{\tau\sigma\le y\le\sigma}\Big\{(1-\sigma)\,\mathrm{h}\Big(\frac{\sigma-y}{1-\sigma}\Big) + \sigma\,\mathrm{h}\Big(\frac{y}{\sigma}\Big)\Big\}.$$

The function on the right-hand side of this inequality is concave. Its global maximum equals $\mathrm{h}(\sigma)$ and is attained for $y = \sigma^2$. Thus, assuming that $\sigma < \tau$, we conclude that the constrained maximum occurs for $y = \tau\sigma$, which gives the following bound on $n^{-1}\bar{F}_{n,t}(\sigma)$:

$$n^{-1}\bar{F}_{n,t}(\sigma) \le -\,\mathrm{h}(\sigma) + (1-\sigma)\,\mathrm{h}\Big(\frac{\sigma(1-\tau)}{1-\sigma}\Big) + \sigma\,\mathrm{h}(\tau) + \varepsilon_n.$$

As long as the right-hand side of the this inequality is negative, the previous proof implies that the code corrects all errors of multiplicity up to $\sigma\tau N$. □

From the expression of this proposition we observe that (as $n \to \infty$) the value of σ_0 approaches τ, so the ensemble $\mathscr{C}_2(2, A)$ contains codes that correct up to a τ^2 proportion of errors, where $\tau n = d_0/2$ is the error-correcting capability of the code A. This result parallels the product bound on the error-correcting radius of direct product codes. As in the case of product and expander codes (e.g., [17]), the proportion of correctable errors can be improved from $\tau^2 = (d_0/(2n))^2$ by using a more powerful decoding algorithm.

4.3.2 The ensemble $\mathscr{C}_2(l, A)$

In this section we first state a sufficient condition for the existence of at least one subprocedure within each step of Algorithm II that reduces the number of errors, and then perform the analysis of random hypergraphs to show that with high probability this condition is satisfied. Overall this will show that the number of errors in at least one of the candidates in the list generated after a few iterations is reduced to a desired level.

Denote by $E(v)$ the set of edges incident to a vertex $v \in V$. Let $\mathcal{C} \in \mathscr{C}_2(l, A)$ be a code and let $\mathrm{H}(V, E)$ be its associated graph. Let $\mathcal{E} \subset E$ be the set of errors at the start of some iteration of the algorithm. The next set of arguments will refer to this iteration. Let $G_i = \{v \in V_i : |E(v) \cap \mathcal{E}| \leq t\}$ be the set of vertices such that each of them is incident to no more than t edges from $\mathcal{E}$ (such errors will be corrected upon one decoding). Let $B_i = \{v \in V_i : |E(v) \cap \mathcal{E}| \geq d_0 - t\}$ be the set of vertices that can introduce errors after one decoding iteration. Note that each of such vertices introduces at most t errors.

The main condition for successful decoding is given in the next lemma.

Lemma 4.3.4. *Assume that for every $\mathcal{E} \subset E, |\mathcal{E}| \leq \gamma N$ there exists $i = i(\mathcal{E}), 1 \leq i \leq l$ such that $|\mathcal{E}(G_i)| \geq t|B_i| + \varepsilon N$, where $\mathcal{E}(G_i)$ is the set of edges of $\mathcal{E}$ incident to the vertices of G_i and $\varepsilon > 0$. Then for any $0 < \beta < \gamma$, Algorithm II will reduce any γN errors in the received vector to at most βN errors in $c(\beta, \gamma, \varepsilon)$, iterations where c is a constant independent of N.*

Proof. We need to prove that at least one of the subprocedures will find a vector with no more than βN errors after a constant number of iterations. In any given iteration by the assumption of the lemma there exists a component V_i for which the ith subprocedure will decrease the count of errors by $|\mathcal{E}(G_i)| - tB_i \geq \varepsilon N$. Thus, in each iteration there exists a subprocedure that reduces the number of errors by a positive fraction. □

Next we show that the assumption of Lemma 4.3.4 holds with high probability over the ensemble. Consider the function

$$\tilde{F}_{n,t}(\gamma) = \max_{\boldsymbol{z} \in \mathscr{M}(t,\gamma)} \Big(\mathrm{h}(\boldsymbol{z}) + \sum_{i=0}^{n} z_i \log \binom{n}{i} \Big), \tag{4.8}$$

where in this section the region $\mathscr{M}(t, \gamma)$ will be as follows:

$$\mathscr{M}(t, \gamma) = \Big\{ \boldsymbol{z} \in \mathcal{Z}_n : \sum_{i=1}^{n} i z_i = \gamma n, \sum_{i=1}^{t} i z_i = \sum_{i=d_0-t}^{n} t z_i \Big\}. \tag{4.9}$$

Lemma 4.3.5. *Let $m \to \infty$ and let*

$$\gamma_0 = \sup\{x > 0 : \forall_{0<\gamma\leq x} \, (l/n)\tilde{F}_{n,t}(\gamma) < (l-1)\,\mathrm{h}(\gamma)\}. \tag{4.10}$$

A hypergraph from the ensemble of l-partite uniform n-regular hypergraphs with probability $1-2^{-\Omega(N)}$ has the property that for all $\mathcal{E}\subset E, |\mathcal{E}|<\gamma_0 N$, and some $\varepsilon>0$, the inequality $|\mathcal{E}(G_i)|\geq t|B_i|+\varepsilon N$ holds for at least one $i\in\{1,\dots,l\}$.

Proof. Let $\mathcal{E}\subset E, |\mathcal{E}|=\gamma N$. Let $m_i=|\{v\in V_1: |E(v)\cap\mathcal{E}|=i\}|, i=1,\dots,n$. Clearly $|\mathcal{E}(G_1)|=\sum_{i=0}^{t} i m_i$ and $|B_1|=\sum_{i=d_0-t}^{n} m_i$. We have

$$p \triangleq \Pr(|\mathcal{E}(G_i)| \leq t|B_i| + \varepsilon N) = \frac{1}{\binom{N}{\gamma N}} \sum_{\boldsymbol{\mu}\in M_\varepsilon(t,\gamma)} \binom{m}{\boldsymbol{\mu}} \prod_{i=0}^{n} \binom{n}{i}^{m_i},$$

where $\boldsymbol{\mu}=\{m_1,\dots,m_n\}$,

$$M_\varepsilon(t,\gamma)=\{\boldsymbol{\mu}\in(\mathbb{Z}_+\cup 0)^n : \sum_{i=1}^{n} m_i \leq m, \sum_{i=1}^{n} i m_i = \gamma N, \sum_{i=1}^{t} i m_i \leq \sum_{i=d_0-t}^{n} t m_i + \varepsilon N\}.$$

Denote by $\mathscr{L}(\mathcal{E})$ the event that for a given subset $\mathcal{E}\subset E, |\mathcal{E}|=\gamma N$ no part V_i of H satisfies the assumption of Lemma 4.3.4. Then $\Pr(\mathscr{L}(\mathcal{E}))=p^l$ and

$$\Pr\{\exists\mathcal{E}:\ (|\mathcal{E}|\leq\gamma N)\wedge(\mathscr{L}(\mathcal{E}))\}\leq N\binom{N}{\gamma N}p^l.$$

Letting L to be the logarithm of the left-hand side of this inequality divided by N and omitting $o_N(1)$ terms, we obtain

$$L\leq -(l-1)\,\mathrm{h}(\gamma)+\frac{l}{n}\max_{\boldsymbol{z}\in\mathscr{M}'(t,\gamma)} g(\boldsymbol{z}), \tag{4.11}$$

where $g(\boldsymbol{z})$ is defined in (4.7),

$$\mathscr{M}'(t,\gamma)=\{\boldsymbol{z}\in\mathcal{Z}_n : \sum_{i=1}^{n} i z_i=\gamma n, \sum_{i=1}^{t} i z_i \leq \sum_{i=d_0-t}^{n} t z_i\}$$

and $z_i=m_i/m$ (as in the previous section, we have omitted ε which can be made arbitrarily small).

The proof will be complete if we show that the optimization region $\mathscr{M}'$ can be replaced by $\mathscr{M}$. For that we follow the logic of the second part of the proof of Theorem 4.3.2. As before, the maximum of $g(\boldsymbol{z})$ without the constraint $\sum_{i=1}^{t} i z_i \leq \sum_{i=d_0-t}^{n} t z_i$ is attained at the point $\boldsymbol{z}^*(\gamma)=(z_0^*,z_1^*,\dots,z_n^*)\in\mathcal{Z}_n$, where

$$z_i^*=z_i^*(\gamma)=\binom{n}{i}\gamma^i(1-\gamma)^{n-i},\quad i=1,\dots,n.$$

We need to show that as long as $0 \le \gamma < \gamma_0$, the point $\boldsymbol{z}^* \notin \mathscr{M}'(t,\gamma)$. By concavity of the objective function and the optimization region, this will imply that the maximum is on the boundary. As before, it is possible to show that in the neighborhood of $\gamma = 0$,

$$\sum_{i=1}^{t} i z_i^* > \sum_{i=d_0-t}^{n} t z_i^*.$$

and thus for $\gamma < \beta$, where β is the smallest positive root of $\sum_{i=1}^{t} i z_i^* = \sum_{i=d_0-t}^{n} t z_i^*$, the point $\boldsymbol{z}^*(\gamma) \notin \mathscr{M}'(t,\gamma)$. Let

$$\bar{\gamma} = \sup\{\gamma : \ \forall\, 0 < x < \gamma, \ \text{rhs of (4.11)} < 0\}.$$

We note that for all $\gamma \le \bar{\gamma}$,

$$\max_{\boldsymbol{z} \in \mathscr{M}'(t,\sigma)} g(\boldsymbol{z}) < (l-1) n \, \mathrm{h}(\gamma).$$

On the other hand, $g(\boldsymbol{z}^*(\beta)) = n \, \mathrm{h}(\beta)$. This implies that $\bar{\gamma} < \beta$, and so for all $\gamma < \bar{\gamma}$, the point $\boldsymbol{z}^*(\gamma) \notin \mathscr{M}'(t,\gamma)$. Thus the region $\mathscr{M}'$ in the maximization can be replaced with $\mathscr{M}$ (and $\bar{\gamma} = \gamma_0$). □

This lemma establishes that the number of errors in at least one of the candidates in the list generated after a few iterations is reduced to a desired level. After that the residual errors can be removed by another procedure as described above. In this situation we say that the errors are correctable by Algorithm II, without explicitly mentioning the second stage.

In the next theorem, which is the main result of this section, δ refers to the lower estimate of the average relative distance of the hypergraph code ensemble $\mathscr{H}$ from Theorem 4.3.8 below.

Theorem 4.3.6. *Let $t \ge 2$ be the number of errors correctable by the local code A. Algorithm II corrects any combination of up to $N(\min(\gamma_0, \delta/2))$ errors for any code $C \in \mathscr{C}_2(l, A)$ except for a proportion of codes that declines exponentially with the code length $N = nm, m \to \infty$.*

Proof. With high probability over the ensemble of hypergraphs considered, for a given hypergraph $\mathrm{H}(V, E)$ a constant number s of iterations of the algorithm will decrease the weight of error from $\gamma_0 N$ to any given positive proportion β for at least one of the l^s candidates in the list $Y_1^{(s+1)}$. Take $\beta = \sigma_0$, where σ_0 is the quantity given by Theorem 4.3.2. Next consider the bipartite graph $G(V_G = V_1 \cup V_2, E_G)$ where V_1, V_2 are the parts of H and where $(v_1, v_2) \in E_G$ if $v_1, v_2 \in e$ for some edge $e \in E$. By the previous section, with high probability these $\sigma_0 N$ errors can be corrected with $O(\log m)$ iterations of Algorithm I. Finally, the correct codevector will be selected from the list of candidates because the proportion of errors is assumed not to exceed $N\delta/2$. □

The complexity of this decoding is $O(N \log N)$ where the implicit constant depends on the code A.

In the following theorem we extend the results of this section to the case of A being a perfect single-error correcting Hamming code of length $n = 2^r - 1$ for some $r = 3, 4, \ldots$. In this case the maximum on $\boldsymbol{z}$ in the above proof can be computed in a closed form. As remarked above, in this case in the last part of the error correction procedure we use the decoding algorithm of [106] to remove residual errors from the candidate vectors.

Theorem 4.3.7. *Suppose that the local codes A are taken to be one-error-correcting Hamming codes and let $\delta = \delta(\mathscr{C}_2)$ be the relative average distance (4.1) of the ensemble $\mathscr{C}_2(l, A)$. Then almost all codes in the ensemble $\mathscr{C}_2(l, A)$ can be decoded to correct $N \min(\gamma_0, \delta/2)$ errors, where γ_0 is given by (4.10) and*

$$\tilde{F}_{n,1}(\gamma) = -\gamma n \log x + \log\biggl(1 + 2\sqrt{n \sum_{i=2}^{n} \binom{n}{i} x^{i+1}}\biggr) \tag{4.12}$$

where x is the only positive root of the equation

$$\frac{\sum_{i=2}^{n} (i+1)\binom{n}{i} x^{i+1}}{2n \sum_{i=2}^{n} \binom{n}{i} x^{i+1} + \sqrt{n \sum_{i=2}^{n} \binom{n}{i} x^{i+1}}} = \gamma.$$

Proof. It is obtained by maximizing the function $g(\boldsymbol{z})$ over the region

$$\mathscr{M}(1, \gamma) = \{\boldsymbol{z} \in \mathbb{Z}_n : \sum_{i=1}^{n} i z_i = \gamma n, z_1 = \sum_{i=2}^{n} z_i\}.$$

The Lagrangian takes the form

$$\mathrm{h}(\boldsymbol{z}) + \sum_{i=2}^{n} z_i \Bigl(\log n + \log \binom{n}{i}\Bigr) + \lambda \Bigl(\sum_{i=2}^{n} (i+1) z_i - \gamma n\Bigr),$$

where $\boldsymbol{z} = (z_1, z_2, \ldots, z_n, 1 - \sum_i z_i)$ and $z_1 = \sum_{i=2}^{n} z_i$ and λ is an arbitrary multiplier. Setting the partial derivatives to zero, we find the value λ to satisfy $2^x = \lambda$, where x is given above. The calculations are tedious but straightforward and will be omitted. □

The last theorem enables us to find the proportion of correctable errors for the case when A is the Hamming code of length $n = 2^r - 1, t = 1$. Since the examples below rely on the value of the ensemble-average distance, we rephrase and restate the theorem 2.4.5 from Chapter 2.

Theorem 4.3.8. *Let $\delta(\mathscr{C}_2)$ be the asymptotic average relative distance of codes in the l-hypergraph ensemble constructed from the local code A of length n and distance d_0. Then*

$$\delta(\mathscr{C}_2) \geq \sup_{\omega>0} \left\{ \omega : \frac{l}{n} \log \frac{1+\sum_{i=d_0}^{n} \binom{n}{i} x_0^i}{x_0^{\omega n}} < (l-1)\,\mathrm{h}(\omega) \right\}$$

where $x_0 = x_0(\omega)$ is the positive solution of the equation

$$\omega n + \sum_{i=d_0}^{n} \binom{n}{i} (\omega n - i) x^i = 0.$$

For instance, for the case $n = 31, l = 5$ this theorem gives the value of the relative distance $\delta(\mathscr{C}_2) \geq 0.01618$ (the rate of codes $R \geq 6/31$). Performing the calculation in (4.12), we find that the average code from the ensemble $\mathscr{C}_2(5, A)$ the proportion of errors correctable by codes in the ensemble using Algorithm II to be at least $\gamma_0 = 1.2 \times 10^{-5}$.

We include some more examples. In the following table $n = 2^9 - 1$.

l	17	23	28	34
Rate	0.7006	0.5949	0.5069	0.4012
γ_0	0.000235	0.000401	0.000521	0.000644
$\delta(\mathscr{C}_2)$	0.00415	0.00504	0.00558	0.00608

l	40	45	51
Rate	0.2955	0.2074	0.1018
γ_0	0.000747	0.000821	0.000898
$\delta(\mathscr{C}_2)$	0.00648	0.00676	0.00704

It is also of interest to compute the values of γ_0 for code rate $R(\mathcal{C}) \approx 0.5$.

n	127	255	511	1023
l	9	16	28	51
Rate	0.5039	0.4980	0.5068	0.5015
γ_0	0.0002012	0.0004873	0.0005207	0.0004227
$\delta(\mathscr{C}_2)$	0.01157	0.008658	0.005581	0.003394

These estimates are at least an order of magnitude better than the corresponding results in [26, 106] obtained for LDPC codes and their generalizations based on the "flipping" algorithm of [107].

The case of large n. As in the previous section, it is interesting to examine the case of long local codes A because it reveals some parallels with the analysis of the decoding algorithm in the case of nonrandom hypergraphs [14]. We begin with the observation that the proportion γ_0 of correctable errors for the ensemble $\mathscr{C}_2(l, A)$ computed above is a function of the number of errors t that each local code corrects in each iteration.

Lemma 4.3.9. *Let $t = \tau n$, $d_0 = \delta_0 n$. The ensemble $\mathscr{C}_2(l, A)$ contains codes that correct γN errors for any $\gamma < \gamma_0(\tau) \triangleq \min(\tau, x_0(\tau))$ where*

$$x_0(\tau) = \sup\{x > 0 : \Big(1 - \frac{x}{\delta_0}\Big)\,\mathrm{h}\,\Big(\frac{x\tau}{\delta_0 - x}\Big) + \frac{x}{\delta_0}\,\mathrm{h}(\delta_0 - \tau) + \varepsilon_n < (1 - 1/l)\,\mathrm{h}(x)\}$$

and $\varepsilon_n = \log n/n$.

Proof. Referring to the proof of Lemma 4.3.5, we aim at establishing conditions for the exponent L of the event $\mathscr{L}(\mathcal{E})$ to be negative as m approaches infinity. We assume that $\gamma \le \tau$ (otherwise our estimates do not imply that the convergence condition of Lemma 4.3.4 holds with high probability over the graph ensemble).

From (4.11), (4.7) we have

$$L \le -(l-1)\,\mathrm{h}(\gamma) + l \max_{\boldsymbol{z} \in \mathscr{M}(t,\gamma)} \sum_{i=0}^{n} z_i\,\mathrm{h}\,\Big(\frac{i}{n}\Big) + \frac{l \log n}{n},$$

where $\mathscr{M}(t, \gamma)$ is defined in (4.9). Next, write

$$\sum_{i=0}^{t} z_i\,\mathrm{h}\,\Big(\frac{i}{n}\Big) \le \lambda\,\mathrm{h}\,\Big(\frac{\sum_{i=1}^{t} i z_i}{\lambda n}\Big) = \lambda h\Big(\frac{\mu_1}{\lambda}\Big), \tag{4.13}$$

where we have denoted $\sum_{i=0}^{t} z_i = \lambda$, $\sum_{i=1}^{t} i z_i = \mu_1 n$. In addition let us put $\sum_{i=d_0-t}^{n} i z_i = \mu_2 n$, then the values of the sums $\sum_i z_i$ and $\sum_i i z_i$ over each of the three intervals $I_1 = [0, t]$, $I_2 = [t+1, d_0 - t - 1]$, $I_3 = [d_0 - t, n]$ can be found from the following table:

	I_1	I_2	I_3
$\sum z_i$	λ	$1 - \lambda - \mu_1/\tau$	μ_1/τ
$\sum \frac{i}{n} z_i$	μ_1	$\gamma - \mu_1 - \mu_2$	μ_2.

The variables introduced above depend on the point $\boldsymbol{z}$ and satisfy the following natural constraints: for any $\boldsymbol{z} \in \mathscr{M}(t, \gamma)$,

$$\begin{gathered} \mu_1 \le \tau\lambda \\ \tau\Big(1 - \lambda - \frac{\mu_1}{\tau}\Big) \le \gamma - \mu_1 - \mu_2 \le (\delta_0 - \tau)\Big(1 - \lambda - \frac{\mu_1}{\tau}\Big) \\ (\delta_0 - \tau)\frac{\mu_1}{\tau} \le \mu_2 \le \frac{\mu_1}{\tau}. \end{gathered} \tag{4.14}$$

Proceeding as in (4.13), we can estimate the sum on z_i in L as follows:

$$\sum_{i=0}^{n} z_i\,\mathrm{h}\,\Big(\frac{i}{n}\Big) \le f(\lambda, \mu_1, \mu_2) \tag{4.15}$$

where

$$f(\lambda,\mu_1,\mu_2)=\lambda\,\mathrm{h}\left(\frac{\mu_1}{\lambda}\right)+\left(1-\lambda-\frac{\mu_1}{\tau}\right)\mathrm{h}\left(\frac{\gamma-\mu_1-\mu_2}{1-\lambda-(\mu_1/\tau)}\right)+\frac{\mu_1}{\tau}\,\mathrm{h}\left(\frac{\mu_2\tau}{\mu_1}\right).$$

Our plan is to prove that some of the inequalities in (4.14) can be replaced by equalities, thereby expressing the variables λ,μ_1,μ_2 as functions of γ,τ. We will rely on the fact that the function f is concave in its domain. The proof of this claim follows.

First we prove that the function

$$\phi(x,y)=(1-x)\,\mathrm{h}\left(\frac{\gamma-y}{1-x}\right)$$

is concave (not necessarily in the strict sense) for $0<x,y<1, 0<\gamma-y<1-x$. For that, let us compute its Hessian matrix:

$$H=\frac{1}{\ln 2}\begin{pmatrix}\frac{\gamma-y}{(1-x)(\gamma-y+x-1)} & -\frac{1}{\gamma-y+x-1}\\ -\frac{1}{\gamma-y+x-1} & \frac{1-x}{(\gamma-y)(\gamma-y+x-1)}\end{pmatrix}$$

The eigenvalues of H are

$$0,\quad \frac{(\gamma-y)^2+(1-x)^2}{(1-x)(\gamma-y)(\gamma-y-(1-x))}<0,$$

so $H\preceq 0$, and so ϕ is concave. Next observe that the function

$$\left(1-\lambda-\frac{\mu_1}{\tau}\right)\mathrm{h}\left(\frac{\gamma-\mu_1-\mu_2}{1-\lambda-(\mu_1/\tau)}\right)$$

can be obtained from ϕ by a linear change of variables

$$x=\lambda+\mu_1/\tau,\quad y=\mu_1+\mu_2$$

and therefore is also concave. Finally, the functions $\lambda\,\mathrm{h}(\mu_1/\lambda)$ and $(\mu_1/\tau)\,\mathrm{h}(\mu_2\tau/\mu_1)$ are also concave, and thus so is the function $f(\lambda,\mu_1,\mu_2)$.

