List-Decodable Codes: (Randomized) Constructions and Applications

Nicolas Resch

CMU-CS-20-113

May 2020

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Venkatesan Guruswami, Co-Chair
Bernhard Haeupler, Co-Chair
Ryan O'Donnell
Madhu Sudan, Harvard University

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.

This research was sponsored by a fellowship from the National Sciences and Engineering Research Council Graduate Scholarships Doctoral program award number CGSD2-502898; and from three awards from the National Science Foundation: award number CCF1814603; award number CCF1422045; and, award number CCF1563742. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of any sponsoring institution, the U.S. government or any other entity.

Keywords: Coding Theory, List-Decoding, Pseudorandomness, Algebraic Constructions, Complexity Theory.

For my family (including Sparky and Loki!).

Abstract

Coding theory is concerned with the design of error-correcting codes, which are combinatorial objects permitting the transmission of data through unreliable channels. List-decodable codes, introduced by Elias and Wozencraft in the 1950's, are a class of codes which permit nontrivial protection of data even in the presence of extremely high noise. Briefly, a (ρ, L)-list-decodable code $\mathcal{C} \subseteq \Sigma^n$ guarantees that for any $z \in \Sigma^n$ the number of codewords at distance at most ρ from z is bounded by L. In the past twenty years, they have not only become a fundamental object of study in coding theory, but also in theoretical computer science more broadly. For example, researchers have uncovered connections to pseudorandom objects such as randomness extractors, expander graphs and pseudorandom generators, and they also play an important role in hardness of approximation.

The primary focus of this thesis concerns the construction of list-decodable codes. Specifically, we consider various random ensembles of codes, and show that they achieve the optimal tradeoff between decoding radius and rate (in which case we say that the code "achieves capacity"). Random linear codes constitute the ensemble receiving the most attention, and we develop a framework for understanding when a broad class of combinatorial properties are possessed by a typical linear code. Furthermore, we study random low-density parity-check (LDPC) codes, and demonstrate that they achieve list-decoding capacity with high probability. We also consider random linear codes over the rank metric, a linear-algebraic analog of the Hamming metric, and provide improved list-decoding guarantees.

We also provide non-randomized (i.e., explicit) constructions of list-decodable codes. Specifically, by employing the tensoring operation and some other combinatorial tricks, we obtain capacity achieving list-decodable codes with near-linear time decoding algorithms. Furthermore, the structure of these codes allows for interesting local decoding guarantees.

Finally, we uncover new connections between list-decodable codes and pseudorandomness. Using insights gleaned from recent constructions of list-decodable codes in the rank metric, we provide a construction of lossless dimension expanders, which are a linear-algebraic analog of expander graphs.

Acknowledgments

First of all, I would like to offer my sincerest thanks to my advisors, Venkat Guruswami and Bernhard Haeupler. While I know that this section is overly long, I cannot resist the urge to share a story. At the start of my second year of studies, I was feeling a bit depressed. While I was still very interested in theoretical computer science, the honest truth was that I was not overly excited about my own research. In my darkest moments, I wondered if I truly enjoyed research, and whether I had made the right choice to pursue a PhD. At the time Bernhard was my sole advisor, and when I revealed my doubts to him he was very understanding, and encouraged me to look around for research problems that I would find stimulating. At this time, I was taking *A Theorist's Toolkit*, an introductory theoretical computer science course co-instructed by Venkat. I began speaking to Venkat more regularly (including one particularly memorable trip to *Stack'd* where I learned about a problem concerning k-lifts of graphs) and, after I revealed to him that I was feeling a bit unenthused about my research directions, he promised to help "find [me] a problem that would keep [me] up at night".[1] Over time, we began working together more-and-more closely, and he later agreed to officially co-advise my thesis.

While I would like to say that from that moment onwards the rest of my PhD was smooth sailing, the reality is of course much messier. There were still highs and (many) lows. Fortunately, I now had two brilliant researchers looking out for me, and with their guidance I gradually got used to the trials and tribulations of research in theoretical computer science. Their patience and encouragement gave me the space to search for problems that excited me. Fortunately for me, Venkat shared my passion for algebra and he introduced me to its myriad uses in coding theory. As promised, I quickly built up a list[2] of fascinating research problems, and sure enough I began to suffer some sleepless nights as matrices and polynomials danced in my mind. Of course, this might sound to some like a cruel and unusual form of torture, but compared to the alternative of feeling unmotivated, this was a very welcome change of pace.

In our research discussions, the technical brilliance of my advisors helped guide me towards promising solutions (and away from the many deadends I discovered). Furthermore, at a more basic level, it was wonderful to interact regularly with people whose company I truly enjoyed. As much as I valued our stimulating research discussions, some of my best memories from my PhD experience involve shared pitchers of beer.[3] For all these reasons, and

[1]This was probably my introduction to Venkat's uncanny ability to pithily encapsulate complex ideas. Of course, this manifests itself in his technical writing, but also in his everyday conversations.

[2]Many uses of the word "list" in this thesis could be construed as a pun, given the thesis' topic. Unless otherwise noted, please excuse these as unintentional.

[3]Usually IPAs.

the many others that I do not have space to list, my advisors merit a most gracious thank you.

While I consider myself lucky to have had two advisors to guide me in my studies, I in fact received valueable guidance from many other mentors. First of all, I would like to thank the other members of my thesis committee, Ryan O'Donnell and Madhu Sudan. Both have been willing to discuss research problems with me when I asked them to, and their insightful questions have provided me with useful perspectives on the problems presented in this thesis.

Next, I was fortunate to have had multiple opportunities to visit academic institutions and work with other professors. First of all, in the summer of 2016 I visited Eric Blais at the University of Waterloo, where I learned all about property testing (and also was introduced to what it is like to be scooped). While the research we conducted there does not directly appear in this thesis, I would like to think that it influenced my presentation of the "local properties" we will encounter in Chapter 3. Next, during Venkat's sabbatical I had the opportunity to visit the *Center for Mathematical Sciences and Application (CMSA)*, affiliated with Harvard. There, I met more researchers than I have space to thank, but the many stimulating talks I attended and discussions I engaged in certainly left an indelible mark. In particular, I met Madhu; and furthermore, I met Noga Ron-Zewi, who invited me to visit her at the University of Haifa for a semester. There, along with fellow PhD student Shashwat Silas, we would engage in long daily research discussions. In particular, she introduced me to many effective combinatorial techniques for constructing list-decodable codes, and her expertise has greatly influenced the material and presentation of this thesis. Furthermore, I was constantly amazed by her desire to make our trip as comfortable as possible: for a specific example, I recall her making a large number of phone calls to local hospitals to ensure that Shashwat would be able to see an eye specialist after he suffered a tennis-induced injury.