Note that for all $\boldsymbol{z}\in\mathcal{Z}_n$ the sum

$$\sum_{i=0}^{n} z_i\,\mathrm{h}\left(\frac{i}{n}\right)\le \mathrm{h}(\gamma)$$

and that it equals $\mathrm{h}(\gamma)$ at the point $\tilde{\boldsymbol{z}}$ such that $z_i=1$ for $i=\lceil\gamma n\rceil$ and $z_i=0$ elsewhere. Also note that since $\gamma<\tau$, the point $\tilde{\boldsymbol{z}}$ is outside the region $\mathscr{M}(t,\gamma)$ and thus, by concavity,

$$a:=\max_{\boldsymbol{z}\in\mathscr{M}(t,\gamma)}\sum_{i=0}^{n} z_i\,\mathrm{h}\left(\frac{i}{n}\right)<\mathrm{h}(\gamma).$$

Let $\boldsymbol{z}_1$ be the point at which this maximum is attained, and let $\boldsymbol{x}_1 = (\lambda, \mu_1, \mu_2)$ be the corresponding point for the arguments of f. By construction, the point $\boldsymbol{x}_1$ satisfies the inequalities of (4.14). At the same time, consider the function $f(\cdot)$ on the line $\lambda = \mu_1 = \mu_2$. As the variables approach 0 along this line, the value $f(\lambda, \mu_1, \mu_2)$ approaches $\mathrm{h}(\gamma)$.

To summarize, we have found two points, $\boldsymbol{x}_1$ and $\boldsymbol{x}_2 = (0, 0, 0)$ that are located on different sides of the hyperplane

$$\tau\Big(1 - \lambda - \frac{\mu_1}{\tau}\Big) = \gamma - \mu_1 - \mu_2$$

such that $f(\boldsymbol{x}_1) \geq a, f(\boldsymbol{x}_2) > a$. Invoking concavity of the function f, we now conclude that there is a feasible point $\boldsymbol{x}'$ on this hyperplane such that $f(\boldsymbol{x}') \geq a$.

Therefore, put $\mu_2 = \gamma - \tau(1 - \lambda)$ and write

$$f_1(\lambda, \mu_1) = \lambda\,\mathrm{h}\left(\frac{\mu_1}{\lambda}\right) + \Big(1 - \lambda - \frac{\mu_1}{\tau}\Big)\,\mathrm{h}(\tau) + \frac{\mu_1}{\tau}\,\mathrm{h}\left(\frac{\tau(\gamma - \tau(1-\lambda))}{\mu_1}\right)$$

where the variables are constrained as follows: for any $\boldsymbol{z} \in \mathscr{M}(t, \gamma)$,

$$\mu_1 \leq \tau\lambda$$

$$\tau(1 - \lambda) - \mu_1 \geq 0 \tag{4.16}$$

$$(\delta_0 - \tau)\frac{\mu_1}{\tau} \leq \gamma - \tau(1 - \lambda) \leq \frac{\mu_1}{\tau}. \tag{4.17}$$

Since f_1 is a restriction of f to a hyperplane, it is still concave. Now notice that $f_1(1, \tau) = \mathrm{h}(\gamma)$ and that the point $(1, \tau)$ does not satisfy inequality (4.16) and the left of the inequalities (4.17). Repeating the above argument, we claim that the function f in (4.15) can be further restricted to the intersection of the planes $\tau(1 - \lambda) = \mu_1$ and $(\delta_0 - \tau)(\mu_1/\tau) = \gamma - \tau(1 - \lambda)$. Altogether this gives:

$$\lambda = 1 - \gamma/\delta_0, \quad \mu_1 = \gamma\tau/\delta_0.$$

Let us substitute these values into the expression for f_1 and rewrite (4.15) as follows: for any $0 \leq \gamma < \tau$,

$$\max_{\boldsymbol{z} \in \mathscr{M}(t,\gamma)} \sum_{i=0}^{n} z_i\,\mathrm{h}\left(\frac{i}{n}\right) \leq \Big(1 - \frac{\gamma}{\delta_0}\Big)\,\mathrm{h}\left(\frac{\gamma\tau}{\delta_0 - \gamma}\right) + \frac{\gamma}{\delta_0}\,\mathrm{h}(\delta_0 - \tau). \tag{4.18}$$

Thus if the condition in the statement is fulfilled then $L < 0$. This concludes the proof. □

The main part of the proof is estimating the solution of the following linear program

$$\max_{\boldsymbol{z}} \sum_{i=1}^{n} z_i\,\mathrm{h}\left(\frac{i}{n}\right)$$

$$\boldsymbol{z} = (z_0, z_1, \ldots, z_n) \in \mathscr{M}(t, \gamma)$$

where the variables define a probability distribution on $\{0, 1, \ldots, n\}$. It is clear from concavity that the maximum is attained at the point where among all the indices $i \in I_1$ at most one value z_i is nonzero, and the same applies to I_2 and I_3. We have shown that the value of the program is bounded above by the right-hand side of (4.18). The following point gives this value and is therefore a maximizing point:

$$z_{i_1} = 1 - \frac{\gamma}{\delta_0},\ z_{i_2} = \frac{\gamma}{\delta_0},\ z_i = 0 \text{ otherwise},$$

where $i_1 = n\gamma\tau/(\delta_0 - \gamma), i_2 = n(\delta_0 - \tau)$. Since

$$\frac{\gamma\tau}{\delta_0 - \gamma} \leq \tau,$$

this shows that the worst-case allocation of errors to vertices in a given part of the graph assigns no edges to vertices that are neither good nor bad. This also confirms the intuition suggested by Lemma 4.3.4 that bad vertices (vertices assumed to add errors) should each be assigned the smallest possible number of error edges $d_0 - t$.

The next proposition is now immediate.

Proposition 4.3.10. *The ensemble $\mathscr{C}_2(l, A)$ with long local codes contains codes that can be decoded using Algorithm II to correct all error patterns whose weight is less than $\gamma_0 N$, where*

$$\gamma_0 = \max_{0<\tau\leq\delta_0/2} \gamma_0(\tau). \tag{4.19}$$

Estimating the number of correctable errors for the ensemble $\mathscr{C}_2(l, A)$ from Proposition 4.3.10 analytically is difficult because it involves optimization on τ (generally, the local codes should be used to correct a smaller than $\delta_0/2$ proportion of errors). We note that in the particular case of $\tau = \delta_0/2$ the proof of Lemma 4.3.9 can be considerably simplified, although the resulting value of γ is not always optimal.

Example: Let $l = 3$. Using local codes with $\delta_0 = 0.05$ we can construct hypergraph codes of rate $R \geq 0.19$. From Corollary 2.4.6, the ensemble-average relative distance is at least $\delta \approx 0.0112$ and the proportion of errors correctable by Algorithm II is found from (4.19) to be $\gamma_0 \approx 0.0035$.

Example: Let $\delta_0 = 0.01$ and $l = 10$. In this case, we find from 2.4.6 the value of the relative distance $\delta \approx 0.00599$. The code rate satisfies $R \geq 0.14$. Performing the computations in (4.19) and Lemma 4.3.9 we find the estimate of the proportion of correctable errors to be $\gamma_0 \approx 0.002198$.

In conclusion we have estimated the proportion of errors correctable by codes from ensembles defined by random l-partite graphs, $l \geq 2$. In contrast to the case of expander codes [93], [104], [17], [19], [14] our calculations cover the case of local codes of arbitrary given length and distance, including small values of the distance. In this part of the dissertation we provided answers to a set of basic questions regarding networks of short linear binary codes. This extends our perspective of concatenated code constructions to the case of sparse regular graphs.

Part III

Matrices

CHAPTER 5

Compressed Sensing and the RIP

5.1 Introduction

In the next two chapters we study applications of error-correcting codes in the problem of *compressed sensing*. Compressed sensing is a technique of recovering sparse N-dimensional signals from low-dimensional sketches, i.e., their linear images in $\mathbb{R}^m, m \ll N$. In formal terms the problem can be stated as follows. Let $\Phi : \mathbb{R}^m \to \mathbb{R}^N$ be a linear operator used to create a "sketch" of a signal represented by a real vector $\boldsymbol{x} \in \mathbb{R}^N$. In other words, we observe a vector $\boldsymbol{r} = \Phi\boldsymbol{x}$, where Φ is an $m \times N$ sampling matrix. Recovering $\boldsymbol{x}$ from $\boldsymbol{r}$ is generally impossible because the system of equations is under-determined, and the solutions form an affine subspace in $\mathbb{R}^N$. The problem becomes tractable if we know that $\boldsymbol{x}$ is sparse, i.e., have only $k \ll N$ nonzero entries. In particular we seek an approximation of $\boldsymbol{x}$ by a vector $\hat{\boldsymbol{x}}$, such that

$$\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_{p_1} \leq C \min_{\boldsymbol{x}' \text{ is } k\text{-sparse}} \|\boldsymbol{x} - \boldsymbol{x}'\|_{p_2} \tag{5.1}$$

for some $p_1, p_2 \geq 1$ and some constant C. A k-sparse vector is a vector with k or fewer nonzero coordinates, where $k \ll N$. Note that if $\boldsymbol{x}$ is itself k-sparse, then (5.1) implies that $\hat{\boldsymbol{x}} = \boldsymbol{x}$. Moreover, the recovery is stable: if $\boldsymbol{x}$ is approximately sparse (has only k "significant" entries), then the approximation error is bounded by (5.1).

In this formulation, the study of the compressed sensing problem has been focused on the design of good sampling matrices Φ in conjunction with low-complexity recovery algorithms that provide an error guarantee of the form (5.1) based on as few samples as possible. As one of the first examples, it was shown that random Gaussian matrices provide a $(p_1 = 2, p_2 = 1)$ error guarantee with $m = O(k \log(N/k))$ sketch length and a polynomial-time (linear programming) recovery algorithm [32].

It is known that at least $m = \Omega(k \log(N/k))$ samples are required for any recovery algorithm with an error guarantee of the form (5.1) (see, for example, [69], [7]); and the best known guarantee is given by random matrices with independent

Gaussian entries [32, 33]. It has also been shown that the same error guarantee is provided by independent Bernoulli random variables taking values in the set $\{\pm\frac{1}{\sqrt{m}}\}$ with the same number of samples $m = \Omega(k\log(N/k))$ [33]. However, constructing such a matrix requires $mN = \Omega(kN\log(N/k))$ random bits, so this approach is very far from being explicit (deterministic) even for small values of k.

Arguably one of the most efficient ways of constructing deterministic sampling matrices relies on their links with error-correcting codes. We pursue this link, providing constructions of matrices for sketch length $m = O(k^2\log N)$ where no previous deterministic constructions were known (in particular, the constructions of [46] and subsequent works require $\Theta(k^2\log^2 N)$ samples).

One notational difference this part has from the other parts of the dissertation is that the length of the binary code concerned is denoted here by m instead of n, and the cardinality of the code is denoted by N (rather than the notation M used elsewhere). We adopt this change in order not to deviate too much from the existing literature on the compressed sensing problem.

A useful tool for the construction of sampling matrices is provided in the works of Candés et al. [32, 33] who showed that recovery is possible with a $(p_1 = 2, p_2 = 1)$ error guarantee if the matrix Φ has the *restricted isometry property (RIP)*. In particular most matrices in the ensemble of the random Gaussian and Bernoulli matrices satisfy the RIP. Thus the construction problem of sampling operators reduces to the problem of construction of RIP matrices. Checking whether a given matrix has the RIP property is computationally infeasible unless the matrix has some structural properties. In this chapter we give explicit constructions of matrices with RIP based on error-correcting codes.

The RIP is known to hold if the columns of Φ are near-orthonormal, i.e., $\|\phi_i\|_2^2 = 1$, $|\phi_i^T\phi_j| \leq \mu$ for all $i \neq j$ and some $\mu < 1$. Such collections of vectors are also known as *incoherent dictionaries* (e.g., [100]). As such dictionaries are chosen from the unit sphere in $\mathbb{R}^m$, the problem becomes equivalent to the construction of good *spherical codes* [41]. If we further restrict the vectors in the dictionary to have binary coordinates $\frac{\pm 1}{\sqrt{m}}$, then the problem reduces to construction of binary codes in which the Hamming distance between every pair of vectors is close to $m/2$ (the code has a *narrow distance distribution*). This fact was used implicitly in [46, 60] and later more explicitly in [11, 24, 28, 88] and other works. Thus, this thesis is not the first work to pursue the link between sampling operators and codes. The performance limits of these construction, set by bounds on spherical and binary codes, precludes them from approaching the optimal sketch length. The new results obtained in this chapter pertain to improvements over the existing work: they lead to a deterministic construction of sampling matrices with RIP for $m = O(k^2\log N)$ which is by a factor of $\log N$ smaller than what was known before, and is a factor k away from the optimal (shortest possible) sketch length.

The results of this chapter appear in [11, 13].

5.1.1 The restricted isometry property of sampling matrices

Let $I \subseteq [N] := \{1, \dots, N\}$. Denote by $\Phi_I \in \mathbb{R}^{m \times |I|}$ the matrix formed of the columns of Φ with indices in I.

Definition 5.1.1 (The Restricted Isometry Property (RIP)). *A matrix $\Phi \in \mathbb{R}^{m \times N}$ is said to satisfy the (k, δ)* restricted isometry property, *or (k, δ)-RIP, $k \leq m$, $0 \leq \delta \leq 1$, if for all $I \subset [N]$ such that $|I| = k$ and for all $\boldsymbol{u} \in \mathbb{R}^k$,*

$$(1 - \delta)\|\boldsymbol{u}\|_2^2 \leq \|\Phi_I \boldsymbol{u}\|_2^2 \leq (1 + \delta)\|\boldsymbol{u}\|_2^2. \tag{5.2}$$

It is known [34] that a $(2k, \sqrt{2} - 1)$-RIP matrix enables one to approximate any k-sparse signal with $(p_1 = 2, p_2 = 1)$ error guarantee (5.1). Namely, the basis pursuit algorithm of Candés et al. [33] solves the following linear program:

$$\|\boldsymbol{x}\|_1 \to \min$$
$$\text{subject to } \Phi \boldsymbol{x} = \boldsymbol{r},\ \Phi \in \mathbb{R}^{m \times N}.$$

The above optimization problem can be solved with time complexity at most $O(N^3)$. In [34], it is shown that if Φ is $(2k, \sqrt{2} - 1)$-RIP then the solution $\hat{\boldsymbol{x}}$ of the above linear program satisfies, for some constant C,

$$\|\boldsymbol{x} - \hat{\boldsymbol{x}}\|_2 \leq \frac{C}{\sqrt{k}} \min_{\boldsymbol{x}' \text{ is } k\text{-sparse}} \|\boldsymbol{x} - \boldsymbol{x}'\|_1.$$

In the rest of this chapter our aim will be to construct $m \times N$ (k, δ)-RIP matrices with minimum possible number of samples (m).

5.2 RIP and codes

5.2.1 Sampling matrices as incoherent dictionaries

While random matrices with high probability have RIP, constructing structured sampling matrices is related to introducing certain restrictions on their entries. One such constructive approach assumes that every column $\phi_i \in \mathbb{R}^m$ of Φ is a unit-length real vector (i.e., $\|\phi_i\|_2^2 = 1$). A system of such vectors is characterized by their coherence parameter

$$\mu = \max_{i \neq j} |\phi_i^T \phi_j|.$$

For every $I \subset [N]$ with $|I| = k$ we have

$$\Phi_I^T \Phi_I = \mathrm{Id}_k + F,$$

where Id_l is the $l \times l$ identity matrix and the absolute value of every entry of F is at most μ.

Theorem 5.2.1. (The Gershgorin circle theorem [64, p.344], [72, p.240]) *Let A be an $n \times n$ complex matrix and a_{ij} is the (i,j)th entry of A. Define the set of n disks in $\mathbb{C}$ given by $|z - a_{ii}| = \sum_{j \neq i} |a_{ij}|$, $1 \leq i \leq n$. Then every eigenvalue of A is contained in one of these disks.*

We use this theorem for the matrix $\Phi_I^T \Phi_I$ which is real symmetric with 1 in the diagonal entries and the off-diagonal terms whose magnitude is bounded above by μ. Therefore its eigenvalues, which are real, satisfy $|\lambda - 1| \leq (k-1)\mu$. In other words, the matrix Φ is $(k, k\mu)$-RIP.

Sets of unit vectors with small coherence (incoherent dictionaries) are called *spherical codes*. An (m, N, μ) spherical code $\mathcal{C} \subset \mathbb{R}^m$ of size $|\mathcal{C}| = N$ is a set of unit-norm vectors such that the points of $\mathcal{C}$ are well-separated, i.e., $\boldsymbol{x}_1^T \boldsymbol{x}_2 \leq \mu$ for any two distinct $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{C}$. Thus, bounds for spherical codes [41] can be used to quantify the tradeoff between size, dimension, and coherence of the dictionary. In particular the Shannon bound implies that for large m there exist dictionaries with coherence μ such that

$$m \leq \frac{2 \ln N}{\mu^2}(1 + o(1)).$$

At the same time, by the Kabatiansky-Levenshtein bound, for all dictionaries

$$m \geq \frac{4 \ln N}{\mu^2 \ln(4e/\mu^2)}(1 - o(1)).$$

Further lower bounds on the sketch length for dictionaries from spherical codes are given by [75].

5.2.2 RIP property of matrices from binary codes

Further restricting the alphabet of dictionaries, we construct sampling matrices from binary codes. Let $\mathcal{C}$ be a binary code of length m and size N (briefly, an (m, N) code), i.e., a set of N vectors in $\{0,1\}^m$. Given a codeword $\boldsymbol{x} \in \mathcal{C}$, let us map it to a unit vector $\phi \in \mathbb{R}^m$ by setting $0 \to +\frac{1}{\sqrt{m}}$ and $1 \to -\frac{1}{\sqrt{m}}$. In this way, a binary code $\mathcal{C}$ gives rise to a matrix $\Phi = (\phi_1, \ldots, \phi_N)$. The inner product of any two column of Φ equals

$$\phi_i^T \phi_j = 1 - \frac{2\mathrm{d}(\boldsymbol{x}_i, \boldsymbol{x}_j)}{m}$$

where $\boldsymbol{x}_i, \boldsymbol{x}_j$ are the codewords of $\mathcal{C}$ that correspond to ϕ_i, ϕ_j. Assume that for every $\boldsymbol{x}_i, \boldsymbol{x}_j \in \mathcal{C}, i \neq j$, the Hamming distance between them satisfies $|\mathrm{d}(\boldsymbol{x}_i, \boldsymbol{x}_j) - \frac{m}{2}| \leq w_m$. If the code $\mathcal{C}$ satisfies this assumption for some w_m, we call the number w_m the *width* of $\mathcal{C}$ (to be precise, this is the "width" of the distance distribution of the code). From the above discussion we conclude:

Proposition 5.2.2. *A binary code $\mathcal{C}$ with width w_m gives rise to a (k, δ)-RIP matrix with $\delta = \frac{2kw_m}{m}$.*

Therefore, a sufficient condition of (k, δ)-RIP is given by $w_m \le \frac{m\delta}{2k}$. Thus, we would like to design a code $\mathcal{C} \in \{0,1\}^m$ of a given size N such that the following condition is satisfied.

Property 1: For a given k and any distinct $\boldsymbol{c}_1, \boldsymbol{c}_2 \in \mathcal{C}$,

$$\left| \mathrm{d}(\boldsymbol{c}_1, \boldsymbol{c}_2) - \frac{m}{2} \right| \le \frac{m\delta}{2k}.$$

For a linear binary code $\mathcal{C}$ the distribution of Hamming distances is identical to the distribution of Hamming weights (see Sec. 1.2). Therefore, we can restate Property 1 as follows:

Property 2: All nonzero vectors $\boldsymbol{c} \in \mathcal{C}$ satisfy

$$\left| \mathrm{wt}(\boldsymbol{c}) - \frac{m}{2} \right| \le \frac{m\delta}{2k}. \tag{5.3}$$

Linear codes with bounded width have been considered in coding theory. One prominent example is given by codes dual to primitive BCH codes of length m that correct $t < \sqrt{m}/2$ errors. Their width is related to bounds on exponential sums and is given by the Carlitz-Uchiyama bound [80, p.280]. This and related constructions of incoherent dictionaries were considered in [1, 6, 65, 103]. General constructions of codes of small width were considered in Alon et al. [2], with the current best construction by Ben-Aroya and Ta-Shma [18]. Independently, DeVore [46] followed the same line of thought, considering binary images of Reed-Solomon codes over $\mathbb{F}_q$ under the trivial map that sends the symbol $a \in \mathbb{F}_q$ to its q-dimensional indicator vector. The number of samples required in his construction is $m = \Theta(k^2 \log^2 N)$.

5.2.3 Linear codes with random generator matrices

Next we present a randomized construction that uses only $m \log N$ random bits as input and provides a matrix satisfying (k, δ)-RIP with high probability. We use a version of the GV bound in an argument similar to Prop. 1.5.1, switching from random parity-check matrices to random generator matrices.

Theorem 5.2.3. *Let $l = \log N$ and let $G = (\boldsymbol{g}_1, \ldots, \boldsymbol{g}_m)$ be an $l \times m$ binary matrix whose columns are chosen independently with $P[\boldsymbol{g}_i = \boldsymbol{y}] = 2^{-l}$ for all $\boldsymbol{y} \in \{0,1\}^l$. Let $m = \lceil 4(k/\delta)^2 \ln N \rceil$. Then the linear code $\mathcal{C}$ spanned by the rows of G satisfies Property 2 with probability approaching 1 as $N \to \infty$.*

Proof. Let X be the random number of codewords of weight w in the code $\mathcal{C}$ such that $|w - \frac{m}{2}| \ge \frac{m\delta}{2k}$. Let $\boldsymbol{u} \in \{0,1\}^l$ be a nonzero row vector. We have $P[\boldsymbol{u}\boldsymbol{g}_i = 0 \pmod 2] = P[\boldsymbol{u}\boldsymbol{g}_i = 1 \pmod 2] = 1/2$. Then the probability that the codeword $\boldsymbol{u}G$ has weight w equals $\binom{m}{w} 2^{-m}$. Hence the expected number of vectors of weight $w > 0$ in $\mathcal{C}$ equals

$$\mathbb{E}A_w = \frac{(N-1)}{2^m} \binom{m}{w}. \tag{5.4}$$

Therefore, using (1.6)

$$\mathbb{E}X \leq 2\frac{N-1}{2^m} \sum_{w=0}^{m(\frac{1}{2}-\frac{\delta}{2k})} \binom{m}{w} \leq N2^{1-m(1-\mathrm{h}(\frac{1}{2}-\frac{\delta}{2k}))}. \tag{5.5}$$

Next we use the Taylor series expansion for $\mathrm{h}(x)$ around $1/2$:

$$1-\mathrm{h}(1/2-x) = (2/\ln 2)(x^2+2/3x^4+O(x^6)) \qquad (0 \leq x < 1/2). \tag{5.6}$$

Finally, since $m \geq 4(k/\delta)^2 \ln N$ we obtain $\mathbb{E}X \leq 2/N$. However, $\Pr(X>0) = \sum_{i\geq 1}\Pr(X=i) \leq \mathbb{E}X$, and thus the proportion of codes for which Property 2 holds approaches one as N increases. □

5.2.4 Explicit RIP matrices

Here we derandomize the theorem and the proof from the previous section. This results in an explicit construction of k-RIP matrices with complexity $O(mN)$.