Beyond the aforementioned researchers, I have benefited greatly from collaborations with many other people. In particular, I would like to aknowledge my co-authors: Venkat Guruswami, Swastik Kopparty, Ray Li, Jonathan Mosheiff, Noga Ron-Zewi, Shubhangi Saraf, Shashwat Silas, Mary Wootters, and Chaoping Xing. The work in this thesis has benefited greatly from their insights and efforts, and would not have been possible without them. In particular, while I have only known Jonathan for about a year and a half now, he introduced me to a new viewpoint on random linear codes which has greatly influenced my thinking.

Finally, I should probably address the elephant in the room. Or, to be precise, the elephant in my bedroom, where I am currently forced to work. I am writing this in spring 2020 during the height of the COVID-19 pandemic. When one is forced to socially isolate, it is easy to see just how valuable one's interpersonal relationships are. I consider myself truly fortunate to

have made so many great friends in the past five years. To my roommates Laxman Dhulipala, Fait Poms, Roie Levin, and Greg Kehne[4], thank you for the great companionship and indulging my culinary adventures. I like to say that I enjoy having roommates; equally likely is that I have been very fortunate with my choice of roommates. To Naama Ben-David, Angela Jiang, Ellen Vitercik, David Wajc and Rohan Sawhney: thank you for the Friday lunches. To Colin White, Rajesh Jarayam, Jonathan Laurent and my other great officemates: thank you for making the workplace fun. To John Wright, Euiwoong Lee, David Witmer and others: thank you for welcoming me into the theory group. To Sol Boucher and Connor Brem: thanks for introducing me to cycling (and especially for the insane early morning rides at freezing temperatures). To the Semi Regular Lorelei Crew: thanks for the (usually responsible) late-night drinking. Thanks also to Anson Kahng, Ellis Hershkowitz, Alex Wang, Bailey Flanigan, Nick Sharp and many others for all the great memories. I would also like to thank Faith Adebule for invaluable conversations. Last, but certainly not least, thanks to Vijay Bhattiprolu: for all the trips to cafes, being a great roommate in Boston, watching basketball with me, and many other cherished memories.

Lastly, I would like to thank some people from my life prior to coming to Pittsburgh. First of all, there are few people I'd rather virtually share a beer with than Kelsey Adams and Alex Freibauer: for all the FaceTime (now Zoom) calls, thank you. But, most importantly, I am greatly indebted to my family: my sister Katrin, my father Lothar and my mother Anne. For my entire life, they have been my greatest supporters, encouraging me in any intellectual pursuit that happens to take my fancy. I cannot imagine how I would have completed this thesis if I had not been able to unload all of my concerns on my parents in our weekly FaceTime calls (I have not quite managed to convince them to use Zoom yet). As much as I like to joke that I call mostly to see our beautiful dog (Sparky and later Loki), the reality is that I truly enjoy and value our conversations. For this, and for uncountably many other reasons, I will never be able to repay my debt to them. I hope this thesis is a small indication that their investment is paying off.

[4]Note these roommates were not concurrent!

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Consider the following scenario, which might hit a little too close to home in light of the current state of affairs.[1] There is an deadly outbreak of a new virus and the World Health Organization has announced a pandemic. For this reason, all persons have been asked to practice "social distancing", i.e., to maintain a greater than usual physical distance from one another and to avoid large congregations of people. Therefore, most people are spending nearly all of their time in their homes, and only venturing outdoors for basic necessities.

Feeling lonely, Alice wishes to send a message to her friend Bob. Unfortunately, Bob is feeling quite ill and, while he is unsure if he has contracted the virus (due to a dearth of available tests), is required to isolate himself at home for the next fourteen days. Thus, Alice chooses to send a message from a safe distance; perhaps she uses email or another online messaging service. Unfortunately, these communication networks can be unreliable: a package might be dropped, or some other error could be introduced in the transmission.

Fortunately for Alice, *error-correcting codes* have been introduced 70 years ago in the seminal works of Shannon [Sha48] and Hamming [Ham50] to address precisely this issue. Alice can take her desired message (for instance, "Get well soon!") and add some judiciously chosen redundancy: the message with the additional redundancy is called a *codeword*. Alice can then transmit this codeword to Bob and, so long as the channel connecting them does not introduce too much noise, Bob can decode the noisy codeword to obtain the message "Get well soon!".

However, in light of the dire state of affairs, more errors than expected are introduced in the transmission, and it is impossible for Bob to determine precisely what message Alice sent. Fortuitously, this eventuality was foreseen by Elias [Eli57] and Wozencraft [Woz58]. They proposed the study of *list-decodable* codes, which guarantee that Bob will be able to compute a short list of messages Alice could have sent. Ideally, Bob can use side information to deduce the message that Alice intended to send. And, even if this is not the case, a short list of possible messages is certainly more comforting

[1] This chapter was written in March 2020.

than no message at all.

List-decodable codes are the main object of study in this thesis. Our contributions can be largely divided into three categories. First, we study various random ensembles of codes, and develop tools to understand the list-decodability of a random code drawn from these distributions. Next, we provide explicit constructions of list-decodable codes which come equipped with extremely efficient decoding algorithms. Finally, we uncover new connections between list-decodable codes and other fields in theoretical computer; specifically, pseudorandomness.

In the next section, we provide a gentle introduction to error-correcting codes. In Section 1.2, we introduce list-decodable codes, and provide further motivation for their study.[2] A brief snapshot of the contributions of this thesis is given in Section 1.3.

1.1 Error-Correcting Codes

Briefly, error-correcting codes provide a systematic method of adding redundancy to messages so that two parties as above can communicate, even in the presence of noise. That is, if certain symbols in Alice's messages are corrupted, then Bob can still determine the message that Alice sent.

While we defer formal definitions to Chapter 2, in order to introduce error-correcting codes some terminology is useful. First, as alluded to earlier, the message with the additional redundancy is referred to as a *codeword*. The set of all codewords that could be obtained from a feasible set of messages is called an *error-correcting code*, or just a *code* for short. The potentially unreliable medium through which Alice transmits her message is called a *channel*.

If Alice encodes her length k message into a length n codeword, we say that Alice's code has *rate* $\frac{k}{n}$. This is a measure of the code's efficiency, or (non-)redundancy. In more detail, the larger a code's rate, the more information Alice is transmitting per symbol transmitted through the channel. For this reason, it is desirable to have codes with rate as large as possible: equivalently, Alice would like to add as little redundancy as possible.

However, if Alice does not add any redundancy, then the code will not provide any noise-resilience, which is the initial motivation for error-correcting codes! Thus, some redundancy is necessary. But how can we determine whether or not the redundancy is useful? That is, how can we mathematically ensure that the resulting code is actually capable of correcting errors?

To quantify a code's fault-tolerance, following Hamming [Ham50],[3] we consider a code's *distance*. Given two words, their distance is the fraction of symbols that need to

[2] We hope that our future readers will not find the "social distancing" motivation particularly relevant.

[3] In Shannon's model [Sha48], errors are introduced randomly. While this is an extremely important model studied by a thriving community of researchers, in this thesis we exclusively study Hamming's model of worst case errors.