The derandomization procedure that we employ is very similar to explicit constructions of linear codes achieving GV bound. Linear $[m, \log N]$ codes reaching the GV bound can be shown to exist relying both on random parity-check matrices and random generator matrices (see, Sec. 1.5). Derandomizing the parity-check ensemble involves complexity $O(m^3 2^{m-\log N})$ and is easily accomplished using a greedy procedure. However in our setting $m = \Theta(k^2 \log N)$, i.e, the resulting complexity is $O(N^{k^2})$. We will show that the construction relying on random generator matrix of the previous section can also be derandomized, resulting in RIP matrices constructible with a lower complexity of $O(mN)$.

Derandomizing Gilbert-Varshamov codes from generator matrices was recently addressed by Porat and Rotschild [90]. We follow their main idea with some slight technical changes. In particular, we tailor it to our goal of constructing codes of small width as opposed to codes with large distance. We note that codes of small width constructed below also meet the Gilbert-Varshamov bound on the minimum distance.

The method of conditional expectations, used below and also in [90], is due to Spenser and Raghavan (see [4]). We will recursively select columns in the $l \times m$ generator matrix G, $l = \log N$. Before any columns are selected, the expected number of vectors of weight w in the code $\mathcal{C}$ (the row space of G) is as given in (5.4) and the expected number of vectors of weight far from $m/2$ (outliers) is given by (5.5). The algorithm selects columns one by one so that the expectation of the number of outlying vectors *conditioned* on the columns already chosen is the smallest possible.

Theorem 5.2.4. *Let $m = \lceil 4(k/\delta)^2 \ln N \rceil$. It is possible to deterministically construct in time $O(k^2 N \log N)$ a linear code of width $\frac{m\delta}{2k}$ (see (5.3)). Therefore, a deterministic $m \times N$ sampling matrix Φ with (k,δ)-RIP can be constructed in time $O(k^2 N \log N)$.*

Proof. In this proof we denote by $\boldsymbol{g}_i, i = 1, \ldots, m$ random variables taking values in the set of vectors $\{0,1\}^l$ (random columns of G) and denote by g_i realizations of these random variables. As before, let X be the random number of vectors in the code with generator matrix $G = (\boldsymbol{g}_1, \ldots, \boldsymbol{g}_m)$ whose weight w satisfies $|w - \frac{m}{2}| \geq \frac{m\delta}{2k}$. The expectation $\mathbb{E}X = \mathbb{E}_{\boldsymbol{g}_1,\ldots,\boldsymbol{g}_m} X$ is given by (5.5). Define a sequence of random variables $X_0 = X, X_1, \ldots, X_m$. Here for $i = 1, \ldots m$ the variable $X_i = X_i(g_1, \ldots, g_i)$ is the random number of outlying vectors conditioned on the specific choice of the first i columns

$$X_i = \left| \left\{ \boldsymbol{x} \in \mathcal{C} : |\operatorname{wt}(\boldsymbol{x}) - \frac{m}{2}| \geq \frac{m\delta}{2k} \text{ given that } \boldsymbol{g}_j = g_j, 1 \leq j \leq i \right\} \right|.$$

The dependence of X_i on the vectors $g_1, \ldots, g_i$ is understood and will be suppressed below. The quantity X_m is a (nonrandom) number of outlying vectors in the row space of G, and our purpose is to construct a code with $X_m \leq \mathbb{E}X < 1$.

Choose g_1 arbitrary nonzero. Suppose that $\boldsymbol{g}_j = g_j, j = 1 \ldots, i$ have been chosen. For a given $\boldsymbol{u} \in \{0,1\}^m$ consider the probability

$$\Pr(\operatorname{wt}(\boldsymbol{u}G) = w \mid g_1, \ldots, g_i) = \Pr(\operatorname{wt}(\boldsymbol{u}G) = w \mid \boldsymbol{g}_1 = g_1, \ldots, \boldsymbol{g}_i = g_i).$$

We have

$$\begin{aligned} &\Pr(\operatorname{wt}(\boldsymbol{u}G) = w \mid g_1, \ldots, g_i) \\ &\qquad = \sum_{g_{i+1} \in \{0,1\}^l} 2^{-l} \Pr(\operatorname{wt}(\boldsymbol{u}G) = w \mid g_1, \ldots, g_i; \boldsymbol{g}_{i+1} = g_{i+1}). \quad (5.7) \end{aligned}$$

Denoting the number of vectors of weight w in $\mathcal{C}$ by A_w, we have

$$\mathbb{E}(A_w \mid g_1, \ldots, g_i) = \sum_{\boldsymbol{u} \neq 0} \Pr(\operatorname{wt}(\boldsymbol{u}G) = w \mid g_1, \ldots, g_i)$$

for all $0 < i, w \leq m$ (if the values of i, w are inconsistent with each other, then the summation terms are 0). Finally, by (5.7)

$$\begin{aligned} \mathbb{E}X_i &= \sum_{w : |w - m/2| \geq \frac{m\delta}{2k}} \mathbb{E}(A_w \mid g_1, \ldots, g_i) \\ &= \sum_{w : |w - m/2| \geq \frac{m\delta}{2k}} \sum_{\boldsymbol{u} \neq 0} \Pr(\operatorname{wt}(\boldsymbol{u}G) = w \mid g_1, \ldots, g_i) \\ &= \sum_{\substack{w: \\ |w - m/2| \geq \frac{m\delta}{2k}}} \sum_{\boldsymbol{u} \neq 0} \sum_{g_{i+1} \in \{0,1\}^l} 2^{-l} \Pr(\operatorname{wt}(\boldsymbol{u}G) = w \mid g_1, \ldots, g_i, g_{i+1}) \\ &\geq \min_{g_{i+1} \in \{0,1\}^l} \sum_{\substack{w: \\ |w - m/2| \geq \frac{m\delta}{2k}}} \sum_{\boldsymbol{u} \neq 0} \Pr(\operatorname{wt}(\boldsymbol{u}G) = w \mid g_1, \ldots, g_i, g_{i+1}). \end{aligned}$$

This shows that for every $i = 1, \dots, m$. there is a choice of the $(i+1)$st column such that $\mathbb{E}X_i \geq \mathbb{E}X_{i+1}$. Since $\mathbb{E}X_0 = \mathbb{E}X$, also $\mathbb{E}X_m = X_m \leq \mathbb{E}X$. From (5.5)-(5.6),

$$X_m \leq N2^{1-m(1-\mathrm{h}(\frac{1}{2}-\frac{\delta}{2k}))} \leq N2^{1-2m\frac{\delta^2}{4k^2}}.$$

Finally, substituting the value of m from the statement, we observe that $X_m < 1$ for all $N > 2$, i.e., the width of $\mathcal{C}$ is $m\delta/2k$.

To estimate the complexity of the procedure described we need to specify a way to compute the probabilities $\Pr(\mathrm{wt}(\boldsymbol{u}G) = w \mid g_1, \dots, g_i), i = 1, 2, \dots,$ which can be used to compute $\mathbb{E}X_i$.

Let $G_i := (g_1, \dots, g_i)$ and let $\mathcal{C}_i$ be the row space of G_i. For a given choice of the first i columns $g_1, \dots, g_i$ and a given $\boldsymbol{u} \in \{0,1\}^l$ we have

$$\Pr(\mathrm{wt}(\boldsymbol{u}G) = w \mid g_1, \dots, g_i) = \binom{m-i}{w - \mathrm{wt}(\boldsymbol{u}G_i)} 2^{-(m-i)}$$

if $w \geq \mathrm{wt}(\boldsymbol{u}G_i)$ and 0 otherwise. Therefore,

$$\mathbb{E}(A_w \mid g_1, \dots, g_i) = \sum_{\boldsymbol{u} \neq 0} \binom{m-i}{w - \mathrm{wt}(\boldsymbol{u}G_i)} 2^{-(m-i)},$$

and

$$\mathbb{E}X_i = \sum_{\boldsymbol{u} \neq 0} f(i, \mathrm{wt}(\boldsymbol{u}G_i)),$$

where

$$f(i,s) = \sum_{\substack{w: \\ |w - m/2| \geq m\delta/2k}} \binom{m-i}{w-s} 2^{-(m-i)}.$$

The complexity of the algorithm is determined by the cost of finding the value of g_{i+1}. For that we must, for every possible g_{i+1}, compute $\mathbb{E}X_{i+1}$ and find the smallest of these quantities. Computing $\mathbb{E}X_{i+1}$ takes finding the value $f(i+1, \mathrm{wt}(\boldsymbol{u}G_{i+1}))$ for every choice of $\boldsymbol{u}$. There are at most $2m$ possible values of f, but finding the weights $\mathrm{wt}(\boldsymbol{u}G_{i+1})$ has to be done for each $\boldsymbol{u}$ and each g_{i+1}, resulting in N^2 evaluations.

To reduce the complexity to $O(mN)$, we follow the idea of [90]. Namely, it is possible to choose the entries of the column g_{i+1} one by one, optimizing the choice in every step. This results in a more cumbersome expression for the expectation, but gives a simpler algorithm. Since $m = \Theta(k^2 \log N)$, we obtain the complexity expression claimed in the theorem. □

Thus we have constructed explicit (k, δ)-RIP matrices of dimensions $m \times N$ where $m = \Theta(k^2 \log N)$ which enables recovery of k-sparse signals from m-dimensional sketches. In the next chapter we examine the question of what happens if we relax the condition of RIP to permit a small proportion of sparse signals to be lost. It will turn out that much shorter sketches should suffice.

5.3 Further remarks on RIP matrices from code ensembles

In this section we make brief comments relating to Theorem 5.2.3. Once it is realized that codes with small width give good sampling matrices, this theorem follows by a standard argument about the existence linear codes achieving the GV bound (as in 1.5.1). In applying it, we are seeking codes whose relative distance differs from $1/2$ by a small amount, namely, by $\delta/2k$. Since k is a growing quantity, the result follows by looking at the asymptotic behavior of the rate of codes achieving the GV bound in the neighborhood of $R = 0$. We note that, apart from the ensemble of linear codes defined by uniform random generator matrices it is possible to consider other code ensembles that contain codes achieving the GV bound or even codes that do not attain it but are nevertheless asymptotically good. The purpose of this consideration is a partial derandomization of the construction of sampling matrices.

This line of thought was examined in [11] where we looked at various families of concatenated codes and codes on graphs and hypergraphs (see Ch. 2) with the purpose of locating code ensembles that give rise to (k, δ)-RIP matrices relying on a small number of *random bits*. The number of random bits employed in Theorem 5.2.3 is clearly km which is $O(k^2 \log^2 N)$ since $k = \log N$ and $m = O(k^2 \log N)$. The smallest number of random bits among the ensembles considered in [11] is required for sampling matrices arising from hypergraph codes. It it slightly less than the above quantity, and for the same range of parameters equals $O(k^2 \log N \log \log N)$.

CHAPTER 6

The Statistical Isometry Properties

6.1 Introduction

There is another aspect of compressed sensing where randomization is built into the signal and recovery model [28, 31, 57, 100]. As in many applied problems (for instance, transmission over noisy channels), performance can be enhanced by permitting an almost always recovery of k-sparse signals with some guarantee of the form of (5.1). The idea behind this approach is a standard one in probabilistic combinatorics, namely, we relax the requirement (5.2) that the sampling operator Φ be a near-isometry from all to almost all sparse vectors. Analyzing the recovery properties under this relaxation is not immediate; however, a number of useful ideas in this direction have been suggested in earlier works [28, 57, 100]. The two properties desired from a sampling matrix that had been put forward by these works are the Statistical RIP (SRIP) and Statistical Unique Recovery Property (SURP).

We show that it is possible to construct sampling matrices from codes that satisfy both SRIP and SURP, i.e. act as near-isometry on most k-sparse signals. Let us define a version of the SRIP used in our analysis below. The definition that we give is slightly stronger than the one in [28] and is close to the definition in [57].

Definition 6.1.1 (SRIP)**.** *An $m \times N$ sampling matrix Φ is said to satisfy the (k, δ, ε)-SRIP $k \leq m$, $0 \leq \delta \leq 1$, $0 \leq \varepsilon < 1$ (is (k, δ, ε)-SRIP), if (5.2) holds for at least $1 - \varepsilon$ proportion of all subsets $I \subset [N]$ such that $|I| = k$.*

The statistical unique recovery property (SURP) is another useful property for sparse signal recovery. The following definition is similar to the notion appearing in [100]. Consider the product measure $P_k \times P_{\boldsymbol{z}}$ where P_k is the uniform distribution on the k-subsets of $[N]$ and where $P_{\boldsymbol{z}}$ is any absolutely continuous probability distribution on the set of k-dimensional real vectors $\boldsymbol{z}$, with respect to the Lebesgue measure on $\mathbb{R}^k$. In the following definition the probability is computed according to this measure.

Definition 6.1.2 (SURP)**.** *Let $k \leq m$, $0 \leq \varepsilon < 1$. An $m \times N$ sampling matrix Φ is*

said to satisfy the (k, ε)*-SURP if*

$$\Pr(\{k\text{-sparse } \boldsymbol{y} \in \mathbb{R}^n, \boldsymbol{y} \neq \boldsymbol{x} : \Phi\boldsymbol{y} = \Phi\boldsymbol{x}\}) < \varepsilon. \tag{6.1}$$

The definition of SURP is close to [28, Def. 2], but not equivalent to it.

The SRIP and SURP are used by Calderbank et al. [28, 29] to show recovery guarantee for k-sparse signals under their reconstruction algorithm. These properties are also used in [100] to show that exact recovery of signals under some random models are possible. The good performance of some sampling matrices are also justified by their statistical recovery properties in [73].

Taking a step back, we show that the above statistical isometry properties hold for matrices constructed from a large class of binary linear codes. This conclusion is made possible by properties of the distance distribution of codes that we study in the next sections.

6.2 Statistical isometry properties of matrices from codes

In this section, we address the task of constructing matrices with statistical recovery properties. The matrices that we examine are constructed from binary codes as in Sec. 5.2.2.

Let $\mathcal{C} \subset \{0,1\}^m$ be an (m, N) code. Surprisingly, we will find that the only condition required from the code $\mathcal{C}$ to yield a statistical RIP matrix is that the first two moments of its distance distribution are the same as those of a random linear code. This is ensured if the dual distance $d^\perp(\mathcal{C}) \geq 3$ which is not a very restrictive condition. In the case of linear codes, this condition can be stated as a requirement that the generator matrix of $\mathcal{C}$ have no identical columns (such linear codes are called projective).

We rely on the concepts of the distance distribution and dual distance of codes, which apply both to linear and unrestricted codes (see Sec. 1.4). To remind ourselves, the distance distribution of an (m, N) code $\mathcal{C}$ is the set of numbers $(A_0 = 1, A_1, \ldots, A_m)$ such that

$$A_w = \frac{1}{N}|\{(\boldsymbol{x}_1, \boldsymbol{x}_2) \in \mathcal{C}^2 : \mathrm{d}(\boldsymbol{x}_1, \boldsymbol{x}_2) = w\}|.$$

Let $d^\perp$ be the dual distance of the code $\mathcal{C}$.

We will need moments of the distance distribution of binary codes. If the dual distance of the code satisfies $d^\perp > l$ for some l, it is possible to find the exact values of the first l moments, which do not depend on the code, and are equal to the moments of the distance distribution of a code from the random parity-check matrix ensemble (the binomial distance distribution).

Lemma 6.2.1. (Pless power moment identities) *Let $\mathcal{C}$ be a binary code of length n and suppose that $d^{\perp}(\mathcal{C}) > l$. Then*

$$\sum_{w=0}^{m} \frac{A_w}{N}\Big(w - \frac{m}{2}\Big)^l = \frac{1}{2^m}\sum_{w=0}^{m}\binom{m}{w}\Big(w - \frac{m}{2}\Big)^l. \tag{6.2}$$

Proof. We use a version of these identities that relates to central moments of the distance distribution of the code. For linear codes a proof is given in [80, p. 132]. With minimal changes it also applies to general codes. □

In the particular case of $l = 2$ we can compute these moments directly. The following lemma will be useful later.

Lemma 6.2.2. *Let $\mathcal{C}$ be an (m, N) binary code such $d^{\perp}(\mathcal{C}) \geq 3$. Suppose that a pair of codewords $\boldsymbol{x}_1, \boldsymbol{x}_2$ is chosen randomly and uniformly out of the $N(N-1)$ such pairs and let $Z = \mathrm{d}(\boldsymbol{x}_1, \boldsymbol{x}_2)$. Then*

$$\mathbb{E}\Big(1 - \frac{2Z}{m}\Big)^2 = \frac{N-m}{m(N-1)} \tag{6.3}$$

and

$$\mathbb{E}\Big|1 - \frac{2Z}{m}\Big| \leq \sqrt{\frac{N-m}{m(N-1)}}. \tag{6.4}$$

Proof. Inequality (6.4) is immediate from (6.3) because $\mathbb{E}|\xi| \leq (\mathbb{E}|\xi|^2)^{1/2}$ for any random variable ξ. What is left to prove is (6.3) which is done below.

The probability that $\mathrm{d}(\boldsymbol{x}_1, \boldsymbol{x}_2) = w$ satisfies

$$\Pr(Z = w) = \frac{|\{(\boldsymbol{x}_1, \boldsymbol{x}_2) \in \mathcal{C}^2 : \mathrm{d}(\boldsymbol{x}_1, \boldsymbol{x}_2) = w\}|}{N(N-1)} = \frac{A_w}{N-1}.$$

Therefore,

$$\begin{aligned}\mathbb{E}\Big(1 - \frac{2Z}{m}\Big)^2 &= \sum_{w=1}^{m}\frac{A_w}{N-1}\Big(1 - \frac{2w}{m}\Big)^2 \\ &= \frac{1}{N-1}\Big(\sum_{w=1}^{m} A_w - \frac{4}{m}\sum_{w=1}^{m} wA_w + \frac{4}{m^2}\sum_{w=1}^{m} w^2 A_w\Big)\end{aligned}$$

For $d^{\perp}(\mathcal{C}) \geq 3$ we have [80, p. 131]

$$\sum_{w=1}^{m} wA_w = \frac{mN}{2}, \quad \sum_{w=1}^{m} w^2 A_w = \frac{Nm(m+1)}{4}.$$

Hence,

$$\mathbb{E}\Big(1-\frac{2w}{m}\Big)^2 = \frac{1}{N-1}\Big(N-1-\frac{4}{m}\frac{mN}{2}+\frac{4}{m^2}\frac{Nm(m+1)}{4}\Big)$$
$$= \frac{N-m}{m(N-1)}.$$

□

The next lemma gives an approximation to the central moments of binomial distribution.

Lemma 6.2.3. *For $l \geq 2$,*

$$\frac{1}{2^m}\sum_{w=0}^{m}\binom{m}{w}\Big(w-\frac{m}{2}\Big)^l = \frac{l!}{2^{l/2}(l/2)!}\Big(\frac{m}{4}\Big)^{l/2} + O(m^{l/2-1})$$
$$\leq \sqrt{2}\Big(\frac{l}{e}\Big)^{l/2}\Big(\frac{m}{4}\Big)^{l/2} + O(m^{l/2-1}).$$

Proof. Central moments of the binomial distribution form a well-studied subject. We quoted the first equality above from [105]. To prove the second inequality, we use the Stirling formula for the factorials [80, p. 309] to compute that for all $l \geq 2$,

$$\frac{l!}{2^{l/2}(l/2)!} \leq \sqrt{2}\Big(\frac{l}{e}\Big)^{\frac{l}{2}}.$$

□

6.2.1 SRIP from codes

We are now ready to state one of the main theorems of this chapter.

Theorem 6.2.4. *Let $\mathcal{C}$ be an (m, N) code with $d^{\perp}(\mathcal{C}) \geq l$, l even, and let Φ be the sampling matrix constructed from it. Suppose that*

$$m \geq \frac{2lk^{2+2/l}}{\delta^2 e\varepsilon^{2/l}}.$$

Then Φ is (k, δ, ε)-SRIP.

Proof. Let $I \subset [N]$ be a uniformly random k-subset. Note that the matrix $\Phi_I^T\Phi_I$ is real symmetric; therefore from the Rayleigh-Ritz theorem [64, Thm. 4.2.2], for any $\boldsymbol{x} \in \mathbb{R}^k$,

$$\lambda_{\min} \leq \frac{\|\Phi_I\boldsymbol{x}\|_2^2}{\|\boldsymbol{x}\|_2^2} \leq \lambda_{\max},$$

where $\lambda_{\min}$ and $\lambda_{\max}$ are the minimum and maximum eigenvalues of $\Phi_I^T\Phi_I$.

For brevity we write $I_i = I \setminus \{i\}$. We again rely on the Gershgorin theorem, Thm. 5.2.1. For any eigenvalue λ of $\Phi_I^T \Phi_I$,

$$|\lambda - 1| \leq \sum_{j \in I_i} |\phi_i^T \phi_j|.$$

for some $i \in I$. The remaining task is to show that the probability

$$\Pr\Big(\exists i \in I : \sum_{j \in I_i} |\phi_i^T \phi_j| > \delta\Big) < \varepsilon,$$

which will imply that all eigenvalues of $\Phi_I^T \Phi_I$ belong to the interval $[1-\delta, 1+\delta]$ with probability at least $1-\varepsilon$. This will prove the theorem.