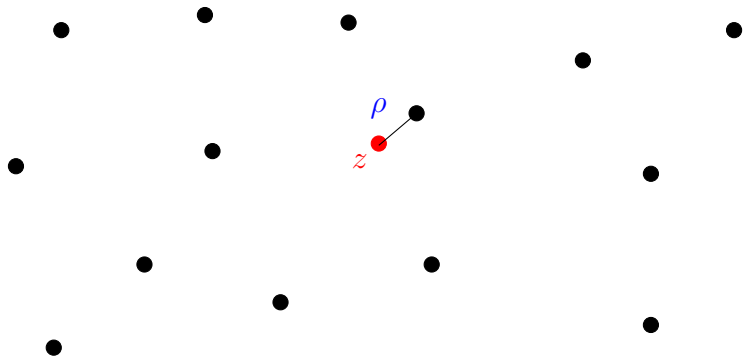

Figure 1.1: In the above figure, the black dots represent codewords and the red dot is the received word z, which is a codeword that has had a ρ-fraction of its symbols corrupted. So long as $\rho < \delta/2$, where δ is the minimum distance of the code, the codeword closest to z is *unique*, so Bob can determine the codeword (and hence, the message) Alice sent.

be changed to turn one word into the other. A code's distance is then the minimum distance between two distinct codewords.

Why is this measure useful? Suppose Alice's code has distance 0.1, i.e., every pair of codewords differ in at least 10% of their symbols. Furthermore, suppose that the transmission channel always corrupts less than 5% of the symbols in any codeword. Bob can then look for the codeword closest to the word he received: as every pair of codewords differ in 10% of the symbols, this codeword must be unique. In general, if Alice uses a code of distance δ, Bob can uniquely decode from a ρ-fraction of errors, assuming $\rho < \delta/2$ (a formal proof of this assertion follows from the triangle inequality). See Figure 1.1. However, observe that if there are two codewords c_1 and c_2 that differ in exactly a δ fraction of their coordinates where δn is even, there is a word that is at distance exactly $\delta/2$ from both of these codewords. If Bob receives this word, he cannot not be sure if c_1 or c_2 was transmitted. Thus, if $\rho \geq \delta/2$, unique decoding is impossible.

Thus, it is clear that we would like codes which have large distance *and* large rate. However, a moment's reflection shows that these two desiderata are in tension with one another. If a code is to have high rate, then we must include a large number of n-letter words in our code; if we have too many codewords, though, it is inevitable that two will be close together. In fact, understanding how large rate and distance can be simultaneously is one of the most fundamental questions in the theory of error-correcting codes. We discuss some of the known tradeoffs later in Section 2.4; for now, see Figures 1.2 and 1.3 for an illustration of this phenomenon.

Having established the basics of error-correcting codes, the main character of our story is ready to take center stage.

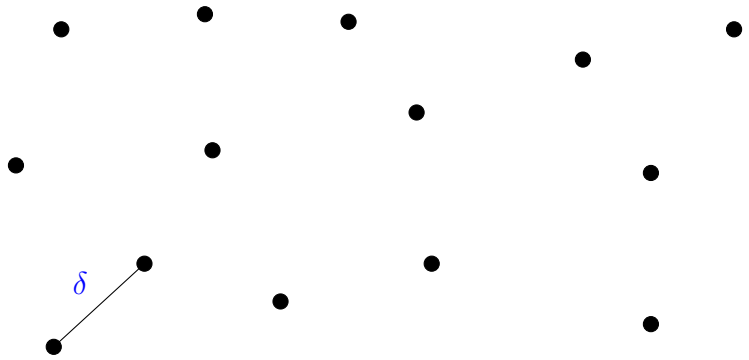

Figure 1.2: A code with minimum distance δ. That is, every pair of codewords differ in at least a δ-fraction of coordinates.

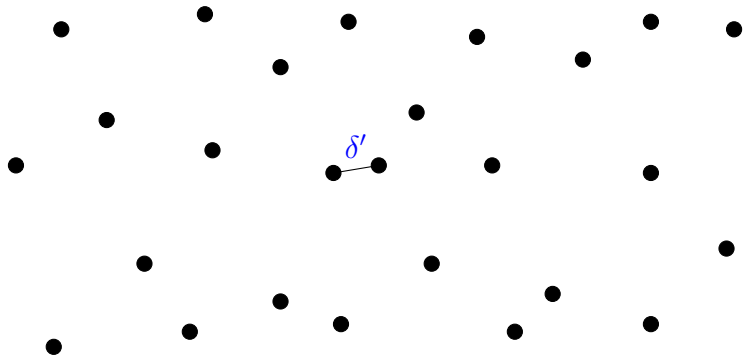

Figure 1.3: A higher rate code. The increased number of codewords leads to a decreased minimum distance $\delta' < \delta$.

1.2 List-Decodable Codes

As stated above, so long as the fraction of errors introduced by the channel is less than half the minimum distance, it is guaranteed that Bob can determine Alice's message. However, what if Alice and Bob are separated by a channel that corrupts, say, 51% of the transmitted symbols? Is there any hope for them to communicate in any meaningful way?

More generally, suppose Alice wishes to use a code of rate R to communicate through a channel which corrupts a ρ-fraction of symbols. Due to known rate-distance tradeoffs, it might be the case that any rate R code has distance at most 2ρ. As discussed earlier, there can be scenarios when Bob cannot uniquely decode Alice's message. Nonetheless, can Bob still hope to derive some useful information about Alice's codeword?

These questions were addressed by Elias [Eli57] and Wozencraft [Woz58]. These authors proposed that Bob could settle for a relaxation of unique decoding called *list-decoding*. In this relaxation, Bob is no longer required to output the unique closest codeword; instead, he merely tries to output a (hopefully short) list of codewords, one of which is guaranteed to be the codeword transmitted by Alice.

One natural hope is for the size of the list output by Bob to not be too large. Indeed, a trivial solution to the above problem would be for Bob to output every single codeword. Ideally, one hopes that Bob can use some additional information, obtained perhaps via other communications with Alice, to pin down the precise message that Alice intended to transmit. But if the list size is extremely large, it is unclear how useful this list is. At a more basic level, if we hope for Bob to be able to efficiently compute this list in time polynomial in n, at the very least the list he outputs must have size polynomial in n. Even better would be for the list size to be a constant, independent of n.

Perhaps surprisingly, *every* code is list-decodable with modest list sizes (e.g., 20) even if the channel corrupts more than $\delta/2$ fraction of the coordinates. In fact, *most* codes are list-decodable with constant list sizes, even if the fraction of errors introduced is very close to δ. Even more startling, nontrivial list-decoding guarantees can be provided even if 99% of the symbols are corrupted: that is, even if the noise far outweighs the signal, we can still recover useful information about the signal.

1.2.1 Motivations for List-Decoding

However, just because the notion of list-decoding is not vacuous, the reader could be justifiably wondering whether list-decoding is a useful notion. First of all, we indicated above that the list Bob outputs could potentially be pruned further if he has access to side information, or if he can engage in extra rounds of communication with Alice. Furthermore, as a code will necessarily be quite sparse, it turns out that it is actually quite rare that Bob will need to output a long list: that is, for most codewords and most error patterns corrupting fewer than a δ-fraction of the coordinates, the list Bob outputs will have size 1: that is to say, Bob will uniquely decode Alice's message. Thus,

providing a nontrivial decoding guarantee even when the channel corrupts more than $\delta/2$ errors is useful even if Bob hopes to uniquely decode.

Moreover, list-decoding and related notions have found an impressive number of applications in theoretical computer science. As a first example, list-decoding[4] has found many uses in computational complexity. Specifically, [Bab+91; STV01] use list-decodable codes to perform hardness amplification, which informally calls for the transformation of a problem which is slightly hard-on-average to another which is very hard-on-average. In a similar vein, [Lip90; CPS99; GRS06] use list-decodable codes to construct average to worst-case reductions.

As another example particularly relevant to certain results in this thesis, the field of pseudorandomness has benefited greatly from interactions with list-decoding and related notions. For instance, a particular list-decodable code called *Pavaresh-Vardy* codes [PV05] (named after the researchers who constructed them) were used by Guruswami, Umans and Vadhan [GUV09] to construct optimal seeded extractors. The substantial web of connections uncovered between list-decodable codes and other pseudorandom objects is expounded upon quite beautifully in a survey by Vadhan [Vad12].

Other applications of (objects connected to) list-decodable codes include cryptography [GL89; Hai+15; KNY17; BKP18], learning theory [GL89; KM93; Jac97], compressed sensing and sparse recovery [NPR12; Gil+13], group testing [INR10], and streaming algorithms [Lar+16]. In light of this extensive list of applications, we hope that even the most skeptical reader is willing to concede that list-decodable codes are worthy of study, and moreover make a compelling topic of study for a thesis.

1.3 Snapshot of Our Contributions

In this section, we provide an informal overview of the results contributed in this thesis. For more details and a discussion of the thesis' structure, please see Section 2.6.

The contributions of this thesis can be broadly broken into three main categories. The first and most substantial segment investigates the list-decodability of random ensembles of codes.[5] The second part provides an explicit construction of a list-decodable code with a notably efficient decoding algorithm. The final part adds to the list[6] of applications of list-decodable codes in other parts theoreotical computer science.

[4]More precisely, some of these applications require "local" list-decoding. Informally, in local list-decoding Bob is just required to output a short description of the potentially sent codewords. This notion is defined formally in Chapter 7; specifically, Section 7.2.

[5]This is the motivation for the word in parentheses in the title.

[6]Pun intended.

1.3.1 Random Ensembles of Codes

Nearly all codes encountered in practice have the desirable algebraic property of *linearity*. That is, mathematically, they are a subspace of a finite vector space. Linear codes offer many advantages over general codes. For example, linear codes come equipped with an efficient representation, which an arbitrary code need not possess. Also, coding theorists have developed many methods of taking a small inner code and, perhaps after combining the inner code with some other outer code, obtaining a larger code that inherits properties of the inner code. In many of these applications, the inner code is required to be linear.

In spite of their utility, our understanding of the list-decodability of linear codes is not complete: it is not clear if linear codes can be list-decoded with list sizes as small as general codes. In response to this, a line of work ([Gur+02; GHK11; CGV13; Woo13; RW14; RW18; LW18]) has studied the list-decodability of *random* linear codes. This is motivated by a remarkably general phenomenon that optimal constructions of combinatorial objects are often furnished by random constructions. While many techniques have been employed, we have not yet succeeded in completely nailing down the performance of random linear codes in all parameter regimes. Thus, a basic question this thesis will address is the following:

Question 1.3.1. How list-decodable are random linear codes?

Beyond the motivation stemming from the applicability of linear codes, the list-decodability of random linear codes shines a spotlight on interesting questions concerning the geometry of finite vector spaces. At a fundamental level, it asks to what extent random subspaces look like uniformly random subsets of the same density from the perspective of a tester that is able to look for densely clustered points. For this reason, answering Question 1.3.1 will necessarily provide deep insights into the geometry of finite dimensional vector spaces, which is mathematically interesting in its own right.

Next, low-density parity-check (LDPC) codes [Gal62] are a subclass of linear codes of fundamental importance: they are widely studied in theory and practice due to their efficient encoding and (unique) decoding algorithms. Unlike the situation with random linear codes, the list-decodability of random LDPC codes has not been previously studied. In this thesis, we show that random LDPC codes are essentially as list-decodable as random linear codes. In fact, we provide a reduction which demonstrates that any results we obtain on the list-decodability of random linear codes will immediately yield roughly equivalent results for random LDPC codes. In particular, this guarantees that LDPC codes can achieve list-decoding capacity (i.e., they approach the optimal tradeoff between rate and decoding radius).

Lastly, while most coding theorists use the Hamming metric to define distance between codewords, motivations stemming from network coding [KS11; SKK08] have led researchers to investigate codes over the *rank metric*, as introduced by Delsarte [Del78]. We provide new results concerning the list-decodability of random linear codes over the rank metric by adapting techniques which have proved effective for the analogous

problem in the Hamming metric.

1.3.2 Explicit Constructions of List Decodable Codes

While we are largely interested in random constructions of list-decodable codes, we also provide an explicit construction of a list-decodable code with an extremely efficient decoding algorithm. In more detail, we show how to use the tensoring operation to construct capacity-achieving codes which can be list-decoded deterministically in near-linear time. Moreover these codes have (nearly) constant list sizes and alphabets.

Moreover, the codes we construct allow for extremely efficient decoding algorithms that give nontrivial information about a single symbol of a codeword from a corrupted version of the codeword; namely, they come equipped with *local* decoding algorithms. By applying various combinatorial techniques (e.g., concatenation), we can prove the existence of interesting local codes over the binary alphabet.

1.3.3 Applications of List-Decodable Codes

As mentioned in Section 1.2.1, list-decodable codes have found numerous applications in other areas of theoretical computer science, and this is especially prominent within the field of pseudorandomness. We study a pseudorandom object called a *dimension expander*, which is an algebraic analog of an expander graph. We show that techniques similar to those developed in the context of list-decoding codes over the rank metric can be employed to construct dimension expanders with very good parameters. In fact, the dimension expanders we construct are *lossless*, which is a feat that has not yet been accomplished for the analogous problem on expander graphs.

1.3.4 Roadmap

In Chapter 2, we establish the necessary background for the technical content of this thesis. Our contributions are contained in Chapters 3–8. In Chapter 9 we summarize our results and discuss directions for future work that we find particularly stimulating.