We have

$$\begin{aligned}
\Pr\Big(\exists i \in I : \sum_{j \in I_i} |\phi_i^T \phi_j| > \delta\Big) &\leq k \Pr\Big(\sum_{j \in I_i} |\phi_i^T \phi_j| > \delta\Big) \\
&\leq k \frac{1}{\delta^l} \mathbb{E}\Big(\sum_{j \in I_i} |\phi_i^T \phi_j|\Big)^l \\
&= k \frac{(k-1)^l}{\delta^l} \mathbb{E}\Big(\frac{1}{k-1} \sum_{j \in I_i} |\phi_i^T \phi_j|\Big)^l \\
&\leq \frac{k(k-1)^{l-1}}{\delta^l} \mathbb{E} \sum_{j \in I_i} (\phi_i^T \phi_j)^l \\
&= \frac{k(k-1)^{l-1}}{\delta^l} \mathbb{E} \sum_{j \in I_i} \Big(1 - \frac{2\mathrm{d}(\boldsymbol{c}_i, \boldsymbol{c}_j)}{m}\Big)^l.
\end{aligned}$$

Here $\boldsymbol{c}_i, \boldsymbol{c}_j$ are the codewords of $\mathcal{C}$ that correspond to ϕ_i, ϕ_j. The expectation on the last line is taken with respect to the choice of a k-subset I. We claim that the value of the expectation does not change if instead, the vectors $\boldsymbol{c}_i, \boldsymbol{c}_j$ are chosen from $\mathcal{C}$ uniformly without replacement. Write out the expectation on the choice of I:

$$\begin{aligned}
\mathbb{E} \sum_{j \in I_i} \Big(1 - \frac{2\mathrm{d}(\boldsymbol{c}_i, \boldsymbol{c}_j)}{m}\Big)^l &= \sum_{i_1 < i_2 < \cdots < i_k} \frac{1}{\binom{N}{k}} \sum_{j=2}^{k} \Big(1 - \frac{2\mathrm{d}(\boldsymbol{c}_{i_1}, \boldsymbol{c}_{i_j})}{m}\Big)^l \\
&= \frac{1}{k! \binom{N}{k}} \sum_{i_1 \neq i_2 \neq \cdots \neq i_k} \sum_{j=2}^{k} \Big(1 - \frac{2\mathrm{d}(\boldsymbol{c}_{i_1}, \boldsymbol{c}_{i_j})}{m}\Big)^l \\
&= \frac{1}{N(N-1)} \sum_{j=2}^{k} \sum_{i_1=1}^{N} \sum_{i_j \neq i_1} \Big(1 - \frac{2\mathrm{d}(\boldsymbol{c}_{i_1}, \boldsymbol{c}_{i_j})}{m}\Big)^l \\
&= (k-1) \mathbb{E}\Big(1 - \frac{2\mathrm{d}(\boldsymbol{c}_i, \boldsymbol{c}_i)}{m}\Big)^l, \qquad (6.5)
\end{aligned}$$

where the expectation on the last line is defined with respect to a pair of uniform distinct random vectors from $\mathcal{C}$. Next,

$$\begin{aligned}\mathbb{E}\Big(1-\frac{2\mathrm{d}(\boldsymbol{c}_i,\boldsymbol{c}_i)}{m}\Big)^l &= \Big(\frac{2}{m}\Big)^l \sum_{w=1}^{m} \frac{A_w}{N-1}\Big(w-\frac{m}{2}\Big)^l \\ &= \Big(\frac{2}{m}\Big)^l \frac{N}{N-1}\Big[\sum_{w=0}^{m}\frac{A_w}{N}\Big(w-\frac{m}{2}\Big)^l - \frac{1}{N}\Big(\frac{m}{2}\Big)^l\Big] \\ &= \Big(\frac{2}{m}\Big)^l \frac{N}{N-1}\Big[\frac{1}{2^m}\sum_{w=0}^{m}\binom{m}{w}\Big(w-\frac{m}{2}\Big)^l - \frac{1}{N}\Big(\frac{m}{2}\Big)^l\Big],\end{aligned}$$

where on the last line we have used Lemma 6.2.1 as $l < d^{\perp}(\mathcal{C})$. Now invoke Lemma 6.2.3 to compute

$$\mathbb{E}\Big(1-\frac{2\mathrm{d}(\boldsymbol{c}_i,\boldsymbol{c}_i)}{m}\Big)^l \leq \Big(\frac{l}{e}\Big)^{l/2}\frac{\sqrt{2}}{m^{l/2}}(1+o(1)).$$

This implies the following inequality:

$$\begin{aligned}\Pr\Big(\exists i\in I : \sum_{j\in I_i}|\phi_i^T\phi_j| > \delta\Big) &\leq \sqrt{2}k\Big(\frac{k-1}{\delta}\Big)^l\Big(\frac{l}{em}\Big)^{l/2}(1+o(1)) \\ &< \varepsilon,\end{aligned}$$

where in the last step we used the assumption for m. □

We can strengthen the previous theorem for some values of k using a stronger inequality than the Markov inequality. Namely, if we have control of the width of the code $\mathcal{C}$ then the number of samples m can be made proportional to $\log 1/\varepsilon$ rather than to $1/\varepsilon^{2/l}$. To be specific, we have:

Theorem 6.2.5. *Let k satisfies $k < 2\ln N\log(k/\varepsilon)$ and $m \geq 8(k/\delta^2)\log(k/\varepsilon)\ln N$. Suppose that $\mathcal{C}$ is a linear (m,N) code of width $\frac{m\delta}{2\sqrt{2k\log(k/\varepsilon)}}$ and $d^{\perp}(\mathcal{C}) > 2$, and let Φ be the sampling matrix constructed from it. Then Φ is (k,δ,ε)-SRIP.*

Proof. The proof of the theorem relies on identifying a martingale sequence and then using the Azuma-Hoeffding inequality. Let I be chosen randomly and uniformly from $[N]$. We use the same first few steps as in the proof of Theorem 6.2.4. We have,

$$\begin{aligned}\Pr\Big(\exists i\in I : \sum_{j\in I_i}|\phi_i^T\phi_j| > \delta\Big) &\leq k\Pr\Big(\sum_{j\in I_i}|\phi_i^T\phi_j| > \delta\Big) \\ &= k\Pr\Big(\frac{2}{m}\sum_{j\in I_i}\Big|\frac{m}{2}-\mathrm{d}(\boldsymbol{c}_i,\boldsymbol{c}_j)\Big| > \delta\Big) \\ &= k\Pr\Big(\sum_{j=1}^{k-1}Y_j > m\delta/2\Big),\end{aligned}$$

where $Y_j = |\frac{m}{2} - \mathrm{d}(\boldsymbol{c}, \boldsymbol{c}_j)|, j = 1, \ldots, k-1$ and $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_{k-1}$ are codewords chosen uniformly from $\mathcal{C} \setminus \{\boldsymbol{c}\}$ in that order without replacement. We want the above probability to be less than ε.

Define the random variables Z_i, $0 \leq i \leq k-1$ as follows.

$$Z_0 = 0 \quad \text{with probability } 1,$$

and for $1 \leq i \leq k-1$

$$Z_i = \sum_{j=1}^{i} Y_j - \sum_{j=1}^{i} \mathbb{E}Y_j.$$

It is easy to verify that $\mathbb{E}[Z_i | Z_{i-1}] = Z_{i-1}$. Moreover, from the condition on the width of $\mathcal{C}$ we have

$$|Z_i - Z_{i-1}| = |Y_i - \mathbb{E}Y_i| \leq \frac{m\delta}{2\sqrt{2k \log(k/\varepsilon)}} \text{ a.s.}$$

Therefore, the sequence Z_i, $0 \leq i \leq k-1$ forms a bounded martingale, and the Azuma-Hoeffding large deviations bound [4, 62] applies. We obtain

$$\Pr(Z_{k-1} > a) \leq \exp\Big(- \frac{16a^2 k \log(k/\varepsilon)}{(k-1)m^2\delta^2}\Big).$$

The code $\mathcal{C}$ satisfies (6.4)

$$\sum_{j=1}^{k-1} \mathbb{E}Y_j \leq k\sqrt{m}/2.$$

We need to prove that $\Pr(\sum_{j=1}^{k-1} Y_j > m\delta/2) < \varepsilon/k$. Let $a = m\delta/2 - k\sqrt{m}/2$. We have

$$\begin{aligned}
\Pr\Big(\sum_{j=1}^{k-1} Y_j > \frac{m\delta}{2}\Big) &= \Pr\Big(Z_{k-1} > \frac{m\delta}{2} - \sum_{j=1}^{k-1} \mathbb{E}Y_j\Big) \\
&\leq \Pr(Z_{k-1} > a) \\
&\leq \exp\Big(- \frac{16m^2\delta^2 k \log(k/\varepsilon)(1 - k/\sqrt{\delta^2 m})^2}{4(k-1)m^2\delta^2}\Big) \\
&\leq \exp\Big(- 4\log(k/\varepsilon)\Big(1 - \sqrt{\frac{k}{8 \ln N \log(k/\varepsilon)}}\Big)^2\Big) \\
&< \exp\Big(- 4\log(k/\varepsilon)(1 - 1/2)^2\Big) \\
&= \varepsilon/k,
\end{aligned}$$

where we have used the fact that $k < 2 \ln N \log(k/\varepsilon)$. □

We comment that the matrix Φ of the above theorem can be constructed deterministically with complexity polynomial in N and k. To do that we need to construct a linear (m, N) code $\mathcal{C}$ with length $m \geq 8(k/\delta^2)\log(k/\varepsilon)\ln N$, width $\frac{m\delta}{2\sqrt{2k\log(k/\varepsilon)}}$ and $d^{\perp}(\mathcal{C}) > 2$. We can modify Thm. 5.2.4 to construct a code with the claimed length, width and dual distance. The modification deals with the issue of $d^{\perp} > 2$ which is guaranteed by constructing a generator matrix with *distinct* columns. We omit the specific details to have the focus on the main part of the theorem.

6.2.2 SURP from codes

Next we prove that matrices constructed from some binary codes have the SURP. Here we rely on the ideas of [100],making some changes related to our construction of sampling matrices.

Theorem 6.2.6. *Let $\mathcal{C}$ be an (m, N) code and let $d^{\perp}(\mathcal{C}) > l$ for some even l. Suppose that*

$$m \geq \max\Big(\frac{6lk^{2+2/l}}{\delta^2 e\varepsilon^{2/l}}, \frac{2kl}{\varepsilon^{2/l}(1-\delta)}\Big).$$

Then the sampling matrix Φ constructed from $\mathcal{C}$ is (k, ε)-SURP and $(k, \delta, \varepsilon/3)$-SRIP.

Since $0 < \delta < 1$ is an absolute constant (for instance, we can assume that $\delta \leq \sqrt{2} - 1$ (5.2)), we assume below $k > \delta^2/(1-\delta)$.

We need the following lemma in which $\mathrm{Col}(A)$ denotes the column space over $\mathbb{R}$ of the matrix A.

Lemma 6.2.7. *Suppose that a sampling matrix Φ is constructed from a binary (m, N) code $\mathcal{C}$ with $d^{\perp}(\mathcal{C}) > l$ for some even l. Suppose further that Φ is $(k, \delta, \varepsilon/3)$-SRIP. Let $I, J \subset [N]$ be two k-subsets. If I is uniformly random, then*

$$\Pr(\dim(\mathrm{Col}(\Phi_I) \cap \mathrm{Col}(\Phi_J)) \leq k-1) \geq 1 - \frac{2\varepsilon}{3}$$

whenever $m > \frac{2kl}{\varepsilon^{2/l}(1-\delta)}$.

Proof. Without loss of generality we assume the dimension of column space of Φ_J is k, otherwise there is nothing to prove. Let $Q_I = \Phi_I(\Phi_I^T\Phi_I)^{-1}\Phi_I^T$ be the orthogonal projection on the space $\mathrm{Col}(\Phi_I)$ (indeed, $Q_I^2 = Q_I$ and $Q_I^T = Q_I$). Let $J \neq I$ and $i \in J \setminus I$. We will prove that $\|Q_I\phi_i\|^2 < \|\phi_i\|^2 = 1$, which will imply that $\phi_i \notin \mathrm{Col}(\Phi_I)$.

Since Φ satisfies $(k, \delta, \varepsilon/3)$-SRIP, the absolute value of any eigenvalue of $\Phi_I^T\Phi_I$ is at least $1-\delta$ with probability at least $1-\varepsilon/3$. Therefore, with probability at least

$1-\varepsilon/3$,

$$\begin{aligned}\|Q_I\phi_i\|^2 = \phi_i^T Q_I^T Q_I \phi_i &= (\Phi_I^T\phi_i)^T(\Phi_I^T\Phi_I)^{-1}(\Phi_I^T\phi_i)\\ &\leq \frac{\|\Phi_I^T\phi_i\|^2}{1-\delta}.\end{aligned}$$

Denote by $\boldsymbol{c}_i$ the codeword of $\mathcal{C}$ that corresponds to the column ϕ_i. We have

$$\|\Phi_I^T\phi_i\|^2 = \sum_{j\in I}|\phi_j^T\phi_i|^2 = \sum_{j\in I}\Big(1-\frac{2\mathrm{d}(\boldsymbol{c}_j,\boldsymbol{c}_i)}{m}\Big)^2$$

and therefore for any even $l \leq d^\perp$,

$$\begin{aligned}\|\Phi_I^T\phi_i\|^l &= \Big(\sum_{j\in I}\Big(1-\frac{2\mathrm{d}(\boldsymbol{c}_j,\boldsymbol{c}_i)}{m}\Big)^2\Big)^{l/2}\\ &\leq k^{l/2-1}\sum_{j\in I}\Big(1-\frac{2\mathrm{d}(\boldsymbol{c}_j,\boldsymbol{c}_i)}{m}\Big)^l.\end{aligned}$$

Hence,

$$\begin{aligned}\mathbb{E}\|\Phi_I^T\phi_i\|^l &\leq k^{l/2-1}\mathbb{E}\sum_{j\in I}\Big(1-\frac{2\mathrm{d}(\boldsymbol{c}_j,\boldsymbol{c}_i)}{m}\Big)^l\\ &\leq \sqrt{2}\Big(\frac{kl}{em}\Big)^{l/2}(1+o(1));\end{aligned}$$

here we have used Lemmas 6.2.1 and 6.2.3 and the derivation in (6.5) as in the proof of Theorem 6.2.4.

From the Markov inequality we further have

$$\begin{aligned}\Pr(\|\Phi_I^T\phi_i\|^2 \geq 1-\delta) &\leq \frac{1}{(1-\delta)^{l/2}}\mathbb{E}\|\Phi_I^T\phi_i\|^l\\ &\leq \sqrt{2}\Big(\frac{kl}{em(1-\delta)}\Big)^{l/2}(1+o(1))\\ &< \frac{\varepsilon}{3}\end{aligned}$$

where in the last step we used the assumption about m.

Now combining the facts above we conclude that with probability at least $1-2\varepsilon/3$,

$$\|Q_I\phi_i\|^2 \leq \frac{\|\Phi_I^T\phi_i\|^2}{1-\delta} < 1,$$

which proves the lemma. □

Proof of Theorem 6.2.6. The assumption on m together with Theorem 6.2.4 implies that Φ is $(k, \delta, \varepsilon/3)$-SRIP. Therefore, by (5.2), we have that $\Pr(I : \operatorname{rank}(\Phi_I) = k) \geq 1 - \varepsilon/3$, where the probability is computed with respect to the uniform choice of k-subsets I of $[N]$.

Let $\boldsymbol{x}$ be a k-sparse vector supported on I, let $\boldsymbol{s} = \Phi_I \boldsymbol{x}$ and let $\operatorname{rank}(\Phi_I) = k$. If there is a k-sparse vector $\boldsymbol{y}$ such that $\Phi_I \boldsymbol{y} = \boldsymbol{s}$ then the support of $\boldsymbol{y}$ is different from I. Therefore, if a k-sparse $\boldsymbol{x}$ is supported on some randomly chosen k-subset I and $\boldsymbol{y}$ is another k-sparse vector with the same sketch, then with probability at least $1 - \varepsilon/3$ the supports of $\boldsymbol{x}$ and $\boldsymbol{y}$ are different.

Assuming that $\boldsymbol{x}$ and $\boldsymbol{y}$ have different supports, we observe that the vector $\boldsymbol{s}$ lies in $\operatorname{Col}(\Phi_I) \cap \operatorname{Col}(\Phi_{I'})$ where $I' = \operatorname{supp}(\boldsymbol{y})$. By Lemma 6.2.7, with probability $\geq 1 - 2\varepsilon/3$ with respect to the uniform choice of I, the vector $\boldsymbol{s}$ lies in an at most $(k-1)$-dimensional subspace, and so does $\boldsymbol{x}$ whenever Φ_I has rank k. Now let us use the absolute continuity of the distribution $P_{\boldsymbol{z}}$ with respect to the Lebesgue measure. A random vector from $\mathbb{R}^k$ chosen with respect to $P_{\boldsymbol{z}}$ falls in a $(k-1)$-dimensional space with probability 0.

To conclude, a random vector $\boldsymbol{x}$ fails the unique recovery condition (6.1) either if there is $\boldsymbol{y}$ with the same support as $\boldsymbol{x}$ (probability $< \varepsilon/3$) or if Lemma 6.2.7 fails to hold (probability $< 2\varepsilon/3$), so Φ is (k, ε)-SURP. □

Remark 3. The two aspects of the compressed sensing problem analyzed in this thesis correspond to the recovery of all sparse signals and 'almost all' sparse signals. They are close in spirit to combinatorial vs. probabilistic error correction for transmission over noisy channels. As in information theory, moving from the adversarial to the statistical scenario enables us to use much fewer samples for reliable signal recovery.

From previous works [10, 60, 88] it is known that RIP matrices can be constructed from binary codes (as well as from codes over other alphabets). However, constructions of this kind are limited by the classical bounds on error correcting codes (binary or spherical). We have shown that it is possible to surpass these limitations by permitting a small proportion of sparse signals for which the sampling operator fails to show near-isometry. In particular, *any binary code* with $d^{\perp} \geq l$ can be used to construct an $m \times N$ sampling matrix with (k, δ, ε)-SRIP and (k, ε)-SURP, with $m = O(k^{2+2/l}/(\varepsilon^{1/l}\delta)^2)$. Note that the dual distance of the Delsarte-Goethals codes employed in [29] equals 8. In that case the code structure enables one to prove the reliable recovery properties. Here we used standard properties of codes to be able to make a more general claim.

The dual distance of the code is known to control how far the code is from a random code. Examples of this principle include the behavior of moments of the distance distribution [80] as well as of the CDF of this distribution (Sidelnikov's theorem [80, pp.295ff]). If the code is sufficiently random (has large $d^{\perp}$), the eigenvalue statistic should match that of random Gaussian matrices. This has been stud-

ied experimentally in [6]. A proof has been recently announced in [8].

Part IV

Permutations

CHAPTER 7

Codes in Permutations: Bounds

7.1 Introduction

So far in this dissertation, we considered codes in the binary Hamming space $\{0,1\}^n$. The underlying geometric idea of constructing good packings extends to other metric spaces such as the sphere in $\mathbb{R}^n$ and a range of finite spaces of diverse nature. One of these spaces is the set of permutations of n elements, i.e., the *symmetric group* $\mathfrak{S}_n$. Codes in permutations form a classical subject of coding theory. Various metric functions on $\mathfrak{S}_n$ have been considered, giving rise to diverse combinatorial problems. The most frequently studied metric on $\mathfrak{S}_n$ is the Hamming distance. Codes in $\mathfrak{S}_n$ with the Hamming distance, traditionally called permutation arrays, have been the subject of a large number of papers; see, e.g., the works by Blake et al. [20], Colbourn et al. [38] and the survey by [30]. In this part of the dissertation we are interested in a different metric on $\mathfrak{S}_n$ which is defined below. For any two $a < b \in \mathbb{Z}$, let

$$[a,b] = \{a, a+1, a+2, \ldots, b\}.$$

If $a \geq b = 1$ we write $[a]$ instead of $[1,a]$.

Definition 7.1.1. [70] *Let $\sigma = (\sigma(1), \ldots, \sigma(n))$ be a permutation of the set $[n]$. The* Kendall tau distance $\mathrm{d}_\tau(\sigma,\pi)$ *from σ to another permutation π is defined as the minimum number of transpositions of pairwise adjacent elements required to change σ into π.*

Denote by $\mathcal{X}_n = (\mathfrak{S}_n, \mathrm{d}_\tau)$ the metric space of permutations on n elements equipped with the distance d_τ. We use the vector notation for permutations: for instance (2,1,3) refers to the permutation $\binom{1,2,3}{2,1,3}$. For a permutation $\sigma = (\sigma(1), \ldots, \sigma(n))$ we denote its inverse by $\sigma^{-1} = (\sigma^{-1}(1), \ldots, \sigma^{-1}(n))$, where if $\sigma(i) = j$ then $\sigma^{-1}(j) = i$.

The Kendall distance originates in statistics and has been adopted as a measure of quality of codes under the so-called rank modulation scheme, first considered by Chadwick and Kurz [35]. In this scheme, data is encoded into permutations

of n elements while the information is carried by the relative magnitude (rank) of elements in the transmitted sequences rather than by the absolute value of the elements. The motivation for considering this scheme in [35] stems from systems in which transmitted signals are subjected to impulse noise that changes the value of the signal substantially but has less effect on the relative magnitude of the neighboring signals.

7.1.1 Flash memory and the rank modulation scheme

Recently substantial attention in the literature was devoted to coding problems for non-volatile memory devices, including error correction in various models as well as data management in memories [10,66–68]. Non-volatile memories, in particular, flash memory devices store data by injecting charges of varying levels in memory cells that form the device. Consider a block of n cells in floating-gate flash memory. Each cell is capable of storing some amount of electrical charge, called the capacity of the cell. One can quantize this capacity of charge storage into q levels and write information in the memory using a q-ary alphabet: each level of charge represents an element in $\{0, \ldots, q-1\}$. Reliability of the data stored in flash memory is affected by the drift in the charge of the cells caused for instance by ageing devices or other reasons. Because of the drift, after some amount of time all (or most) of the cells will contain erroneous values, and conventional error-correcting coding will fail to recover the message written into the memory. It is advantageous to encode the message into the ranking of the charge-levels of n cells (i.e., a permutation). Recently (and independently of [35]) codes in permutation and the *rank modulation scheme* was suggested by Jiang et al. [67, 68] as a means of efficient writing of information into flash memories. Errors occur in the data stored in rank modulation scheme only if the loss of charge in different cells occurs at different speed. In this case errors introduced in the data are adequately accounted for by tracking the Kendall distance between the permutations. Details of both the writing and the error processes in memory are given in [68] and references in that paper.