Chapter 2

Preliminaries

In this chapter, we begin by setting notations and discussing certain conventions that we have adhered to as best we could to provide the clearest possible exposition. Then, in Section 2.2, we provide the basic definitions for error-correcting codes. Section 2.3 then provides the definition of list-decodable codes, as well as other related notions that we study in this thesis. Section 2.4 collects certain combinatorial facts about codes, especially known rate-distance tradeoffs, that we will often refer to later in the thesis. Lastly, Section 2.5 discusses explicit constructions of list-decodable codes, which provides context for the results presented in Chapter 7.

2.1 Notations, Conventions, and Basic Definitions

Given a positive integer n, we let $[n] = \{1, 2, \ldots, n\}$. For a finite set X, $|X|$ denotes its size, i.e., the number of elements in X. For a set X and an integer k, $\binom{X}{k}$ denotes the family of all k element subsets of X.

The symbols $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$ and $\mathbb{C}$ refer to (as normal) the set of positive integers,[1] the set of all integers, the set of rational numbers, the set of real numbers, and the set of complex numbers respectively. For an integer n, $\mathbb{Z}/n\mathbb{Z}$ refers to the ring of integers modulo n. The symbol $\mathbb{F}$ will always denote a field. Of particular importance are the *Galois fields* of order q, which we denote by $\mathbb{F}_q$. It is well-known that such fields exist if and only if q is a prime power, and moreover $\mathbb{F}_q$ and $\mathbb{F}_\ell$ are isomorphic if and only if $q = \ell$. Thus, we will refer to $\mathbb{F}_q$ as "the finite field of order q".[2]

We review the asymptotic Landau notation we use. Given two functions f and g of a growing (decreasing) positive parameter x, $f(x) = O(g(x))$ asserts that there exists a constant $C > 0$ such that for all x large enough (resp., small enough), $f(x) \leq Cg(x)$. We may also denote this by $f(x) \lesssim g(x)$. We write $f(x) = \Omega(g(x))$ (or $f(x) \gtrsim g(x)$) if

[1] So $0 \notin \mathbb{N}$.

[2] To be completely formal, one should fix an algebraic closure of $\mathbb{Z}/p\mathbb{Z}$ and then there is a unique degree d extension of $\mathbb{Z}/p\mathbb{Z}$ for each $d \in \mathbb{N}$.

$g(x) = O(f(x))$ and $f(x) = \Theta(g(x))$ if $f(x) = O(g(x))$ and $f(x) = \Omega(g(x))$.

To assert that f grows strictly slower than g, we write $f(x) = o(g(x))$ if $\lim_{x\to+\infty} \frac{f(x)}{g(x)} = 0$ when x is a growing parameter, and $f(x) = o(g(x))$ if $\lim_{x\to 0+} \frac{f(x)}{g(x)} = 0$ when x is a decreasing parameter. Complementarily, we write $f(x) = \omega(g(x))$ if $g(x) = o(f(x))$. We also occasionally write $o_n(1)$ to denote a quantity tending to 0 as $n \to \infty$.[3] Finally, the notation $f(x) \sim g(x)$ implies $f(x)/g(x) \to 1$ as x tends to its limit. If we us the symbol $\approx$, that means that we are being informal, and the equality is mostly stated for intuition's sake.

If a quantity C depends on some other parameter α and we wish to emphasize this dependence, we write $C = C_\alpha$ or $C = C(\alpha)$. Similarly, if the implicit constant in the Landau notation depends on a parameter α, we subscript it, e.g., $f(x) = O_\alpha(g(x))$ means there exists a positive constant $C = C_\alpha$ for which $f(x) \le C_\alpha g(x)$ for all x.

Unless specified otherwise, all logarithms are base 2; the logarithm with base e is denoted by $\ln$. We use $\exp(x)$ as shorthand for e^x. We also occasionally write $\exp_y(x)$, which we define to equal y^x.

Probabilistic notation. In general, we like to use ***boldface*** to denote random variables. Thus, if $(\Omega, \mathcal{F}, \mu)$ is a probability measure space, the notation $\boldsymbol{x} \sim \mu$ indicates that $\boldsymbol{x}$ is a random variable such that for each measurable set $A \in \mathcal{F}$,

$$\mathbb{P}(\boldsymbol{x} \in A) = \mu(A) \ .$$

If we wish to emphasize the fact that $\boldsymbol{x}$ is distributed according to μ, we will write

$$\mathop{\mathbb{P}}_{\boldsymbol{x}\sim\mu}(\boldsymbol{x} \in A) \ .$$

Admittedly, we will not typically require the full generality afforded by measure spaces: most distributions we encounter will be discrete (in fact, finite). In this case, for a countable universe U, $\mu : U \to [0,1]$ is a function satisfying $\sum_{x\in U} \mu(x) = 1$. To say that $\boldsymbol{x} \sim \mu$ then means that for each $x \in U$,

$$\mathop{\mathbb{P}}_{\boldsymbol{x}\sim\mu}(\boldsymbol{x} = x) = \mu(x) \ .$$

The *support* of $\boldsymbol{x}$ is $\operatorname{supp}(\boldsymbol{x}) := \{x \in U : \mathbb{P}(\boldsymbol{x} = x) > 0\}$, and we can analogously speak of the support of a distribution μ, i.e., $\operatorname{supp}(\mu) := \{x \in U : \mu(x) > 0\}$. For a finite subset $S \subseteq U$, we shorthand $\boldsymbol{x} \sim S$ to indicate that $\boldsymbol{x}$ is sampled uniformly from S. That is,

$$\mathop{\mathbb{P}}_{\boldsymbol{x}\sim S}(\boldsymbol{x} = x) = \begin{cases} \frac{1}{|S|} & \text{if } x \in S \\ 0 & \text{if } x \notin S \end{cases} \ .$$

[3]This notation is useful to emphasize what the growing parameter is, as 1 is of course independent of said parameter.

For an event $\mathcal{E}$, we let $\mathbb{I}(\mathcal{E})$ denote the random variable which is 1 if the event $\mathcal{E}$ occurs and 0 otherwise. This implies $\mathbb{E}[\mathbb{I}(\mathcal{E})] = \mathbb{P}(\mathcal{E})$.