The focus of our work is on bounds and constructions of codes in the Kendall space $\mathcal{X}_n$. The size of the maximum packing in the space $\mathcal{X}_n$ is related to finding the volume of the sphere (see (1.3) in the Introduction). Spheres in the Kendall space were studied by analytic means in a number of earlier works [79, 81] relying on the well-known correspondence of permutations and their inversion vectors; however it turned out that code bounds that can be obtained from these works do not cover the range of parameters of interest to us. Regarding specific code families for correcting Kendall errors, the only previous work is that by Jiang et al. [68] who constructed a codes of cardinality $M \geq \frac{1}{2}(n-1)!$ that correct one Kendall error.

An (n, M, d) code $\mathcal{C} \subset \mathcal{X}_n$ is a set of M permutations in which any two distinct permutations are at least d distance units apart. To distinguish these codes from codes in the Hamming space, we call them rank permutation codes (or rank modulation codes) . In this chapter we discuss several possible ways to bound the

size of codes for rank modulation, i.e. bounding M as a function of n and d. We derive a Singleton-type bound and sphere-packing bounds on such codes. Since the maximum value of the distance in $\mathcal{X}_n$ is $\binom{n}{2}$, this leaves a number of possibilities for the scaling rate of the distance for asymptotic analysis, ranging from $d = O(n)$ to $d = \Theta(n^2)$. These turn out to be the two extremes for the size of optimal rank permutation codes. Namely, earlier work in combinatorics of permutations implies that the log-cardinality of a code with distance $d = \Theta(n^2)$ occupies a vanishing proportion of $\log |\mathcal{X}_n|$ while a code of distance $O(n)$ can take a close-to-one proportion. We cover the intermediate cases, showing that the size of optimal codes with distance $d \sim n^{1+\varepsilon}, 0 < \varepsilon < 1$ scales as $\exp((1-\varepsilon)n \ln n)$. It is interesting that unlike many other asymptotic coding problems, the Kendall space of permutations affords an exact answer for the growth rate of the size of optimal codes. The proof of the bounds relies on weight-preserving embeddings of $\mathcal{X}_n$ into other metric spaces which provide insights into the asymptotic size of codes.

We also show the existence of a family of rank permutation codes that correct a constant number of errors and have size within a constant factor of the sphere packing bound. The construction relies on the well-known Bose-Chowla Theorem in additive number theory.

The results of this chapter appeared in [10].

7.2 Basics of permutations

We begin with a description of basic properties of the distance d_τ such as its relation to the number of inversions in the permutation, and weight-preserving embeddings of $\mathfrak{S}_n$ into other metric spaces. Their proofs and a detailed discussion are found for instance in the books by Comtet [39] or Knuth [71, Sect. 5.1.1].

The distance d_τ is a right-invariant metric which means that $\mathrm{d}_\tau(\sigma_1, \sigma_2) = \mathrm{d}_\tau(\sigma_1\sigma, \sigma_2\sigma)$ for any $\sigma, \sigma_1, \sigma_2 \in \mathfrak{S}_n$ where the operation is the usual multiplication of permutations. Therefore, we can define the weight of the permutation σ as its distance to the identity permutation $e = (1, 2, \ldots, n)$.

Because of the invariance, the graph whose vertices are indexed by the permutations and edges connect permutations one Kendall step apart, is regular of degree $n-1$. At the same time it is not distance-regular, and so the machinery of algebraic combinatorics does not apply to the analysis of the code structure. The diameter of the space $\mathcal{X}_n$ equals $N \triangleq \binom{n}{2}$ and is realized by pairs of opposite permutations such as $(1, 2, 3, 4)$ and $(4, 3, 2, 1)$.

The main tool to study properties of d_τ is provided by the inversion vector of the permutation. An inversion in a permutation $\sigma \in \mathfrak{S}_n$ is a pair $(\sigma(i), \sigma(j))$ such that $i < j$ and $\sigma(i) > \sigma(j)$. It is easy to see that $\mathrm{d}_\tau(\sigma, e) = I(\sigma)$, the total number of inversions in σ. Therefore, for any two permutations σ_1, σ_2 we have $\mathrm{d}_\tau(\sigma_1, \sigma_2) = I(\sigma_2\sigma_1^{-1}) = I(\sigma_1\sigma_2^{-1})$. In other words,

$$\mathrm{d}_\tau(\sigma, \pi) = |\{(i, j) \in [n]^2 : i \neq j, \pi^{-1}(i) > \pi^{-1}(j), \sigma^{-1}(i) < \sigma^{-1}(j)\}|.$$

To a permutation $\sigma \in \mathfrak{S}_n$ we associate an *inversion vector* $\boldsymbol{x}_\sigma \in \mathcal{G}_n \triangleq [0,1] \times [0,2] \times \cdots \times [0,n-1]$, where $\boldsymbol{x}_\sigma(i) = |\{j : j < i+1, \sigma(j) > \sigma(i+1)\}|, i = 1, \ldots, n-1$. It is well known that the mapping from permutations to the space of inversion vectors is one-to-one, and any permutation can be easily reconstructed from its inversion vector. Let $J : \mathcal{G}_n \to \mathfrak{S}_n$ be the map such that $J(\boldsymbol{x}_\sigma) = \sigma$: for instance, $J((1,0,1,2,0,2,0,1)) = (2,1,6,4,3,7,5,9,8)$. Moreover,

$$I(\sigma) = \sum_{i=1}^{n-1} \boldsymbol{x}_\sigma(i). \tag{7.1}$$

For the type of errors that we consider below we introduce the following ℓ_1 distance function on $\mathcal{G}_n$:

$$\mathrm{d}_1(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i=1}^{n-1} |\boldsymbol{x}(i) - \boldsymbol{y}(i)| \qquad (\boldsymbol{x}, \boldsymbol{y} \in \mathcal{G}_n) \tag{7.2}$$

where the computations are performed over the integers, and write $\|\boldsymbol{x}\|$ for the corresponding weight function (this is not a properly defined norm because $\mathcal{G}_n$ is not a linear space). For instance, let $\sigma_1 = (2,1,4,3), \sigma_2 = (2,3,4,1)$, then $\boldsymbol{x}_{\sigma_1} = (1,0,1), \boldsymbol{x}_{\sigma_2} = (0,0,3)$. To compute the distance $\mathrm{d}_\tau(\sigma_1, \sigma_2)$ we find

$$I(\sigma_2 \sigma_1^{-1}) = I((1,4,3,2)) = \|(0,1,2)\| = 3.$$

Observe that the mapping $\sigma \to \boldsymbol{x}_\sigma$ is a weight-preserving bijection between $\mathcal{X}_n$ and the set $\mathcal{G}_n$. At the same time, this mapping is not distance-preserving. However, a weaker property is true, namely,

$$\mathrm{d}_\tau(\sigma_1, \sigma_2) \geq \mathrm{d}_1(\boldsymbol{x}_{\sigma_1}, \boldsymbol{x}_{\sigma_2}). \tag{7.3}$$

Indeed, if the Kendall distance between two permutations is 1, then the ℓ_1 distance between the corresponding two inversion vectors is 1 as well. The converse is not necessarily true.

From (7.3), if there exists a code in $\mathcal{G}_n$ with ℓ_1 distance d then there exists a code of the same size in $\mathcal{X}_n$ with Kendall distance at least d.

Another embedding of $\mathcal{X}_n$ is given by mapping each permutation to a binary N-dimensional vector $\boldsymbol{a}$ whose coordinates are indexed by the pairs $(i,j) \subset [n]^2, i < j$, and $a_{(i,j)} = 1$ if the pair (i,j) is an inversion and $a_{(i,j)} = 0$ otherwise. Clearly the Hamming weight of $\boldsymbol{a}$ equals $I(\sigma)$, and so this mapping is an isometry between $\mathcal{X}_n$ and a subset of the Hamming space $\{0,1\}^N$. This mapping was first considered in [36].

7.3 Bounds on the size of rank permutation codes

Let $X(n,d)$ be the maximum size of the code in $\mathcal{X}_n$ with distance d. For the purposes of asymptotic analysis we define the rate of a code $\mathcal{C} \subset \mathcal{X}_n$ of size M as

$$R(\mathcal{C}) = \frac{\ln M}{\ln n!}. \tag{7.4}$$

Let

$$C(d) = \lim_{n\to\infty} \frac{\ln X(n,d)}{\ln n!}$$

be the *capacity* of rank permutation codes of distance d (our proof of Theorem 7.3.1 will imply that the limit exists). The main result of this section is given in the following theorem whose proof is given in Sections 7.3.2 and 7.3.3 below.

Theorem 7.3.1.

$$C(d) = \begin{cases} 1 & \text{if } d = O(n) \\ 1-\varepsilon & \text{if } d = \Theta(n^{1+\varepsilon}),\ 0 < \varepsilon < 1 \\ 0 & \text{if } d = \Theta(n^2). \end{cases} \tag{7.5}$$

Remark 4. As will be seen from the proof, the equality $C(d) = 1-\varepsilon$ holds under a slightly weaker condition, namely, $d = n^{1+\varepsilon}\alpha(n)$, where $\alpha(n)$ grows slower than any positive power of n.

7.3.1 A Singleton bound

Recall the well-known Singleton bound on the size of codes in the Hamming space over a q-ary alphabet, $q \geq 2$ [80]. Suppose A is such a code in Hamming space with distance d and length n. Let us delete coordinates $1, \ldots, d-1$ from every vector in A. In the resulting set, all the vectors are distinct, and their number is not more than the total number of vectors of length $n-d+1$, i.e., $|A| \leq q^{n-d+1}$. In this section we adapt this idea for the space $\mathcal{X}_n$.

Theorem 7.3.2. *Let $d > n-1$, then*

$$X(n,d) \leq \left\lfloor 3/2 + \sqrt{n(n-1) - 2d + 1/4} \right\rfloor !. \tag{7.6}$$

Proof. Let $\mathcal{C} \subset \mathcal{X}_n$ be an (n, M, d) code. Since the metric d_τ is right-invariant, we can assume that $\mathcal{C}$ contains the identity permutation e.

Let $k \leq n$ and let $\mathcal{C}_k \in \mathfrak{S}_k$ be a code derived from $\mathcal{C}$ in the following way. Let $\psi_k : \mathfrak{S}_n \to \mathfrak{S}_k$ be a mapping that acts on σ by deleting elements $k+1, \ldots, n$ from it. Thus, $\psi_k(\sigma)$ is a permutation on k elements that maintains the relative positions of the elements of $[k]$ given by σ.

Let k be the greatest number such that ψ_k is not injective. Then ψ_{k+1} is injective, and $M \leq (k+1)!$. Suppose that permutations $\sigma_1, \sigma_2 \in \mathfrak{S}_n$ are such that

$\psi_k(\sigma_1) = \psi_k(\sigma_2)$. Because of the last equality, none of the first k entries of the permutation $\sigma_2\sigma_1^{-1}$ contain pairs that form inversions. Therefore,

$$d \le \mathrm{d}_\tau(\sigma_1, \sigma_2) \le \binom{n}{2} - \binom{k}{2}.$$

This gives

$$k \le \frac{1 + \sqrt{4n(n-1) - 8d + 1}}{2},$$

which proves inequality (7.6). This estimate is nontrivial if

$$\frac{3}{2} + \sqrt{n(n-1) - 2d + 1/4} < n$$

which is equivalent to the condition $d > n - 1$. □

To gain an insight into this bound, let $d = \delta N$. Using the inequality $m! \le (m/2)^m$ in (7.6), we obtain the asymptotic inequality

$$X(n, \delta N) \le \exp(n(\ln n)\sqrt{1-\delta}(1 + c(\ln n)^{-1})),$$

where the constant c does not depend on n. As we will show in the next section, the $\sqrt{1-\delta}$ in this bound can in fact be improved to a quantity that decays as $(\ln n)^{-1}$ as n grows.

7.3.2 Sphere packing bounds

Sphere packing bounds on codes in the Kendall space $\mathcal{X}_n$ are related to the count of inversions in permutations. In this section we discuss several classic and new results in this area, showing that they imply the asymptotic scaling order of $C(d)$ for very small or very large values of d.

Denote by $\mathcal{B}_r(\mathcal{X}_n, e)$ the ball of radius r in $\mathcal{X}_n$ centered at the identity permutation e. It is evident that in $\mathcal{X}_n$ the volume of a ball of a given radius does not depend on the center of the ball. From (1.3),

$$\frac{n!}{|\mathcal{B}_{2r}(\mathcal{X}_n, e)|} \le X(n, 2r+1) \le \frac{n!}{|\mathcal{B}_r(\mathcal{X}_n, e)|}. \tag{7.7}$$

The embeddings of $\mathcal{X}_n$ into other metric spaces can be used to derive estimates of $X(n, d)$ based on these inequalities. In particular, estimating the volume of the ℓ_1-metric ball in $[n]^n = \{1, \dots, n\}^n$ and using (7.12), both lower and upper bounds will follow from the embedding of $\mathcal{X}_n$ in the space $[n]^n$.

Let $K_n(k) = |\{\sigma \in \mathfrak{S}_n : I_n(\sigma) = k\}|$ be the number of permutations with k inversions. By (7.1), $K_n(k)$ is the number of solutions of the equation

$$\sum_{i=1}^{n-1} x_i = k, \quad \text{where } x_i \in [0, i].$$

Then clearly $K_n(k) = 0$ for $k > N$ and

$$K_n(k) = K_n\Big(N - k\Big) \quad \text{for } 0 \le k \le \frac{1}{2}N.$$

The number of inversions in a random permutation is asymptotically Gaussian with mean $\frac{1}{2}N$ and variance $\frac{2n^3+3n^2-5n}{72} \approx \frac{n^3}{36}$, [50, p.257]. This suggests that codes with distance greater than $\frac{1}{2}N$ cannot have large size. We show that this is indeed the case in Sect. 7.3.4.

The generating function for the numbers $K_n(k)$ has the form

$$K(z) = \sum_{k=0}^{\infty} K_n(k) z^k = \prod_{i=1}^{n} \frac{1 - z^i}{1 - z}. \tag{7.8}$$

For $1 \le k \le n$ the number of permutations with k inversions can be found explicitly [71]:

$$\begin{aligned} K_n(k) = &\binom{n+k-2}{k} - \binom{n+k-3}{k-2} \\ &+ \sum_{j \ge 2} (-1)^j \Big[\binom{n+k-u_j-1}{k-u_j} + \binom{n+k-u_j-j-1}{k-u_j-j} \Big], \end{aligned} \tag{7.9}$$

where $u_j = (3j^2 - j)/2$ and the summation extends for as long as the binomial coefficients are positive (it contains about $1.6\sqrt{k}$ terms). This implies that $|\mathcal{B}_1(\mathcal{X}_n, e)| = n$, and $X(n, 3) \le (n-1)!$. As shown in [81], for $n = k+m, k \ge 0$

$$K_n(k) = (0.289\ldots)\frac{2^{m+n-1}}{\sqrt{\pi m}}(1 + O(m^{-1})). \tag{7.10}$$

The case of $k > n$ is much more difficult to analyze. An obvious route for finding asymptotic approximation of $K_n(k)$ is to start with the integral representation of the coefficients of $K(z)$ (7.8). Namely, since $K(z)$ converges for every z in the finite plane, we can write

$$K_n(k) = \frac{1}{2\pi i} \oint \prod_{\ell=1}^{n} \Big(\frac{1 - z^\ell}{1 - z} \Big) z^{-k-1} dz.$$

where the integration is over any circle around the origin. In particular, for $|z| = 1$ we obtain

$$\begin{aligned} K_n(k) &= \frac{1}{2\pi} \int_{-\pi}^{\pi} \prod_{\ell=1}^{n} \frac{1 - e^{i\ell\omega}}{1 - e^{i\omega}} e^{-ik\omega} d\omega \\ &= \frac{2}{\pi} \int_0^{\pi/2} \cos\left(x\,(N - 2k)\right) \prod_{\ell=1}^{n} \frac{\sin(x\ell)}{\ell \sin x} dx. \end{aligned} \tag{7.11}$$

Asymptotic analysis of this expression involves saddle point calculations and is rather involved even in the particular cases for which it has been accomplished, see Margolius [81] and Louchard and Prodinger [79]. The next theorem is a combination of results of these works, stated here in the form suitable for our context.

Theorem 7.3.3. *There exist constants c_1 and c_2 such that*

$$\begin{aligned} K_n(k) &\le \exp(c_1 n) && \text{if } k = O(n), \\ K_n(k) &= n!/\exp(c_2 n) && \text{if } k = \Theta(n^2). \end{aligned}$$

The implicit constants in this theorem can be found in cited references. From this theorem and inequalities (7.7), we obtain the two boundary cases of the expression for $C(d)$ in (7.5).

7.3.3 Bounds from embedding in the ℓ_1 space

In this section we prove the main part of Theorem 7.3.1. Our idea is to derive bounds on $C(d)$ by relating the Kendall metric to the ℓ_1 metric on $\mathfrak{S}_n$. From the results of Diaconis and Graham [48],

$$1/2 D(\sigma_1, \sigma_2) \le \mathrm{d}_\tau(\sigma_1^{-1}, \sigma_2^{-1}) \le D(\sigma_1, \sigma_2). \tag{7.12}$$

where $D(\sigma_1, \sigma_2) = \sum_{i=1}^n |\sigma_1(i) - \sigma_2(i)|$. Therefore, existence of any code $\mathcal{C} \subset \mathfrak{S}_n$ with Kendall distance d must imply existence of a code $\mathcal{C}' = \{\sigma^{-1} : \sigma \in \mathcal{C}\}$ of same size that have ℓ_1 distance at least d. On the other hand existence of any code $\mathcal{C} \subset \mathfrak{S}_n$ with ℓ_1 distance d implies the code $\mathcal{C}' = \{\sigma^{-1} : \sigma \in \mathcal{C}\}$ will have Kendall distance at least $d/2$.

Remark 5. Define $T(\sigma_1, \sigma_2)$ to be the number of inversions of (not necessarily adjacent) symbols needed to change σ_1 into σ_2. Paper [48] in fact shows that

$$\mathrm{d}_\tau(\sigma_1^{-1}, \sigma_2^{-1}) \le D(\sigma_1, \sigma_2) - T(\sigma_1, \sigma_2)$$

which is a stronger inequality than the one given above. We however will not use it in the derivations below.

Proposition 7.3.4. *Let $\mathcal{B}_r([n]^n, \boldsymbol{x})$ be the metric ball of radius r with center at $\boldsymbol{x}$ in the space $[n]^n = \{1, 2, \ldots, n\}^n$ with the ℓ_1 metric. Then the maximum size of a code in $\mathcal{X}_n$ with distance d satisfies*

$$\frac{n!}{\max_{\boldsymbol{x} \in [n]^n} |\mathcal{B}_{2d-1}([n]^n, \boldsymbol{x})|} \le X(n, d) \le \frac{n^n}{\min_{\boldsymbol{x} \in [n]^n} |\mathcal{B}_t([n]^n, \boldsymbol{x})|},$$

where $t = \lfloor (d-1)/2 \rfloor$.

Proof. Under the trivial embedding $\mathfrak{S}_n \to [n]^n$ the ℓ_1 distance does not change, so any code $\mathcal{C}$ in $\mathfrak{S}_n$ with ℓ_1 distance d is also a code in $[n]^n$ with the same distance and as such, must satisfy the sphere-packing bound (see, (1.3)). Together with (7.12) this gives the upper bound of our statement.

Turning to the lower bound, let us perform the standard Gilbert-Varshamov procedure in the space of permutations with respect to the ℓ_1 distance (see Sect. 1.3), aiming for a code $\mathcal{D}$ with ℓ_1 distance m. The resulting code satisfies

$$|\mathcal{D}| \max_{\sigma\in\mathfrak{S}_n} |\mathcal{B}_{m-1}(\mathfrak{S}_n, \sigma)| \geq n!.$$

Since $|\mathcal{B}_r([n]^n, \sigma)| \geq |\mathcal{B}_r(\mathfrak{S}_n, \sigma)|$, we can replace the volume in $\mathfrak{S}_n$ with the volume in $[n]^n$ in the last inequality. In the space $\mathcal{X}_n$, the code $\mathcal{D}' = \{\sigma^{-1} : \sigma \in \mathcal{D}\}$ will then have Kendall distance at least $m/2$. □

Below we consider only spheres in the space $[n]^n$ and omit the reference to it from the notation $\mathcal{B}_r([n]^n, \cdot)$.

Lemma 7.3.5. *Let* $\mathbf{1} = (1, 1, \ldots, 1) \in [n]^n$. *Then for any* $\boldsymbol{z}, \boldsymbol{y} \in [n]^n$,

$$2^{-n}|\mathcal{B}_r(\boldsymbol{z})| \leq |\mathcal{B}_r(\mathbf{1})| \leq |\mathcal{B}_r(\boldsymbol{y})|.$$

Proof. Suppose that $\boldsymbol{x} = (x_1, x_2, \ldots, x_n) \in \mathcal{B}_r(\mathbf{1})$ and $\mathbf{1} \neq \boldsymbol{y} = (y_1, y_2, \ldots, y_n) \in [n]^n$. Consider the mapping $\zeta : \mathcal{B}_r(\mathbf{1}) \to \mathcal{B}_r(\boldsymbol{y})$ where $\boldsymbol{x} \mapsto \boldsymbol{u}$, where $\boldsymbol{u} = (u_1, u_2, \ldots, u_n)$ is given by

$$u_i = \begin{cases} y_i + (x_i - 1) & \text{if } y_i + (x_i - 1) \leq n \\ n - (x_i - 1) & \text{if } y_i + (x_i - 1) > n. \end{cases}$$

Clearly $\boldsymbol{u} \in [n]^n$ and $x_i - 1 \geq |u_i - y_i|$ for $i = 1, \ldots, n$, so every point within distance r of $\mathbf{1}$ is sent to a point within distance r of $\boldsymbol{y}$. Furthermore, this mapping is injective because if $\boldsymbol{x}_1, \boldsymbol{x}_2$ are two distinct points in $\mathcal{B}_r(\mathbf{1})$ then their images can coincide only if in some coordinates

$$y_i + (x_{1,i} - 1) = n - (x_{2,i} - 1).$$

However, the left-hand side of this equality is $\geq y_i$ while the right-hand side is $< y_i$ by definition of u_i. This proves the right inequality.

To prove the lower bound, write $\mathcal{B}_r(\boldsymbol{z})$ as $\boldsymbol{z} + D_r(\boldsymbol{z})$, where $D_r(\boldsymbol{z})$ is the set of differences:

$$D_r(\boldsymbol{z}) = \{\boldsymbol{u} \in \mathbb{Z}^n : |u_i| \leq n-1, 1 \leq i \leq n;\ \sum_{i=1}^{n} |u_i| \leq r \text{ and } \boldsymbol{z}+\boldsymbol{u} \in [n]^n\}.$$

Writing $\mathcal{B}_r(\mathbf{1})$ in the same way as $\mathbf{1} + D_r^+$, we have

$$D_r^+ = \{\boldsymbol{u} \in \mathbb{Z}^n : 0 \leq u_i \leq n-1;\ \sum_{i=1}^{n} |u_i| \leq r\}.$$

By taking the absolute values of the coordinates, any point in $D_r(\boldsymbol{z})$ is sent to a point in D_r^+, and no more than 2^n points have the same image under this mapping. This proves our claim. □

These arguments give rise to the next proposition.

Proposition 7.3.6.

$$\frac{n!}{2^n \sum_{r=0}^{2d-1} Q(n,r)} \le X(n,d) \le \frac{n^n}{\sum_{r=0}^{t} Q(n,r)}, \tag{7.13}$$

where

$$Q(n,r) = \sum_{i \ge 0} (-1)^i K_{n,r}(i)$$

and $K_{n,r}(i) = \binom{n}{i}\binom{n+r-ni-1}{r-ni}$ *and* $t = \lfloor (d-1)/2 \rfloor$.

This claim is almost obvious because, by the previous lemma,

$$\frac{n!}{2^n |\mathcal{B}_{2d-1}(\mathbf{1})|} \le X(n,d) \le \frac{n^n}{|\mathcal{B}_t(\mathbf{1})|}$$

Next,

$$|\mathcal{B}_s(\mathbf{1})| = \sum_{r=0}^{s} Q(n,r),$$

where $Q(n,r)$ is the number of integer solutions of the equation

$$x_1 + x_2 + \ldots + x_n = r,$$

where $0 \le x_i \le n-1$, $1 \le i \le n$. The expression for $Q(n,r)$ given in the statement is well known (e.g., [53, p.1037]).

Expression (7.13) gives little insight into the behavior of the bound. In the remainder of this section we estimate the asymptotic behavior of this bound and derive an estimate of the code capacity.

Lemma 7.3.7. *Suppose that* $r < n^2/\ln n$. *Then*

$$\binom{n+r-1}{r} - n\binom{r-1}{r-n} \le Q(n,r) \le \binom{n+r-1}{r}.$$

Proof. Let $S(n,j) = \sum_{i \ge j} (-1)^i K_{n,r}(i)$. The lemma will follow if we prove that

$$S(n,1) < 0 \text{ and } S(n,2) > 0. \tag{7.14}$$

Under the assumption on r we have

$$\begin{aligned}\frac{\binom{r+n-n(i+1)-1}{r-n(i+1)}}{\binom{r+n-ni-1}{r-ni}} &= \prod_{j=1}^{n-1} \frac{r-ni-n+j}{r-ni+j} \\ &= \prod_{j=1}^{n-1} \Big(1 - \frac{n}{r-ni+j}\Big) \\ &\leq \Big(1 - \frac{n}{r-n(i-1)-1}\Big)^{n-1} \\ &\leq e^{-\frac{n(n-1)}{r-n(i-1)-1}} \\ &\leq n^{-\frac{n-1}{n}} \\ &\leq \frac{\sqrt{2}}{n}.\end{aligned}$$

Thus for $i \geq 1$

$$\frac{K_{n,r}(i+1)}{K_{n,r}(i)} \leq \frac{n-i}{i+1}\frac{\sqrt{2}}{n} < 1.$$

Therefore $-K_{n,r}(2m-1) + K_{n,r}(2m) < 0$ for all m. Since the sum $S(n,1)$ starts with a negative term and the sum $S(n,2)$ with a positive one, the required inequalities in (7.14) follow. □

From the foregoing arguments we now have the following explicit bounds on $X(n,d)$:

$$\frac{n!}{2^n\binom{n+2d-1}{2d-1}} \leq X(n,d) \leq \frac{n^n}{\sum_{r=0}^{t}\big(\binom{n+r-1}{r} - n\binom{r-1}{r-n}\big)}, \tag{7.15}$$

where $t = \lfloor (d-1)/2 \rfloor$. Here the right part is obvious and for the left inequality we used (7.13), Lemma 7.3.7, and the identity $\sum_{i\leq n}\binom{s+i}{i} = \binom{s+n+1}{n}$.

Now we are ready to complete the proof of Theorem 7.3.1. Assume that $d = \Theta(n^{1+\varepsilon})$ for some $0 \leq \varepsilon < 1$. The two boundary cases of (7.5) were established in the previous section. Let us prove the middle equality. From (7.15),

$$X(n,d) \leq \frac{n^n}{\binom{n+t-1}{n-1} - n\binom{t-1}{t-n}}$$

To estimate the denominator, write

$$\begin{aligned}\binom{n+t-1}{n-1} &= \binom{t-1}{t-n}\prod_{j=1}^{n-1}\Big(1+\frac{n}{t-j}\Big)\\ &> \binom{t-1}{t-n}\Big(1+\frac{n}{t}\Big)^{n-1}\\ &> n\binom{t-1}{t-n}e^{(n-1)(\frac{n}{t}-\frac{1}{2}(\frac{n}{t})^2)-\ln n}\\ &= n\binom{t-1}{t-n}e^{\Theta(n^{1-\varepsilon})}\end{aligned}$$

(because of $\ln(1+n/t) > (n/t) - \frac{1}{2}(n/t)^2$, for $n/t < 1$.) So starting with some n we can estimate the denominator below by $1/2\binom{n+t-1}{n-1}$. Therefore,

$$X(n,d) \le \frac{2n^n}{\binom{n+t-1}{t}} \le \frac{2n^n(n-1)^{n-1}}{(n+t-1)^{n-1}}.$$

Next

$$\frac{\ln X(n,\Theta(n^{1+\varepsilon}))}{n\ln n} \le 2-(1+\varepsilon)+o(1) = 1-\varepsilon+o(1).$$

On the other hand, using

$$\binom{n+2d-1}{2d-1} \le \Big(\frac{(n+2d)e}{n}\Big)^n < (2e)^n\Theta(n^{n\varepsilon})$$

and $n! > (n/3)^n$, we obtain from (7.15)

$$X(n,d) \ge \frac{n^n}{(12e)^n\Theta(n^{n\varepsilon})}.$$

Taking the logarithms and the limit, we find that $C(d) \ge 1-\varepsilon$. This completes the proof of Theorem 7.3.1.

7.3.4 Bounds from embedding in the Hamming space

Since the embedding of $\mathcal{X}_n$ into the Hamming space $\{0,1\}^N$ of dimension $N = \binom{n}{2}$ is isometric, the known results for codes correcting Hamming errors can be used to derive estimates and constructions for codes in the Kendall space. In particular, the known bounds on codes in the Hamming space can be rewritten with respect to the space $\mathcal{X}_n$. For instance, the Plotkin bound implies that

$$X(n,d) \le 2d/(2d-N)$$

and thus any code $\mathcal{C} \subset \mathcal{X}_n$ with distance greater than the average (i.e., $\frac{1}{2}N$) satisfies $|\mathcal{C}| = O(N)$.

Given the image of a code $\mathcal{C} \subset \mathcal{X}_n$ in $\{0,1\}^N$ it is easy to reconstruct the code $\mathcal{C}$ itself. Indeed, it is immediate to find the inversion vector of a permutation σ given the image of σ in $\{0,1\}^N$, and then to recover σ from its inversion vector.

Of course, not every code in $\{0,1\}^N$ will have a code in $\mathcal{X}_n$ corresponding to it. The next simple proposition shows that nevertheless, binary codes in $\{0,1\}^N$ can be used to claim existence of good rank permutation codes.

Proposition 7.3.8. *Suppose that there exists a binary linear* $[\binom{n}{2}, k, d]$ *code* $\mathcal{A}$. *Then there exists an* $(n, M \geq \frac{n!}{2^{N-k}}, d)$ *rank permutation code.*

Proof. One of the 2^{N-k} cosets of $\mathcal{A}$ in $\{0,1\}^N$ must contain at least $n!/2^{N-k}$ vectors that map back to valid permutations. □

For example, let us assume that the value N is such that there exists a t-error-correcting binary BCH code of length N (if not, we can add zeros to a shorter BCH code). Its dimension is at least $N - t\log_2(N+1)$. This shows the existence of a t-error-correcting rank permutation code of size $\frac{n!}{(N+1)^t} = \frac{n!}{O(n^{2t})}$.

On the other hand, by the sphere packing bound the size of a t-error-correcting code in $\mathcal{X}_n$ is at most $M \leq O(\frac{n!}{n^t})$. Thus, using the embedding $\mathcal{X}_n \to \{0,1\}^N$ we are not able to close a gap between the existence results and the upper bounds. In the next section we use a different method to construct codes that achieve the sphere packing bound to within a constant factor for any given t.

7.4 Towards optimal t-error-correcting codes

The representation of permutations by inversion vectors provides a way to construct error-correcting rank permutation codes. In this section we construct codes in the space of inversion vectors $\mathcal{G}_n$ equipped with ℓ_1 distance d_1, and claim the existence of rank permutation codes by the inequality on the code distances (7.3). Below $\|\boldsymbol{x}\|$ denotes the ℓ_1 norm of the vector $\boldsymbol{x}$.

We begin with constructing codes over the integers that correct additive errors. Once this is accomplished, we will be able to claim existence of good rank permutation codes. Let A be some subset of $\mathbb{Z}$ and let A^L be the space of L-tuples of integers from A equipped with the ℓ_1 distance (7.2); $L > 0$ is an integer. A code $\mathcal{D} \subset A^L$ is said to correct t additive errors if for any two distinct code vectors $\boldsymbol{x}, \boldsymbol{y}$ and any $\boldsymbol{e}_1, \boldsymbol{e}_2 \in \mathbb{Z}^L$, both of weight at most t (i.e., such that $\|\boldsymbol{e}_1\|, \|\boldsymbol{e}_2\| \leq t$),

$$\boldsymbol{x} + \boldsymbol{e}_1 \neq \boldsymbol{y} + \boldsymbol{e}_2.$$

Remark 6. If in the above definition $\boldsymbol{e}_i \geq 0$ for all i, the code is said to correct t *asymmetric* errors [40]. However below we need to consider the general case.

We assume that A and t are such that $\mathcal{D}$ is well defined: for instance, below we will take $A = [0, s-1]$ where s is some integer sufficiently large compared to t.

Definition 7.4.1. *Let $m \geq L$ and let $h_1, \ldots, h_L, 0 < h_i < m, i = 1, \ldots, L$ be a set of integers. Define the code $\mathcal{C}$ as follows:*

$$\mathcal{C} = \Big\{ \boldsymbol{x} \in A^L \Big| \sum_{i=1}^{L} h_i x_i \equiv 0 \mod m \Big\}. \tag{7.16}$$

This code construction was first proposed by Varshamov and Tenenholtz [102] for correction of one asymmetric error (it was rediscovered later by Constantin and Rao [40] and, in a slightly different context, by Golomb and Welch [54]). Generalizations to more that one error as well as to arbitrary finite groups were studied by Varshamov [101], Delsarte and Piret [44], and others; however, these works dealt with asymmetric errors. Below we extend this construction to the symmetric case.

Proposition 7.4.2. *The code $\mathcal{C}$ defined in (7.16) corrects t additive errors if and only if for all $\boldsymbol{e} \in \mathbb{Z}^L, \|\boldsymbol{e}\| \leq t$ the sums $\sum_{i=1}^{L} e_i h_i$ are all distinct and nonzero modulo m.*

This proposition is obvious as it amounts to saying that all the syndromes of error vectors of weight up to t are different and nonzero.

We will need the following theorem of Bose and Chowla [23]. In the following arguments q is a power of prime and $m = (q^{t+1} - 1)/(q - 1)$.

Theorem 7.4.3. *(Bose and Chowla) There exist $q + 1$ integers $j_0 = 0, j_1, \ldots, j_q$ in $\mathbb{Z}_m$ such that the sums*

$$j_{i_1} + j_{i_2} + \ldots + j_{i_t} \quad (0 \leq i_1 \leq i_2 \leq \ldots \leq i_t \leq q)$$

are all different modulo m.

This theorem provides a way of constructing an asymmetric t additive error-correcting code of length q. This is because for any error vector $\boldsymbol{e}$ with $||\boldsymbol{e}|| \leq t < m$ such that $e_i \geq 0$, the sums $\sum_{i=1}^{q} e_i j_i$ involve at most t of the numbers j_i and thus are all different. In coding theory, this theorem was previously used to construct constant weight codes in the Hamming space [45, 56].

We extend the Theorem 7.4.3 so that one can construct a general t additive error-correcting code. In the following discussions, let $m_t = t(t + 1)m$ if t is odd and $m_t = t(t + 2)m$ if t is even.

Theorem 7.4.4. *For $1 \leq i \leq q + 1$ let*

$$h_i = \begin{cases} j_{i-1} + \frac{t-1}{2} m & \text{for } t \text{ odd} \\ j_{i-1} + \frac{t}{2} m & \text{for } t \text{ even} \end{cases}$$

where the numbers j_i are given by the Bose-Chowla theorem. For all $\boldsymbol{e} \in \mathbb{Z}^{q+1}$ such that $||\boldsymbol{e}|| \leq t$ the sums $\sum_{i=1}^{q+1} e_i h_i$ are all distinct and nonzero modulo m_t.

Proof. Let t be odd and let $H_q = \{0, h_1, \ldots, h_{q+1}\}$. Observe that

$$(t-1)m/2 \leq h_i < (t+1)m/2. \tag{7.17}$$

(i) For any $k_1 \leq k_2 \leq \ldots \leq k_t \in H_q$, the sums $\sum_{i=1}^{t} k_i$ are all distinct modulo m and therefore also modulo m_t. These sums are also nonzero modulo m except for the case when all the k_i's are 0.

(ii) Moreover, for any $k_1 \leq k_2 \ldots \leq k_{2t} \in H_q$, the sum

$$\sum_{i=1}^{2t} k_i < m_t,$$

and is therefore nonzero modulo m_t.

(iii) Finally, for any $0 < k_1, k_2, \ldots, k_{2t} \in H_q$ and any $r < t$,

$$\begin{aligned} \sum_{i=2t-r+1}^{2t} k_i &< r\frac{t+1}{2}m \leq (2t-r)\frac{t-1}{2}m \\ &\leq \sum_{i=1}^{2t-r} k_i. \end{aligned} \tag{7.18}$$

Let us suppose now that there exist nonzero vectors $\boldsymbol{e}_1, \boldsymbol{e}_2 \in \mathbb{Z}^{q+1}$ both of weight at most t such that

$$\begin{aligned} \text{either (a)} \quad & \textstyle\sum_{i=1}^{q+1} e_{1i}h_i = 0 \mod m_t \\ \text{or (b)} \quad & \textstyle\sum_{i=1}^{q+1} e_{1i}h_i = \sum_{i=1}^{q+1} e_{2i}h_i \mod m_t. \end{aligned}$$

However assuming (a) contradicts property (i). On the other hand if (b) is true then one of the following two scenarios can happen. In the first case, $e_{1i} \geq 0$ and $e_{2i} \leq 0$ for all i or $e_{1i} \leq 0$ and $e_{2i} \geq 0$ for all i. It is easy to see that this assumption contradicts property (ii). In all other situations, (b) contradicts either property (i) or property (iii) above.

The claim for t even is proved in an analogous way. Namely, we will have

$$tm/2 \leq h_i \leq (t+2)m/2$$

and

$$\sum_{i=2t-r+1}^{2t} k_i < r\Big(\frac{t+2}{2}\Big)m \leq (2t-r)\frac{tm}{2} \leq \sum_{i=1}^{2t-r} k_i$$

instead of (7.17) and (7.18), respectively. The rest of the proof remains the same. □

Together with Proposition 7.4.2 this theorem implies the existence of a t-error-correcting code $\mathcal{C}$ of length $q+1$ over the alphabet $A = \mathbb{Z}_{m_t}$ that corrects t additive errors. Recall that our goal is to construct a code over the set of inversion vectors $\mathcal{G}_n$ that corrects t additive errors. At this point let us set $q + 1 = n - 1$. Note that, $\mathcal{G}_n$ is a subset of $[0, n-1]^{n-1}$ which is a subset of $\mathbb{Z}_{m_t}^{n-1}$. Since $\mathcal{C}$ is a group code with respect to addition modulo m_t, its cosets in $\mathbb{Z}_{m_t}^{n-1}$ partition this space into disjoint equal parts. At least one such coset contains $M \geq n!/m_t$ vectors from $\mathcal{G}_n$. Invoking (7.3) we now establish the main result of this section.

Theorem 7.4.5. *Let $m = ((n-2)^{t+1} - 1)/(n-3)$, where $n-2$ is a power of a prime. There exists a t-error-correcting rank permutation code in $\mathfrak{S}_n$ whose size M satisfies*

$$M \geq \begin{cases} n!/(t(t+1)m) & (t \textit{ odd}) \\ n!/(t(t+2)m) & (t \textit{ even}). \end{cases}$$

This theorem establishes the existence of codes whose size is of the same order $O(n!/n^t)$ as given by the sphere packing bound of the previous section.

As a final remark, note that the construction is explicit except for the last step where we claim existence of a large-size code in some coset of the code $\mathcal{C}$.

CHAPTER 8

Codes in Permutations: Constructions

8.1 Introduction

In the previous chapter we established the asymptotic scaling of the rate of optimal codes in the Kendall space $\mathcal{X}_n$. However, the main question related to the applications of the rank modulation scheme relates to explicit coding schemes for error correction. Codes correcting one Kendall error were constructed in [68]. In the previous chapter we proved the existence of a family of rank permutation codes that correct a constant number of errors and have size within a constant factor of the sphere packing bound. The major gap in the literature on coding for rank modulation has been the absence of explicit constructions of families of rank modulation codes. Addressing this issue, in the present chapter we provide few general constructions of rank modulation codes that correct errors of multiplicity varying over a large range of values.

The results of this chapter appear in [15].

8.2 Rank modulation codes and permutation polynomials

Our first construction of rank modulation codes is algebraic in nature. We identify the permutations on the elements of a field with the permutation polynomials over the field.

Let $q = p^m$ for some prime p and let $\mathbb{F}_q = (\alpha_0, \alpha_1, \ldots, \alpha_{q-1})$ be the finite field of q elements. A polynomial $g(x) \in \mathbb{F}_q[x]$ is called a *permutation polynomial* if it permutes the elements of $\mathbb{F}_q$ (this means that the values $g(a)$ are distinct for distinct values of $a \in \mathbb{F}_q$) [76, Ch. 7].

Consider the evaluation map $f \mapsto (f(\alpha_0), \ldots, f(\alpha_{q-1}))$ which sends permutation polynomials to permutations of q elements. Evaluations of permutation polynomials of degree $\leq k$ form a subset of a q-ary Reed-Solomon code of dimension $k+1$. Reed-Solomon codes form a family of error-correcting codes in the Hamming space with a number of desirable properties including efficient decoding. For an introduction to them see [80, Ch. 10].

At the same time, evaluating the size of a rank permutation code constructed in this way is a difficult problem because it is hard to compute the number of permutation polynomials of a given degree. In this section we formalize a strategy of constructing codes along these lines. This does not result in a very good rank modulation code; in fact, our later combinatorial constructions will be much better, in terms of the size of the codes with given error-correcting capabilities. Nonetheless, the construction involves some interesting observations which is why we decided to include it.

A polynomial over $\mathbb{F}_q$ is called *linearized* of degree ν if it has the form

$$\mathcal{L}(x) = \sum_{i=0}^{\nu} a_i x^{p^i}$$

Note that a linearized polynomial of degree ν has degree p^ν when viewed as a standard polynomial.

Lemma 8.2.1. *The number of linearized polynomials over $\mathbb{F}_q$ of degree less than or equal to ν that are permutation polynomials in $\mathbb{F}_q$ is at least*

$$\Big(1 - \frac{1}{p-1} + \frac{1}{q(p-1)}\Big) q^{\nu+1} \geq q^\nu.$$

Proof. The polynomial $\mathcal{L}(x)$ acts on $\mathbb{F}_q$ as a linear homomorphism. It is injective if and only if it has a trivial kernel, in other words if the only root of $\mathcal{L}(x)$ in $\mathbb{F}_q$ is 0. Hence, $\mathcal{L}(x)$ is a permutation polynomial if and only if the only root of $\mathcal{L}(x)$ in $\mathbb{F}_q$ is 0.

The total number of linearized polynomials of degree up to ν is $q^{\nu+1}$. We are going to prove that at least a $(1 - \frac{1}{p-1} + \frac{1}{q(p-1)})$ proportion of them are permutation polynomials. To show this, choose the coefficients $a_i, 0 \leq i \leq \nu$, of $\mathcal{L}(x) = \sum_{i=0}^{\nu} a_i x^{p^i}$ uniformly and randomly from $\mathbb{F}_q$. For a fixed $\alpha \in \mathbb{F}_q^*$, the probability that $\mathcal{L}(\alpha) = 0$ is $1/q$. Furthermore, the set of roots of a linearized polynomial is an $\mathbb{F}_p$-vector space, hence the set of non-zero roots is a multiple of $p-1$. The number of 1-dimensional subspaces of $\mathbb{F}_q$ over $\mathbb{F}_p$ is $\frac{q-1}{p-1}$. The probability that one of these sets is included in the set of roots of $\mathcal{L}(x)$ is, from the union bound,

$$\Pr(\exists \alpha \in \mathbb{F}_q^* : \mathcal{L}(\alpha) = 0) \leq \frac{q-1}{p-1} \cdot \frac{1}{q}.$$

Hence, the probability that $\mathcal{L}(x)$ is a permutation polynomial is greater than or equal to $1 - \frac{q-1}{q(p-1)}$. □

8.2.1 Code construction

We can either take $n = q$ or $n = q - 1$ where q is the size of a field and we construct a rank modulation code in $\mathfrak{S}_n$. Note that a linearized polynomial $\mathcal{L}(x)$

always maps zero to zero, so that when it is a permutation polynomial it can be considered to be a permutation of the elements of $\mathbb{F}_q$ and also of the elements of $\mathbb{F}_q^*$. Let t be a positive integer and let $\nu = \lfloor \log_p(n-2t-1) \rfloor$. Let $\mathcal{P}_t$ be the set of all linearized polynomials of degree $\leq \nu$ that permute $\mathbb{F}_q$. Set $n = q-1$ and define the set $A \subset \mathbb{F}_q^n$

$$A = \{(\mathcal{L}(a), a \in \mathbb{F}_q^*),\ \mathcal{L} \in \mathcal{P}_t\}$$

to be the set of vectors obtained by evaluating the polynomials in $\mathcal{P}_t$ at the points of $\mathbb{F}_q^*$. Form a code $\mathcal{C}$ by writing the vectors in A as permutations (for that, we fix some bijection between $[n]$ and $\mathbb{F}_q^*$, which will be implicit in the subsequent discussion). We can have $n = q$ rather than $n = q-1$ if desired: for that we add the zero field element in the first position of the $(q-1)$-tuples of A, and the construction below readily extends.