If we (perhaps implicitly) have a family of events $\mathcal{E} = \{\mathcal{E}_n\}_{n\in\mathbb{N}}$, we say that $\mathcal{E}$ occurs *with high probability (whp)* if $\lim_{n\to\infty} \mathbb{P}(\mathcal{E}_n) = 1$. If $\mathbb{P}(\mathcal{E}_n) \geq 1 - \exp(-\Omega(n))$, we say that $\mathcal{E}$ occurs *with exponentially high probability*. If an event family occurs with high probability, then we say that it *almost surely occurs*. If $\lim_{n\to\infty} \mathbb{P}(\mathcal{E}_n) = 0$, then we say that $\mathcal{E}$ *almost surely does not occur*.[4]

Linear algebra. Given a vector space V over a field $\mathbb{F}$, we write $U \leq V$ if $U \subseteq V$ is a subspace of V. If $U_1, U_2 \leq V$ are subspaces, so is their intersection $U_1 \cap U_2$, as is their sum $U_1 + U_2 = \{u_1 + u_2 : u_1 \in U_1, u_2 \in U_2\}$. If V is equipped with a bilinear form $\langle \cdot, \cdot \rangle : V \times V \to \mathbb{F}$ and $U \leq V$, we can form the *dual space* of U, defined by

$$U^\perp = \{v \in V : \forall u \in U,\ \langle v, u \rangle = 0\} .$$

One can verify that $(U^\perp)^\perp = U$ and $\dim(U^\perp) = \dim(V) - \dim(U)$. For our purposes, we will typically have $V = \mathbb{F}_q^n$ and the bilinear form will be defined by

$$\langle u, v \rangle = \sum_{i=1}^{n} u_i v_i .$$

If the underlying field if $\mathbb{C}$, then the bilinear form will be defined by

$$\langle u, v \rangle = \sum_{i=1}^{n} u_i \overline{v_i} .$$

Next, given a linear map $T : V \to W$, the *image of* T is

$$T(V) = \mathrm{im}(T) = \{Tv : v \in V\} ,$$

which is a subspace of W. Note that if M is a matrix representing the linear transformation T in some (equivalently, any) basis, then $\mathrm{im}(T) = \text{col-span}(M)$, the span of the columns of M. Given a matrix M, we often find it convenient to identify it with the associated linear map, e.g., in the notation $\mathrm{im}(M)$. The *kernel of* T is

$$\ker(T) = \{v \in V : Tv = 0\} ,$$

which is a subspace of V. Note that if M represents T, then $\ker(T) = \text{row-span}(M)^\perp$, the orthogonal complement of the span of the columns of M. As before, we write $\ker(M)$ for the kernel of the associated linear map.

Finally, if M is an $m \times n$ matrix, $M_{i,*}$ for $i \in [m]$ denotes the i-th row of M, while $M_{*,j}$ for $j \in [n]$ denotes the j-th column.

[4]Not to be confused with the assertion that $\mathcal{E}$ does not almost surely occur, which merely means $\limsup_{n\to\infty} \mathbb{P}(\mathcal{E}_n) < 1$.

Information theory. We will require a few definitions from information theory. First, for a discrete[5] random variable $\boldsymbol{x}$, we define its *(Shannon) entropy* by

$$H(\boldsymbol{x}) := \sum_{x \in \operatorname{supp}(\boldsymbol{x})} \mathbb{P}(\boldsymbol{x} = x) \log \left(\frac{1}{\mathbb{P}(\boldsymbol{x} = x)} \right) .$$

(Recall our convention that $\log$ is base 2.) By continuity, we define $0 \log 0 = 0$. Occasionally, it will be convenient to use a base other than 2: for any $q > 1$, we define

$$H_q(\boldsymbol{x}) := \frac{H(\boldsymbol{x})}{\log q} = \sum_{x \in \operatorname{supp}(\boldsymbol{x})} \mathbb{P}(\boldsymbol{x} = x) \log_q \left(\frac{1}{\mathbb{P}(\boldsymbol{x} = x)} \right) .$$

to be the *q-ary (Shannon) entropy* of $\boldsymbol{x}$. If μ is a discrete distribution, we slightly abuse notation and let $H(\mu)$ denote $H(\boldsymbol{x})$ for any random variable $\boldsymbol{x}$ distributed according to μ, and similarly for $H_q(\mu)$.

If $\boldsymbol{y}$ is another random variable, the *joint entropy of $\boldsymbol{x}$ and $\boldsymbol{y}$*, $H(\boldsymbol{x}, \boldsymbol{y})$, is the entropy of the random variable $(\boldsymbol{x}, \boldsymbol{y})$. The *conditional entropy of $\boldsymbol{x}$ given $\boldsymbol{y}$* is

$$H(\boldsymbol{x}|\boldsymbol{y}) := \mathop{\mathbb{E}}_{y \sim \boldsymbol{y}} \left[H(\boldsymbol{x}|\boldsymbol{y} = y) \right] .$$

We record certain well-known facts concerning the entropy function.

Proposition 2.1.1 (Entropy Facts)**.** *Let $\boldsymbol{x}$ and $\boldsymbol{y}$ be discrete random variables.*

- ***Nonnegativity:*** *$H(\boldsymbol{x}) \geq 0$.*
- ***Log-support upper bound:*** *$H(\boldsymbol{x}) \leq \log(\operatorname{supp}(\boldsymbol{x}))$.*
- ***Conditioning cannot increase entropy:*** *$H(\boldsymbol{x}|\boldsymbol{y}) \leq H(\boldsymbol{x})$, with equality if and only if $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent.*
- ***Chain rule:*** *$H(\boldsymbol{x}, \boldsymbol{y}) = H(\boldsymbol{x}) + H(\boldsymbol{y}|\boldsymbol{x})$.*
- ***Data-processing inequality:*** *If $\boldsymbol{x}$ is distributed over a set $\mathcal{X}$ and $f : \mathcal{X} \to \mathcal{Y}$, then $H(f(\boldsymbol{x})) \leq H(\boldsymbol{x})$, with equality iff f is injective.*

Lastly, for two random variables $\boldsymbol{x}$ and $\boldsymbol{y}$, the *mutual information between $\boldsymbol{x}$ and $\boldsymbol{y}$* is

$$I(\boldsymbol{x}; \boldsymbol{y}) := H(\boldsymbol{x}) - H(\boldsymbol{x}|\boldsymbol{y}) = H(\boldsymbol{y}) - H(\boldsymbol{y}|\boldsymbol{x}) = H(\boldsymbol{x}) + H(\boldsymbol{y}) - H(\boldsymbol{x}, \boldsymbol{y}) ,$$

where the equalities are justified by the chain rule for entropy. Using the third bullet point of Proposition 2.1.1 ("conditioning cannot increase entropy"), we deduce that $I(\boldsymbol{x}; \boldsymbol{y}) \geq 0$ with equality if and only if $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent. Finally, $I_q(\boldsymbol{x}; \boldsymbol{y}) := \frac{I(\boldsymbol{x}; \boldsymbol{y})}{\log q}$ is the *q-ary mutual information*.

[5]One can consider the entropy of a continuous random variable, but we will not have cause to do so in this thesis.

2.2 Codes

Let Σ be a finite set of cardinality q and n a positive integer. An *error-correcting code* (or, to be brief, a *code*) is simply a subset $\mathcal{C} \subseteq \Sigma^n$. Elements $c \in \mathcal{C}$ are termed *codewords*. The integer n is referred to as the *block length*. The integer q is the *alphabet size*, and we deem $\mathcal{C}$ to be a *q-ary* code in this case. If $q = 2$, the code is deemed *binary*.