The idea behind the construction is quite simple: the set A is a subset of a Reed-Solomon code that corrects t Hamming errors. Every Kendall error is a transposition, and as such, affects at most two coordinates of the codeword of $\mathcal{C}$. Therefore the code $\mathcal{C}$ can correct up to $t/2$ errors. By handling Kendall errors more carefully, we can actually correct up to t errors. The main result of this part of our work is given by the following statement.

Theorem 8.2.2. *The code $\mathcal{C}$ has length $n = q-1$ and size at least $q^{\lfloor \log_p(n-2t-1) \rfloor}$. It corrects all patterns of up to t Kendall errors in the rank modulation scheme under a decoding algorithm of complexity polynomial in n.*

Proof. It is clear that $|\mathcal{C}| = |A|$, and from Lemma 8.2.1 $|A| \geq q^{\lfloor \log_p(n-2t-1) \rfloor}$.

Let $\sigma = (a_1, a_2, \ldots, a_i, a_{i+1}, \ldots, a_n)$, where $a_j \in \mathbb{F}_q^*, 1 \leq j \leq n$, be a permutation in $\mathcal{X}_n$ (with the implied bijection between $[n]$ and $\mathbb{F}_q^*$) and let $\sigma' = (a_1, a_2, \ldots, a_{i+1}, a_i, \ldots, a_n)$ be a permutation obtained from σ by one Kendall step (an adjacent transposition). We have

$$\sigma - \sigma' = (0, \ldots, 0, \theta, -\theta, \ldots, 0)$$

where $\theta = a_i - a_{i+1} \in \mathbb{F}_q^*$.

Let

$$P = \begin{pmatrix} 1 & 0 & 0 & \cdot & \cdot & 0 \\ 1 & 1 & 0 & \cdot & \cdot & 0 \\ 1 & 1 & 1 & \cdot & \cdot & 0 \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 1 & 1 & 1 & \cdot & \cdot & 1 \end{pmatrix}$$

be an $n \times n$ matrix. Note that

$$P(\sigma - \sigma')^T = (0, \ldots, 0, \theta, 0, \ldots, 0)^T.$$

This means that multiplication by the accumulator matrix P converts one adjacent transposition error into one Hamming error. Extending this observation, we claim that if $\mathrm{d}_\tau(\sigma,\pi) \le t$ with π being some permutation, and any $t \le \frac{n}{2}$, then the Hamming weight of the vector $P(\sigma-\pi)^T$ is not more than t. Here we again take σ and π to be vectors with elements from $\mathbb{F}_q^*$ with the implied bijection between $[n]$ and $\mathbb{F}_q^*$.

Now let $\mathcal{L}(x)$ be a linearized permutation polynomial and let $1, \alpha, \alpha^2, \ldots, \alpha^{q-2}$ be the elements of $\mathbb{F}_q^*$ for some choice of the primitive element α. Let

$$\sigma = (\mathcal{L}(1), \mathcal{L}(\alpha), \mathcal{L}(\alpha^2), \ldots, \mathcal{L}(\alpha^{q-2})).$$

Since $\mathcal{L}(a+b) = \mathcal{L}(a) + \mathcal{L}(b)$, we have

$$P\sigma^T = (\mathcal{L}(\beta_0), \mathcal{L}(\beta_1), \mathcal{L}(\beta_2), \ldots, \mathcal{L}(\beta_{q-2}))^T$$

where

$$\beta_i = \sum_{j=0}^{i} \alpha^j, \quad i = 0, 1, \ldots, q-2.$$

It is clear that $\beta_i \neq 0$, $0 \le i \le n-1$ and also $\beta_i \neq \beta_j$ for $0 \le i < j \le n-1$, Therefore, the vector $P\sigma^T$ is a permutation of the elements of $\mathbb{F}_q^*$. At the same time, it is the evaluation vector of a polynomial of degree $\le n-2t-1$. We conclude that the set $\{P\sigma^T, \sigma \in A\}$ is a subset of vectors of an (extended) Reed-Solomon code of length n, dimension $n-2t$ and distance $2t+1$. Any t errors in a codeword of such a code can be corrected by standard RS decoding algorithms in polynomial time.

The following scheme for writing data with the code $\mathcal{C}$ corrects any t Kendall errors. Suppose $\sigma \in A$ is read off from memory as σ_1. Evaluate $\boldsymbol{z} = P\sigma_1^T$, and use a Reed-Solomon decoding algorithm to correct up to t Hamming errors in the vector $\boldsymbol{z}$, obtaining a vector $\boldsymbol{y}$. If $\mathrm{d}_\tau(\sigma,\sigma_1) \le t$, then $\boldsymbol{y}$ corresponds to a transformed version of σ, i.e., $\boldsymbol{y} = P\sigma^T$. So σ is recovered as $P^{-1}\boldsymbol{y}^T$, i.e.,

$$\sigma_i = y_{i+1} - y_i, \;\; 1 \le i \le n-1; \;\; \sigma_n = y_n.$$

□

We note an earlier use of permutation polynomials for constructing permutation codes in [37]. At the same time, since the coding problem considered in that paper relies on the Hamming metric rather than the Kendall tau distance, its results have no immediate link to the above construction.

8.3 Rank modulation codes from codes in the Hamming space

In this section we present other ideas for constructing rank permutation codes using the weight-preserving embedding of the Kendall space $\mathcal{X}_n$ into a subset of integer vectors discussed in Sect. 7.2. To evaluate the error-correcting capability of the resulting codes, we further link codes over integers with codes correcting Hamming errors.

8.3.1 From inversion vectors to the Hamming space via Gray map

Recall that the mapping from permutations to the space of inversion vectors is one-to-one, and any permutation can be easily reconstructed from its inversion vector with the map J defined in Sect. 7.2.

We will need the *Gray map* which is a mapping ϕ_s from the ordered set of integers $[0, 2^s - 1]$ to $\{0,1\}^s$ with the property that the images of two successive integers differ in exactly one bit. Suppose that $b_{s-1}b_{s-2}\dots b_0$, $b_i \in \{0,1\}, 0 \le i < s$, is the binary representation of an integer $u \in [0, 2^s - 1]$. Set by definition $b_s = 0$ and define $\phi_s(u) = (g_{s-1}, g_{s-2}, \dots, g_0)$, where

$$g_j = (b_j + b_{j+1}) \pmod 2 \qquad (j = 0, 2, \dots s-1) \tag{8.1}$$

(note that generally the Gray map is not uniquely defined for $s \ge 4$).

Now let $i = 2, \dots, n$,

$$m_i = \lfloor \log i \rfloor,$$

and let

$$\psi_i : \{0,1\}^{m_i} \to [0, i-1]$$

be the inverse Gray map $\psi_i = \phi_i^{-1}$. Clearly ψ_i is well defined; it is injective but not onto since the size of its domain is only 2^{m_i}.

Proposition 8.3.1. *Suppose that $\boldsymbol{x}, \boldsymbol{y} \in \{0,1\}^{m_i}$. Then*

$$|\psi_i(\boldsymbol{x}) - \psi_i(\boldsymbol{y})| \ge \mathrm{d}(\boldsymbol{x}, \boldsymbol{y}),$$

where d *denotes the Hamming distance.*

Proof. This follows from the fact that if the difference of magnitude is 1 between the numbers, then their Gray images have Hamming distance 1 between them. If two numbers $u < v$ are such that $|u - v| = d$, then one can obtain the ordered set of $d + 1$ numbers $u, u + 1, u + 2, \dots, v = u + d$. Hence, from the triangle inequality the Hamming distance between the Gray images of u and v is less than or equal to d. □

Consider a vector $\boldsymbol{x} = (\boldsymbol{x}_2|\boldsymbol{x}_3|\dots|\boldsymbol{x}_n)$, where $\boldsymbol{x}_i \in \{0,1\}^{m_i}$, $i = 2,\dots,n$. The dimension m of $\boldsymbol{x}$ equals $\sum_j m_j \approx \log n!$, or more precisely

$$\begin{aligned} m &= \sum_{j=1}^{m_n-1} (2^{j+1}-2^j)j + m_n(n+1-2^{m_n}) \\ &= \sum_{j=1}^{m_n-1} j2^j + m_n(n+1-2^{m_n}) \\ &= (m_n-2)2^{m_n} + 2 + m_n(n+1-2^{m_n}) \\ &= (n+1)m_n - 2^{m_n+1} + 2. \end{aligned}$$

Given a vector $\boldsymbol{x} \in \{0,1\}^m$ let

$$\Psi(\boldsymbol{x}) = \Psi(\boldsymbol{x}_2|\boldsymbol{x}_3|\dots|\boldsymbol{x}_n) = (\psi_2(\boldsymbol{x}_2),\dots,\psi_n(\boldsymbol{x}_n)).$$

Proposition 8.3.2. *Let $\boldsymbol{x},\boldsymbol{y} \in \{0,1\}^m$. Then*

$$\mathrm{d}_1(\Psi(\boldsymbol{x}),\Psi(\boldsymbol{y})) \geq \mathrm{d}(\boldsymbol{x},\boldsymbol{y}),$$

where the distance d_1 is the ℓ_1 distance defined in (7.2) and d is the Hamming distance.

Proof.

$$\begin{aligned} \mathrm{d}_1(\Psi(\boldsymbol{x}),\Psi(\boldsymbol{y})) &= \sum_{i=2}^{n} |\psi_i(\boldsymbol{x}_i) - \psi_i(\boldsymbol{y}_i)| \\ &\geq \sum_{i=2}^{n} \mathrm{d}(\boldsymbol{x}_i,\boldsymbol{y}_i) \\ &= \mathrm{d}(\boldsymbol{x},\boldsymbol{y}). \end{aligned}$$

□

Now we can formulate a general method to construct rank permutation codes: take a binary code of length m and cardinality M in the Hamming space and send each of its vectors to a permutation using the composition map $J \circ \Psi$. Both parts of this map are injective, so the cardinality of the resulting code is M. Moreover, each of the two mappings can only increase the distance (namely, see (7.3) and the above proposition). Summarizing, we have

Theorem 8.3.3. *Let $\mathcal{C}$ be a binary (m,M,d) code, where $m = (n+1)\lfloor \log n \rfloor - 2^{\lfloor \log n \rfloor + 1} + 2$. Then the set of permutations*

$$\mathcal{C}_\tau = \left\{\pi \in \mathfrak{S}_n : \pi = J(\Psi(\boldsymbol{x})), \boldsymbol{x} \in \mathcal{C}\right\}$$

forms an rank modulation code on n elements of size M and distance at least d in the Kendall space $\mathcal{X}_n$.

Example: Consider a t-error-correcting (shortened) BCH code of length $m = (n+1)\lfloor \log n \rfloor - 2^{\lfloor \log n \rfloor + 1} + 2$ and designed distance $2t+1$. If m is one less than a power of two, then the size of the code is

$$M \geq \frac{2^m}{(m+1)^t}.$$

This shows that we can construct a set of (n, M) rank modulation codes that correct t errors. Note that for constant t any code $\mathcal{C}$ in $\mathcal{X}_n$ satisfies $|\mathcal{C}| \leq O(n!/n^t)$. The rank modulation codes constructed from binary BCH codes have size $M = \Omega(n!/\log^t n!) = \Omega(n!/n^t \log^t n)$.

For instance, take $n = 62$, then $m = 253$. Taking twice shortened BCH codes $\mathcal{C}_t$ of length m, we obtain a range of rank modulation codes according to the designed distance of $\mathcal{C}_t$. In particular, there are rank permutation codes in $\mathcal{X}_{62}$ with distance at least $2t+1$ and size M given by:

$\log M$	247	239	231	223 ...
t	1	2	3	4 ...

Similarly, taking $n = 105$, we can construct a suite of rank permutation codes from shortened BCH codes of length $m = 510$, etc.

Consider now the case when the number of errors t grows with n. Since the binary codes constructed above are of length $n \log n$, we can obtain rank permutation codes in $\mathcal{X}_n$ that correct error patterns of Kendall weight $t = \Omega(n \log n)$. But in fact more is true. We need the following proposition.

Proposition 8.3.4. *Let $\boldsymbol{x}, \boldsymbol{y} \in \{0,1\}^m$. Then*

$$\mathrm{d}_1(\Psi(\boldsymbol{x}), \Psi(\boldsymbol{y})) \geq \frac{n-1}{2}\Big(2^{\frac{\mathrm{d}(\boldsymbol{x},\boldsymbol{y})}{n-1}} - 1\Big).$$

Proof. We first claim that, for any $\boldsymbol{x}, \boldsymbol{y} \in \{0,1\}^{m_i}$, the inequality $\mathrm{d}(\boldsymbol{x}, \boldsymbol{y}) \geq w_i \geq 1$ implies that $|\psi_i(x) - \psi_i(y)| \geq 2^{w_i - 1}$. This is true because of the 'reflective' nature of the standard Gray map as is evident from Eq. (8.1).

Now consider vectors $\boldsymbol{x} = (\boldsymbol{x}_2|\boldsymbol{x}_3|\ldots|\boldsymbol{x}_n), \boldsymbol{y} = (\boldsymbol{y}_2|\boldsymbol{y}_3|\ldots|\boldsymbol{y}_n)$ in $\{0,1\}^m$ where $\boldsymbol{x}_i, \boldsymbol{y}_i \in \{0,1\}^{m_i}, 2 \leq i \leq n$. Suppose that $\mathrm{d}(\boldsymbol{x}_i, \boldsymbol{y}_i) = w_i$ for all i, and $\sum_{i=2}^n w_i = w$ where $w = \mathrm{d}(\boldsymbol{x}, \boldsymbol{y})$.

Hence,

$$\begin{aligned}
\mathrm{d}_1(\Psi(\boldsymbol{x}),\Psi(\boldsymbol{y})) &= \sum_{i=2}^{n} |\psi_i(\boldsymbol{x}_i) - \psi_i(\boldsymbol{y}_i)| \\
&\geq \sum_{i:\, w_i>0} 2^{w_i-1} \\
&= \sum_{i=2}^{n} 2^{w_i-1} - \sum_{i:\, w_i=0} \frac{1}{2} \\
&\geq \frac{1}{2}\Big(\min_{w_i>0,\, \sum_{i=2}^{n} w_i = w} \quad \sum_{i=2}^{n} 2^{w_i} - \sum_{i: w_i=0} 1 \Big) \\
&\geq \frac{n-1}{2}\big(2^{\frac{w}{n-1}} - 1\big).
\end{aligned}$$

□

We have the following theorem as a result.

Theorem 8.3.5. *Let $\mathcal{C}$ and $\mathcal{C}_\tau$ be the binary and rank permutation codes defined in Theorem 8.3.3. Suppose furthermore that the minimum Hamming distance d of the code $\mathcal{C}$ satisfies $d \geq \varepsilon m$ where m is the blocklength of $\mathcal{C}$. Then the minimum Kendall distance of the code $\mathcal{C}_\tau$ is $\Omega(n^{1+\varepsilon})$.*

Proof. Observe that $m \geq n(\log n - 3)$ so $d \geq \varepsilon m \geq \varepsilon n(\log n - 3)$. From the previous proposition the minimum distance of $\mathcal{C}_\tau$ is at least

$$\frac{n-1}{2}\Big(2^{\varepsilon n(\log n - 3)} - 1\Big) = \Omega(n^{1+\varepsilon}).$$

□

From the existing asymptotically good families of binary codes with rate $R > 0$ and relative distance $0 < \varepsilon < 1/2$, one can therefore construct rank permutation codes of distance $\Omega(n^{1+\varepsilon})$ and rate R (see (7.4)). The upper limit of $1/2$ on ε is due to the fact that no binary codes of large size (positive rate) are capable of correcting a higher proportion of errors.

The above theorem can be extended to the case when $\varepsilon \geq 1/2$, namely, to obtain rank permutation codes of distance $\Omega(n^{1+\varepsilon})$, $1/2 \leq \varepsilon < 1$ and positive rate. This extension is not direct, and results in an existential claim as opposed to the constructive results above. To be precise, one can show that for any $0 \leq \varepsilon < 1$, there exist infinite families of binary (m, M, d) codes $\mathcal{C}$, with rate $R > 0$, such that the associated rank modulation code $\mathcal{C}_\tau$ for permutations of $[n]$ in Theorem 8.3.3 has minimum Kendall distance $\Omega(n^{1+\varepsilon})$. We will not prove this result here. Instead, in the next section we will present another construction of rank modulation codes that is effective in this range of distance.

8.3.2 Another construction: A quantization map

In this section we describe another construction of rank modulation codes that relies on a different mapping from binary vectors to inversions.

Recall again our notation $\mathcal{G}_n$ for the space of inversion vectors and the map $J : \mathcal{G}_n \to \mathfrak{S}_n$ that sends them to permutations (see Sect. 7.2). To obtain a code in $\mathcal{G}_n$ we will start with a set of binary vectors $\mathcal{C} \in \{0,1\}^n$ and send them to inversion vectors. This is done using the mapping $\vartheta : \{0,1\}^n \to \mathcal{G}_n$ such that

$$\boldsymbol{b} = (b_1, \ldots, b_{n-1}) \stackrel{\vartheta}{\mapsto} \boldsymbol{x} = (x_1, \ldots, x_{n-1})$$
$$x_i = \begin{cases} 0 & \text{if } b_i = 0 \\ i & \text{if } b_i = 1 \end{cases}, \quad i = 1, \ldots, n-1.$$

Next, the obtained subset of $\mathcal{G}_n$ is mapped by J to a subset of $\mathfrak{S}_n$, which we denote by $\mathcal{C}_\tau$.

Theorem 8.3.6. *In the above construction let $\mathcal{C}(n, M, d \geq 2t+1)$ be a code in the binary Hamming space. Then the code $\mathcal{C}_\tau \subset \mathfrak{S}_n$ has cardinality M and corrects any r Kendall errors where $r = t^2/4$ if t is even and $r = (t^2-1)/4$ if t is odd.*

Proof. To prove the claim about error correction, consider the following decoding procedure of the code $\mathcal{C}_\tau$. Let π be a permutation. To decode it with $\mathcal{C}_\tau$, find its inversion vector $\boldsymbol{x}_\pi = (x_1, \ldots, x_{n-1})$ and form a binary vector $\boldsymbol{y}$ by putting

$$y_i = \begin{cases} 0 & \text{if } x_i \leq \lfloor i/2 \rfloor \\ 1 & \text{if } x_i > \lfloor i/2 \rfloor. \end{cases}$$

Next decode $\boldsymbol{y}$ with the code $\mathcal{C}$ to obtain a codevector $\boldsymbol{c}$. Then compute the overall decoding result as $J(\vartheta(\boldsymbol{c}))$.

Let σ be the original permutation, let $\boldsymbol{x}_\sigma$ be its inversion vector, and let $\boldsymbol{c}(\sigma)$ be the corresponding codeword of $\mathcal{C}$. The above decoding can go wrong only if the Hamming distance $\mathrm{d}(\boldsymbol{c}(\sigma), \boldsymbol{y}) \geq t$. For this to happen the ℓ_1 distance between $\boldsymbol{x}_\pi$ and $\boldsymbol{x}_\sigma$ must be large, in the worst case satisfying the condition $\mathrm{d}_1(\boldsymbol{x}_\pi, \boldsymbol{x}_\sigma) \geq \sum_{i=1}^{t} \lfloor i/2 \rfloor$. This gives the claimed result. □

From a binary code in Hamming space of rate R that corrects any τn errors, the above construction produces a rank modulation code of size 2^{Rn} that is able to correct $\Omega(n^2)$ errors.

This construction can be further generalized to construct codes that are able to correct a wide range of Kendall errors by observing that the quantization map employed above is a rather coarse tool which can be refined if we rely on codes in the q-ary Hamming space for $q > 2$. As a result, for any $\varepsilon < 1$ we will be able to construct families of rank permutation codes of rate $R = R(\varepsilon) > 0$ (see, (7.4)) that correct $\Omega(n^{1+\varepsilon})$ errors.

Let $l > 0$ be an integer. Let $Q = \{a_1, a_2, \ldots, a_q\}$ be the code alphabet. Consider a code $\mathcal{C}$ of length $n' = 2(l-1)(q-1)$ over Q and assume that it corrects any t Hamming errors (i.e., its minimum Hamming distance is at least $2t+1$). Let $n = (2l+1)(q-1)$. Consider the mapping $\Theta_q : Q^{n-1} \to \mathcal{G}_n$, defined as $\Theta_q(\boldsymbol{b}) = (\vartheta_1(b_1), \vartheta_2(b_2), \ldots, \vartheta_{n-1}(b_{n-1}))$, $\boldsymbol{b} = (b_1, \ldots, b_{n-1}) \in Q^{n-1}$, where

$$\vartheta_i(a_j) = \begin{cases} 0 & \text{if } i < 3(q-1) \\ (2k-1)(j-1) & \text{if } (2k-1)(q-1) \le i < (2k+1)(q-1) \\ & \qquad k = 2, 3, \ldots, l, \end{cases}$$
$$j = 1, 2, 3, \ldots, q.$$

To construct a code in permutations $\mathcal{C}_\tau$ from the code $\mathcal{C}$ we perform the following steps:

1. Prepend each vector in $\mathcal{C}$ with $3(q-1)-1$ symbols a_1;
2. Map the obtained set of $(n-1)$-dimensional vectors to $\mathfrak{S}_n$ using the map $J \circ \Theta_q$.

The properties of this construction are summarized in the following statement.

Theorem 8.3.7. *In the above construction let $\mathcal{C}(n', M, d \ge 2t+1)$ be a code in the binary Hamming space. Then the code $\mathcal{C}_\tau \subset \mathfrak{S}_n$ has cardinality M and corrects any r Kendall errors where $r = (t+1-(q-1)s)(s+1) - 1$ and $s = \lfloor (t+1)/(2(q-1)) \rfloor$.*

Proof. We generalize the proof of the previous theorem. Consider the following decoding procedure of the code $\mathcal{C}_\tau$. Let π be a permutation. To decode it with $\mathcal{C}_\tau$, we first find its inversion vector $\boldsymbol{x}_\pi = (x_1, \ldots, x_{n-1})$ and form a q-ary vector $\boldsymbol{y}$ by putting

$$y_i = \begin{cases} a_1 & \text{if } i < 3(q-1) \\ a_j & \text{if } (2k-1)(q-1) \le i < (2k+1)(q-1) \\ & \qquad \text{and } (2k-1)(j-1) - (k-1) \le x_i \le (2k-1)(j-1) + k, \\ & \qquad k = 2, 3 \ldots, l \end{cases}$$

for $i = 1, \ldots, n-1$. Next decode $\boldsymbol{y}' = (y_{3(q-1)}, \ldots, y_{n-1})$ with the code $\mathcal{C}$ to obtain a codevector $\boldsymbol{c}$. The vector $\boldsymbol{c}$ is used to obtain the end result (a permutation) using the mapping defined before the theorem.