An important parameter is the *(information) rate* of the code, defined as

$$R = R(\mathcal{C}) := \frac{\log_q |\mathcal{C}|}{n} . \tag{2.1}$$

Intuitively, the rate quantifies the amount of information transmitted per symbol of the codeword. Thus, the rate is a measure of the code's efficiency: the larger R is, the more data we are able to communicate per codeword sent. If we don't normalize by n, we obtain the code's *dimension*, typically denoted $k := Rn$. (The justification for this terminology will be clear when we introduce linear codes.)

Another crucial parameter of a code is its distance. First, we assume that the set Σ^n is endowed with a metric $d : \Sigma^n \times \Sigma^n \to [0, \infty)$. For a code $\mathcal{C}$, its *(minimum) distance* is

$$\delta = \delta(\mathcal{C}) := \min\{d(c, c') : c, c' \in \mathcal{C}, c \neq c'\} .$$

Intuitively, the minimum distance of a code corresponds to its noise-resilience: if the distance of a code is larger, then more errors must be introduced to a codeword to cause it to be confused for a different codeword. For this reason, we seek codes with large distance. Unfortunately, this desideratum comes into conflict with that of large rate (recall Figures 1.2 and 1.3). We will explore this tension further in Section 2.4, but for now we will boldly proclaim that coding theory is at its core devoted to studying the achievable tradeoffs between rate and noise-resilience (where the noise models may vary).

The most popular choice for the metric d is the *(relative) Hamming distance* defined by

$$d_H(x, y) := \frac{1}{n} \cdot |\{i \in [n] : x_i \neq y_i\}| .$$

We also occasionally use the *absolute Hamming distance* $\Delta_H(x, y) := n \cdot d_H(x, y)$. As the Hamming metric is the metric we will most often encounter, it is typically denoted simply by d (and its unnormalized variant by Δ). Having said this, we will encounter another metric in this thesis: the *rank metric*. For two matrices $X, Y \in \mathbb{F}_q^{m \times n}$ with $n \leq m$, we define

$$d_R(X, Y) := \frac{1}{n} \operatorname{rank}(X - Y) .$$

If we don't normalize by n, then we obtain the *absolute rank distance* $\Delta_R(X, Y) := n d_R(X, Y)$.

A code $\mathcal{C} \subseteq \Sigma^n$ is often presented in terms of an *encoding map*, which is an injective function $\mathrm{Enc} : \Sigma_0^k \to \Sigma^n$ for which $\mathrm{Enc}(\Sigma_0^k) = \mathcal{C}$, where $k \in \mathbb{N}$ and Σ_0 is a finite alphabet.

A word $m \in \Sigma_0^k$ is called a *message* and $\mathrm{Enc}(m) \in \mathcal{C}$ is the *encoding of the message* m. We typically assume $\Sigma_0 = \Sigma$, in which case $k = Rn$, where R is the code's rate.

Formally, we will be interested in *families of codes*, which are an infinite collection $\mathcal{C} = \{\mathcal{C}_i : i \in \mathbb{N}\}$ such that each $\mathcal{C}_i$ is a code of blocklength n_i defined over an alphabet of size q_i such that $n_{i+1} > n_i$ and $q_{i+1} \geq q_i$ for all $i \in \mathbb{N}$. Then, we define $R(\mathcal{C}) = \liminf_{i\to\infty} R(\mathcal{C}_i)$ and $\delta(\mathcal{C}) = \liminf_{i\to\infty} \delta(\mathcal{C}_i)$. The code family $\mathcal{C}$ is said to be *asymptotically good* if $R(\mathcal{C}) > 0$ and $\delta(\mathcal{C}) > 0$. Even if this is not made explicit, when we speak of codes they will be defined in a uniform manner for infinitely many block lengths n, and so they do indeed constitute a family of codes satisfying this definition.

In this thesis, we will typically have $\Sigma = \mathbb{F}_q$, in which case $\mathbb{F}_q^n$ naturally forms a vector space of dimension n over the finite field $\mathbb{F}_q$. In this setting, we can speak of a *linear code*, which means that the subset $\mathcal{C} \subseteq \mathbb{F}_q^n$ also forms a subspace of $\mathbb{F}_q^n$. To emphasize this, we write $\mathcal{C} \leq \mathbb{F}_q^n$. Moreover, the formula for the rate a linear code is quite simple:

$$R = \frac{\dim \mathcal{C}}{n} .$$

Furthermore, if the metric d respects the additive structure of $\mathbb{F}_q^n$ in the sense that for all $x, y, z \in \mathbb{F}_q^n$,

$$d(x, y) = d(x + z, y + z) ,$$

then the minimum distance of a linear code satisfies

$$\delta = \min\{\mathrm{wt}(c) : c \in \mathcal{C} \setminus \{0\}\} ,$$

where we define $\mathrm{wt}(c) := d(c, 0)$ to be the *weight* of a codeword. Both metrics we consider in this thesis have this property.

There are two natural ways to present a linear code. Let $k = \dim(\mathcal{C})$. First of all, a linear code may be described via a *generator matrix* $G \in \mathbb{F}_q^{n\times k}$:[6]

$$\mathcal{C} = \mathrm{im}(G) = \text{col-span}(G) = \{Gx : x \in \mathbb{F}_q^k\} .$$

That is, a generator matrix is obtained by choosing a basis for the vector space $\mathcal{C}$, and making them the columns of a matrix. The "dual view" is to look at the linear space $\mathcal{C}$ in terms of the linear contraints defining it. That is, we can take a *parity-check* matrix $H \in \mathbb{F}_q^{(n-k)\times n}$ such that

$$\mathcal{C} = \ker(H) = (\text{row-span}(H))^{\perp} = \{x \in \mathbb{F}_q^n : Hx = 0\} .$$

Observe that if $\mathcal{C}$ has generator matrix G and parity-check matrix H, then $HG = 0$. From a computational perspective, linear codes possess two desirable properties:

- *Efficient Storage.* By storing either the generator matrix or the parity-check matrix, a linear code $\mathcal{C}$ can be stored with only $O(n^2)$ field symbols.

[6]In much of the coding theory literature, it is popular to view codewords as row-vectors. In this thesis, however, we find it more convenient to view codewords as column vectors, so that is the viewpoint we take.

- *Efficient Encoding.* Given a generator matrix G for a linear code and a message $x \in \mathbb{F}_q^k$, we can encode x by computing the matrix-vector product Gx, which can be performed in $O(nk)$ time.

However, the task of efficiently *decoding* a linear code is NP-hard in the worst case, implying that we can't in general expect efficient decoding algorithms for linear codes.

Finally, while we will not make regular use of this notation, it is standard to write $[n, k, d]_q$ for a linear code over $\mathbb{F}_q$ of block length n, dimension k (and hence rate $\frac{k}{n}$) and minimum *absolute* distance d (so minimum (relative) distance $\frac{d}{n}$).