There will be an error in decoding only when $\boldsymbol{y}'$ contains at least $t+1$ Hamming errors. $\boldsymbol{y}'$ contains coordinates $3(q-1)$ to $n-1$ of $\boldsymbol{y}$. Suppose that $t_j, 1 \le j \le l-1$ is the number of errors in coordinates between $(2j+1)(q-1)$ and $(2j+3)(q-1)$. We have $\sum_{j=1}^{l-1} t_j \ge t+1$ and $t_j \le 2(q-1)$. Here the ℓ_1 distance between the

received and original inversion vectors is

$$\begin{aligned}\sum_{j=1}^{l-1} jt_j &\geq \min_{\substack{t_j \leq 2(q-1) \\ \sum_j t_j \geq t+1}} \sum_{j=1}^{l-1} jt_j \\ &= 2(q-1)(1+2+\cdots+s)+(t+1-2(q-1)s)(s+1) \\ &= (q-1)s(s+1)+(t+1-2(q-1)s)(s+1) \\ &= (t+1-(q-1)s)(s+1).\end{aligned}$$

Therefore if the ℓ_1 distance between the received and original inversion vectors is less than or equal to r then decoding $\boldsymbol{y}'$ with the code $\mathcal{C}$ will recover $\boldsymbol{x}_\sigma$. Using (7.3) we complete the proof. □

Choosing $t = 2(l-1)(q-1)\tau$, where $0 \leq \tau \leq 1/4$, the number of errors correctable by $\mathcal{C}_\tau$ is

$$\begin{aligned}r &= (2(l-1)(q-1)\tau - (q-1)\lfloor \tau(l-1) \rfloor)(\lfloor \tau(l-1) \rfloor + 1) - 1 \\ &\approx \tau^2(l-1)^2(q-1) \\ &\approx \frac{\tau^2 n^2}{q}.\end{aligned}$$

For instance, take $q = O(n^{1-\varepsilon})$, $0 < \varepsilon < 1$, then $r = \Omega(n^{1+\varepsilon})$. If the code $\mathcal{C}$ has cardinality $q^{Rn'}$ then $|\mathcal{C}_\tau| = q^{Rn'} = q^{R(n-3(q-1))}$. Using (7.4) yields the value $(1-\varepsilon)R$ for the rate of the code $\mathcal{C}_\tau$.

We have constructed a large class of rank permutation codes, associating them with binary and q-ary codes in the Hamming space. If the latter codes possess efficient decoding algorithms, then the methods discussed above translate these algorithms to decoding algorithms of rank permutation codes of essentially the same complexity. Thus, the existing theory of error-correcting codes can be used to design practical error-correcting codes and procedures for the rank modulation scheme.

Bibliography

[1] N. Ailon and E. Liberty, "Fast dimension reduction using Rademacher series on dual BCH codes," *Discrete & Computational Geometry*, vol. 42, pp. 615–630, 2009.

[2] N. Alon, J. Bruck, M. Naor, J. Naor, and R. Roth, "Construction of asymptotically good low-rate error-correcting codes through pseudo-random graphs," *IEEE Transactions on Information Theory*, vol. 38, no. 2, pp. 309–315, 1992.

[3] N. Alon, O. Goldreich, J. Håstad, and R. Peralta, "Simple constructions of almost k-wise independent random variables," *Random Structures and Algorithms*, vol. 3, pp. 289–304, 1992.

[4] N. Alon and J. Spencer, *The Probabilistic Method*, J. Wiley & Sons, 2000.

[5] I. Andriyanova, V. Rathi, and J.-P. Tillich, "Binary weight distribution of non-binary LDPC codes," *Proc. 2009 IEEE International Symposium on Information Theory*, Seoul, Korea, pp. 65–69, 2009.

[6] L. Appelbaum, S. D. Howard, S. Searle, R. Calderbank, "Chirp sensing codes: Deterministic compressed sensing measurements for fast recovery," *Applied and Computational Harmonic Analysis*, vol. 26, pp. 283–290, 2009.

[7] K. D. Ba, P. Indyk, E. Price, and D. Woodruff, "Lower bounds for sparse recovery," *Proc. 21th ACM SIAM Symposium on Discrete Algorithms*, Austin, 2010.

[8] B. Babadi and V. Tarokh, "Random frames from binary linear codes," *Proc. Conference on Information Sciences and Systems*, 2010.

[9] A. Barg, J. Justesen, and C. Thommesen, "Concatenated codes with fixed inner code and random outer code," *IEEE Transactions on Information Theory*, vol. 47, no. 1, pp. 361–365, 2001.

[10] A. Barg, and A. Mazumdar, "Codes in permutations and error correction for rank modulation," *IEEE Transactions on Information Theory*, vol. 56, no. 7, Jul 2010.

[11] A. Barg and A. Mazumdar, "Small ensembles of sampling matrices constructed from coding theory," *IEEE International Symposium on Information Theory*, Austin, pp. 1963–1967, Jul 13–18, 2010.

[12] A. Barg and A. Mazumdar, "On the number of errors correctable with codes on graphs," *IEEE Transactions on Information Theory*, vol. 57, no. 2, pp. 910–919, 2011.

[13] A. Barg and A. Mazumdar, "General constructions of deterministic matrices with (S)RIP for compressive sampling," *IEEE International Symposium on Information Theory*, 2011.

[14] A. Barg, A. Mazumdar and G. Zémor, "Weight distribution and decoding of codes on hypergraphs," *Advances in Mathematics of Communication*, vol. 2, no. 4, pp. 433–450, 2008.

[15] A. Barg, A. Mazumdar and G. Zémor, "Constructions of rank modulation codes," *IEEE International Symposium on Information Theory*, 2011.

[16] A. Barg and G. Zémor, "Distance properties of expander codes", *IEEE Transactions on Information Theory*, vol. 52, no. 1 , pp. 78–90, 2006.

[17] A. Barg and G. Zémor, "Concatenated codes: Serial and parallel," *IEEE Transactions on Information Theory*, vol. 51, pp. 1625–1634, 2005.

[18] A. Ben-Aroya and A.Ta-Shma, "Constructing small-bias sets from algebraic-geometric codes," *Proc. IEEE Foundations of Computer Science*, 2009.

[19] Y. Bilu and S. Hoory, "On codes from hypergraphs", *European Journal of Combinatorics*, vol. 25, pp. 339–354, 2004.

[20] I. F. Blake, G. Cohen, and M. Deza, "Coding with permutations," *Information and Control*, vol. 43, no. 1, pp. 1–19, 1979.

[21] E. L. Blokh and V. V. Zyablov, *Linear Concatenated Codes* (in Russian), Moscow, U.S.S.R.: Nauka, 1982.

[22] I. E. Bocharova, B. D. Kudryashov, R. Johannesson, and V. V. Zyablov, "Asymptotically good woven codes with fixed constituent convolutional codes," *Proc. IEEE International Symposium on Information Theory*, Nice, France, pp. 2326–2330, 2007.

[23] R. C. Bose and S. Chowla, "Theorems in the additive theory of numbers," *Commentarii Mathematici Helvetici*, vol. 37, no. 1, pp. 141-147, December 1962.

[24] J. Bourgain, S. J. Dilworth, K. Ford, S. Konyagin, and D. Kutzarova, "Explicit constructions of RIP matrices and related problems," arXiv:1008.4535.

[25] J. Boutros, O. Pothier, and G. Zémor, "Generalized low density (Tanner) codes," in *Proc. IEEE Intenational Conference on Communications*, Vancouver, Canada," vol. 1, pp. 441–445, 1999.

[26] D. Burshtein, "On the error correction of regular LDPC codes using the flipping algorithm," *IEEE Transactions on Information Theory*, vol. 54, no. 2, pp. 517–530, 2008.

[27] D. Burshtein and G. Miller, "Asymptotic enumeration methods for analyzing LDPC codes," *IEEE Transactions on Information Theory*, vol. 50, no. 6, pp. 1115–1131, June 2004.

[28] R. Calderbank, S. Howard, and S. Jafarpour "Construction of a large class of deterministic sensing matrices that satisfy a statistical isometry property," *IEEE Journal on Selected Topics in Signal Processing*, vol. 4, no. 2, pp. 358–374, April, 2010.

[29] R. Calderbank, S. Howard and S. Jafarpour, "A sublinear algorithm for sparse reconstruction with ℓ_2/ℓ_2 recovery guarantees," arXiv:0806.3799v2.

[30] P. Cameron, "Permutation codes,"*European Journal Combinatorics*, vol. 21, pp.482–490, 2010.

[31] E. J. Candès and Y. Plan, "Near-ideal model selection by ℓ_1 minimization," *The Annals of Statistics*, vol. 37, no. 5A, pp. 2145–2177, 2009.

[32] E. J. Candès, J. Romberg and T. Tao, "Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information," *IEEE Transactions on Information Theory*, vol. 52, no. 2, pp. 489–509, 2006.

[33] E. J. Candès and T. Tao, "Decoding by linear programming," *IEEE Transactions on Information Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[34] E. J. Candès, "The restricted isometry property and its implications for compressed sensing," *C. R. Academy of Science Paris*, Ser. I 346, pp. 589–592, 2008.

[35] H. Chadwick and L. Kurz, "Rank permutation group codes based on Kendall's correlation statistic," *IEEE Transactions on Information Theory*, vol. 15, no. 2, pp. 306-315, 1969.

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

[37] W. Chu, C.J. Colbourn ad P. Dukes, "Constructions for permutation codes in powerline communications," *Designs, Codes and Cryptography*, vol. 32, pp. 51–64, 2004.

[38] C. J. Colbourn, T. Kløve, and A. C. H. Ling, "Permutation arrays for power line communications and mutually orthogonal Latin squares," *IEEE Transactions on Information Theory*, vol. 50, no. 6, pp. 1289–1291, 2004.

[39] L. Comtet, *Advanced Combinatorics*, Dordrecht, Netherlands: Reidel, 1974.

[40] S. D. Constantin and T. R. N. Rao, "On the theory of binary asymmetric error correcting codes," *Information and Control*, vol. 40, pp. 20–36, 1979.

[41] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*, Springer-Verlag, New York-Berlin, 1988.

[42] W. Dai, O. Milenkovic and H. V. Pham, "Structured sublinear compressive sensing via dense belief propagation," arXiv:1101.3348.

[43] S. Dasgupta and A. Gupta, "An elementary proof of the Johnson Lindenstrauss lemma," *Random Structures and Algorithms*, vol. 22, no. 1, pp. 60–65, 2002.

[44] P. Delsarte and P. Piret, "Spectral enumerators for certain additive-error-correcting codes over integer alphabets," *Information and Control*, vol. 48, no. 3, pp. 193-210, 1981.

[45] H. Derksen, "Error-correcting codes and B_h-sequences," *IEEE Transactions on Information Theory*, vol. 50, no. 2, pp. 476–485, 2004.

[46] R. A. DeVore, "Deterministic constructions of compressed sensing matrices," *Journal of Complexity* vol. 23, pp. 918–925, August 2007.

[47] M. Deza and T. Huang, "Metrics on permutations, a survey," *Journal of Combinatorics, Information and System Sciences*, Vol. 23, no. 1-4, pp. 173–185, 1998.

[48] P. Diaconis and R. L. Graham, "Spearman's footrule as a measure of disarray," *Journal of the Royal Statistical Society, Series B*, vol. 39, no. 2, pp. 262-268, 1977.

[49] P. Elias, "Error-free coding," *IEEE Transactions on Information Theory*, PGIT-4: 29-37, 1954.

[50] W. Feller, *An Introduction to Probability Theory and its Applications*, vol. 1, 3rd. ed. New York, NY: Wiley, 1968.

[51] G. D. Forney, Jr., *Concatenated Codes*, Cambridge, MA: MIT Press, 1966.

[52] R. G. Gallager, *Low-Density Parity-Check Codes*, Cambridge, MA: MIT Press, 1963.

[53] I. M. Gessel and R. P. Stanley, "Algebraic enumeration," *Handbook of combinatorics*, vol. 2, pp. 1021-1061, Amsterdam, Netherlands: Elsevier, 1995.

[54] S. W. Golomb and L. R. Welch, "Perfect codes in the Lee metric and the packing of polyominoes," *SIAM Journal on Applied Mathematics*, vol. 18, no. 2, pp. 302–317, 1970.

[55] P. Gopalan, V. Guruswami and P. Raghavendra, "List decoding tensor products and interleaved codes," arXiv:0811.4395.

[56] R. L. Graham and N. J. A. Sloane, "Lower bounds for constant weight codes," *IEEE Transactions on Information Theory*, vol. 26, no. 1, pp. 37–43, 1980.

[57] A. Gurevich and R. Hadani, "The statistical restricted isometry property and the Wigner semicircle distribution of incoherent dictionaries," arXiv:0903.3627.

[58] S. Gurevich, R. Hadani and N. Sochen, "On some deterministic dictionaries supporting sparsity," *Journal on Fourier Analysis and Applications*, vol. 14, pp. 859–876, 2008.

[59] V. Guruswami and P. Indyk, "Near-optimal linear-time codes for unique decoding and new list-decodable codes over smaller alphabets," *Proc. ACM Symposium on Theory of Computing (STOC)*, Montreal, QC, Canada, pp. 812–821, May 2002.

[60] V. Guruswami, J. Lee, and A. Razborov, "Almost Euclidean sections of L_1^N via expander codes," *Proc. 19th ACM SIAM Symposium on Discrete Algorithms*, San Francisco, pp. 353–362, 2008.

[61] V. Guruswami and A. Rudra, "Concatenated codes can achieve list-decoding capacity," *Proc. Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, 258–267, ACM, New York, 2008.

[62] W. Hoeffding, "Probability inequalities for sums of bounded random variables," *Journal of American Statistical Associaton*, vol. 58, no. 301, pp. 13-30.

[63] S. Hoory, N. Linial and A. Wigderson, "Expander graphs and their applications," *Bulletin of American Mathematical Society (N.S.)*, vol. 43, no. 4, pp. 439–561, 2006.

[64] R. Horn and C. Johnson, *Matrix Analysis*, Cambridge, 1985.

[65] M. A. Iwen, "Simple deterministically constructible RIP matrices with sublinear Fourier sampling requirements," *Proc. Conference on Information Sciences and Systems*, pp. 870–875, 2009.

[66] A. Jiang, M. Langberg, R. Mateescu, and J. Bruck, "Data movement in flash memories," *Proc. 46th Annual Allerton Conference on Communication, Control and Computing*, pp. 1031–1038, Sept 2009.

[67] A. Jiang, R. Matescu, M. Schwartz, and J. Bruck, "Rank modulation for flash memories," *Proc. IEEE International Symposium on Information Theory (ISIT)*, Toronto, Canada, pp. 1731–1735, July 2008.

[68] A. Jiang, M. Schwartz, and J. Bruck, "Error-Correcting Codes for Rank Modulation," *Proc. IEEE International Symposium on Information Theory (ISIT)*, Toronto, Canada, pp. 1736–1740, July 2008.

[69] B. Kashin and V. Temlyakov, "A remark on compressed sensing," *Mathematical Notes*, vol. 82, no. 5-6, pp. 748–755, 2007.

[70] M. Kendall and J. D. Gibbons, Rank Correlation Methods, 5th ed., Edward Arnold Publishing, London, 1990.

[71] D. E. Knuth, *The Art of Computer Programming, Volume 3: Sorting and Searching*, Reading. MA: Addison-Wesley, 1973.

[72] P. D. Lax, *Linear Algebra*, J. Wiley & Sons, New York, N.Y., 1997.

[73] K. Li, L. Gan and C. Ling, "Orthogonal symmetric Toeplitz matrices for compressed sensing: statistical isometry property," arxiv:1012.5947, 2010.

[74] M. Lentmaier and K. Sh. Zigangirov, "On generalized low-density parity-check codes based on Hamming component codes," *IEEE Communications Letters*, vol. 3, pp. 248–260, 1999.

[75] V. I. Levenshtein, "Bounds on the maximum cardinality of codes with a bounded modulus of the scalar product," *Soviet Mathematics Doklady*, vol. 263, no. 6, pp. 1303–1308, 1982.

[76] R. Lidl, H. Niederreiter, *Finite Fields*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1983.

[77] S. Litsyn, V. Shevelev, "On ensembles of low-density parity-check codes: Asymptotic distance distributions," *IEEE Transactions on Information Theory*, vol. 48, no. 4, pp. 887–908, 2002.

[78] S. Litsyn, V. Shevelev, "Distance distributions in ensembles of irregular low-density parity-check codes," *IEEE Transactions on Information Theory*, vol. 49, no. 12, pp. 3140–3159, 2003.

[79] G. Louchard and H. Prodinger, "The number of inversions in permutations: A saddle point approach," *Journal of Integer Sequences*, vol. 6, Article 03.2.8 (electronic), 2003.

[80] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, 1977.

[81] B.H. Margolius, "Permutations with inversions," *Journal of Integer Sequences*, vol. 4, no. 2, Article 01.2.4, 13 pp. (electronic), 2001.

[82] A. Mazumdar, R. Roth and P. Vontobel, "On linear balancing sets," *Advances in Mathematics of Communications*, vol. 4, no. 3, August 2010.

[83] A. Mazumdar, A. Barg and N. Kashyap, "Coding for high density recording on a 1-d granular magnetic medium," *IEEE Transactions on Information Theory* (to appear), arxiv:1012.1895, 2010.

[84] A. Mazumdar and A. Barg, "Channels with intermittent errors," *IEEE International Symposium on Information Theory*, 2011.

[85] O. Milenkovic, E. Soljanin and P. Whiting, "Asymptotic spectra of trapping sets in regular and irregular LDPC code ensembles," *IEEE Transactions on Information Theory*, vol. 53, no. 1, pp. 39–59, 2007.

[86] A. Nilli, "On the second eigenvalue of a graph," *Discrete Mathematics*, vol. 91, no. 2, 207–210, 1991.

[87] A. Orlitsky, K. Viswanathan and J. Zhang,"Stopping set distribution of LDPC code ensembles," *IEEE Transactions on Information Theory*, vol. 51, no. 3, March 2005.

[88] H. Pham, W. Dai, and O. Milenkovic, "Sublinear compressive sensing reconstruction via belief propagation decoding," *Proc. IEEE International Symposium on Information Theory*, Seoul, South Korea, pp. 674–678, 2009.

[89] G. Sh. Poltyrev, "Bounds on the decoding error probability of binary linear codes via their spectra," *IEEE Transactions on Information Theory*, vol. 40, number 4, pp. 1284–1292, 1994.

[90] E. Porat, and A. Rothschild, "Explicit Non-Adaptive Combinatorial Group Testing Schemes," *Proc. 35th Int. Colloquium on Automata, Languages and Programming (ICALP)*, pp. 748–759, 2008.

[91] T. Richardson and R. Urbanke, *Modern Coding Theory*, Cambridge University Press, 2008.

[92] M. Schwartz, P. H. Siegel, and A. Vardy, "On the asymptotic performance of iterative decoders for product codes," *Proc. IEEE International Symposium on Information Theory*, pp. 1758–1762, 2005.

[93] M. Sipser and D. A. Spielman, "Expander codes," *IEEE Transactions on Information Theory*, vol. 42, no. 6, 1710–1722, 1996.

[94] V. Skachek and R. Roth, "Generalized minimum distance decoding of expander codes," *Proc. IEEE Information Theory Workshop*, Paris, France, pp. 245–248, Mar. 2003.

[95] X. Tang and R. Koetter, "Performance of iterative algebraic decoding of codes defined on graphs: An initial investigation," *Proc. 2007 IEEE Information Theory Workshop*, Lake Tahoe, CA, Sept. 2–6, pp. 254–259, 2007.

[96] R. M. Tanner, "A recursive approach to low-complexity codes," *IEEE Transactions on Information Theory*, vol. 26, no. 5, pp. 1710–1722, 1981.

[97] R. M. Tanner, "Superproduct codes with improved minimum distance," *Proc. 2002 IEEE International Symposium on Information Theory*, Sorrento, Italy, pp. 283.

[98] C. Thommesen, "The existence of binary linear concatenated codes with Reed-Solomon outer codes which asymptotically meet the Gilbert-Varshamov bound," *IEEE Transactions on Information Theory*, vol. 29, pp. 850–853, 1983.

[99] L. M. G. M. Tolhuizen, " The generalized Gilbert-Varshamov bound is implied by Turan's theorem," *IEEE Transactions on Information Theory*, vol. 43, no. 5, pp.1605–1606, 1997.

[100] J. A. Tropp, "On the conditioning of random subdictionaries," *Applied and Computational Harmonic Analysis*, vol. 25, pp. 1–24, 2008.

[101] R. R. Varshamov, "A class of codes for asymmetric channels and a problem from the additive theory of numbers", *IEEE Transactions on Information Theory*, vol. 19, no. 1, pp. 92–95, 1973.

[102] R. R. Varshamov and G. M. Tenenholtz, "A code for correcting a single asymmetric error," *Automat. Telemekh.*, vol. 26, no. 2, pp. 288–292, 1965.

[103] N. Y. Yu, "Deterministic compressed sensing matrices from multiplicative character sequences," arXiv:1011.2740.

[104] G. Zémor, "On expander codes," *IEEE Transactions on Information Theory*, vol. 47, no. 2, 835–837, 2001.

[105] M. Žnidarič, "Asymptotic expansion for inverse moments of Binomial and Poisson distributions,", *Open Statistics & Probability Journal*, vol. 1, pp. 7–10, 2009.

[106] V. V. Zyablov, R. Johannesson, and M. Lončar, "Low-complexity error correction of Hamming-code-based LDPC codes," *Problems of Information Transmission*, vol. 45, no. 2, pp. 95–109, 2009.

[107] V. V. Zyablov and M. S. Pinsker, "Estimation of the error-correcting complexity of Gallager low-density codes," *Problems of Information Transmission*, vol. 11, no. 1, pp. 18–28, 1975.

[108] V. V. Zyablov, "An estimate of complexity of constructing binary linear cascade codes," *Problems of Information Transmission*, voo. 7, no. 1, pp. 3–10, 1971.