2.2.1 Random Ensembles of Codes

This thesis is largely concerned with random ensembles of codes. That is, we fix a distribution over codes contained in (i.e., subsets of) Σ^n, and consider the performance of a code drawn from this distribution with respect to various measures. For now, we introduce two basic random models of codes; we will introduce more as we progress.

The simplest random ensemble is the *uniform* ensemble. For a desired rate $R \in (0, 1)$, a *uniformly random code of rate* R is a random subset $\boldsymbol{\mathcal{C}} \subseteq \Sigma^n$ obtained by including each element $x \in \Sigma^n$ in $\boldsymbol{\mathcal{C}}$ independently with probability $q^{-(1-R)n}$. Note that the expected size of such a code satisfies $\mathbb{E}|\boldsymbol{\mathcal{C}}| = q^n \cdot q^{-(1-R)n} = q^{Rn}$: thus, in expectation the code has rate R. Furthermore, a Chernoff bound demonstrates that with probability at least $1 - \exp(-\Omega(n))$, $|\boldsymbol{\mathcal{C}}| \geq q^{Rn}/2$, and thus the designed rate and the actual rate differ by a $o(1)$ term with exponentially high probability. For this reason, when we sample a uniformly random code of rate R, we assume it has rate exactly R, as this negligibly affects any of the stated results.

As the collection of events "$x \in \boldsymbol{\mathcal{C}}$" for $x \in \Sigma^n$ are independent, we have the following basic fact.

Proposition 2.2.1. *Let $n \in \mathbb{N}$ and $R \in (0, 1)$. Let $\boldsymbol{\mathcal{C}} \subseteq \Sigma^n$ be a uniformly random code of rate R. Then, for any $S \subseteq \Sigma^n$ of cardinality d,*

$$\mathbb{P}\left(S \subseteq \boldsymbol{\mathcal{C}}\right) = q^{-dn(1-R)} .$$

If this thesis has a protagonist, it is played by the ensemble of random linear codes. A *random linear code* $\boldsymbol{\mathcal{C}}$ of rate $R \in (0, 1)$ is obtained by sampling $\boldsymbol{G} \sim \mathbb{F}_q^{n \times k}$ uniformly, where $k = \lfloor Rn \rfloor$,[7] and setting

$$\boldsymbol{\mathcal{C}} = \mathrm{im}(\boldsymbol{G}) = \text{col-span}(\boldsymbol{G}) = \{\boldsymbol{G}x : x \in \mathbb{F}_q^k\} .$$

Of course, it could happen that $\boldsymbol{\mathcal{C}}$ has dimension smaller than k, which occurs if and only if $\boldsymbol{G}$ has rank less than k. The probability this $\boldsymbol{G}$ has rank k is precisely

$$q^{-nk} \prod_{j=0}^{k-1} (q^n - q^j) = \prod_{j=0}^{k-1} (1 - q^{j-n}) \geq 1 - \sum_{j=0}^{k-1} q^{j-n} = 1 - q^{-n} \sum_{j=0}^{k-1} q^j \geq 1 - q^{k-n} .$$

[7] For readability, in the sequal the floor is typically omitted.

Thus, since $k = \lfloor Rn \rfloor$ and we will always think of $R \in (0,1)$ as being bounded away from 1, we have $\dim(\boldsymbol{\mathcal{C}}) = k$ with exponentially high probability.

Naturally, there is a dual viewpoint on the sampling procedure for a random linear code: one samples $\boldsymbol{H} \sim \mathbb{F}_q^{(n-k)\times n}$ uniformly and then puts

$$\boldsymbol{\mathcal{C}} = \ker(\boldsymbol{H}) = (\text{row-span}(\boldsymbol{H}))^{\perp} = \{x \in \mathbb{F}_q^n : \boldsymbol{H}x = 0\} .$$

While both viewpoints are useful, the writer of this document tends to be biased towards the second viewpoint, and so arguments will predominantly consider random parity-check matrices rather than random generator matrices.

In contrast to Proposition 2.2.1, the following proposition demonstrates that the probability a set is contained in a random linear code is controlled by the rank of the set.

Proposition 2.2.2. *Let $n \in \mathbb{N}$, q a prime power, and $R \in (0,1)$ such that Rn is an integer. Let $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ be a random linear code of rate R. For any set $S \subseteq \mathbb{F}_q^n$ of rank d,*

$$\mathbb{P}\left(S \subseteq \boldsymbol{\mathcal{C}}\right) = q^{-dn(1-R)} .$$

Proof. Let $\{v_1, \ldots, v_d\}$ be a maximal linearly independent set in S. For $k = Rn$, let $\boldsymbol{h}_1, \ldots, \boldsymbol{h}_{n-k}$ denote the rows of $\boldsymbol{H}$, which are independent, uniform vectors in $\mathbb{F}_q^n$. For each $i \in [d]$ and $j \in [n-k]$, $\langle \boldsymbol{h}_j, v_i \rangle$ is distributed uniformly over $\mathbb{F}_q$. Furthermore, the linear independence of $v_1, \ldots, v_d$ guarantees that the random variables $\langle \boldsymbol{h}_j, v_1 \rangle, \ldots, \langle \boldsymbol{h}_j, v_d \rangle$ are stochastically independent for each $j \in [n-k]$. Hence, the set of random variables $\{\langle \boldsymbol{h}_j, v_i \rangle : i \in [d], j \in [n-k]\}$ are independent, uniform elements of $\mathbb{F}_q$. Moreover, for any vector $v \in \text{span}\{v_1, \ldots, v_d\}$, if $\langle \boldsymbol{h}_j, v_i \rangle = 0$ for all i, j, then also $\langle \boldsymbol{h}_j, v \rangle = 0$. Thus,

$$\mathbb{P}\left(\boldsymbol{H}v = 0 \; \forall v \in S\right) = \mathbb{P}\left(\langle \boldsymbol{h}_j, v_i \rangle = 0 \; \forall i \in [d], j \in [n-k]\right) = q^{-d(n-k)} = q^{-dn(1-R)} . \quad \square$$

The takeaway message is that, for a random linear code $\boldsymbol{\mathcal{C}}$, linear independence of a set $\{v_i\}$ implies stochastic independence of the events $\{v_i \in \boldsymbol{\mathcal{C}}\}$.

2.3 List-Decodable Codes and Friends

If random linear codes are the protagonist of this thesis, list-decoding is their challenge.

Throughout this section, Σ denotes a finite alphabet. First of all, we recall the definition of a ball in a metric space, specialized to the setting of codes.

Definition 2.3.1 (Ball)**.** Let $z \in \Sigma^n$ and $\rho > 0$. The *ball of radius ρ centered at z* is

$$B(z, \rho) = \{x \in \Sigma^n : d(x, z) \leq \rho\} .$$

If we wish to emphasize the block length n, we superscript it, i.e., we denote $B^n(z, \rho)$. When the metric d is the Hamming metric, we refer to the corresponding balls as *Hamming balls*. When $d = d_R$ is the rank metric, we call them *rank metric balls*.

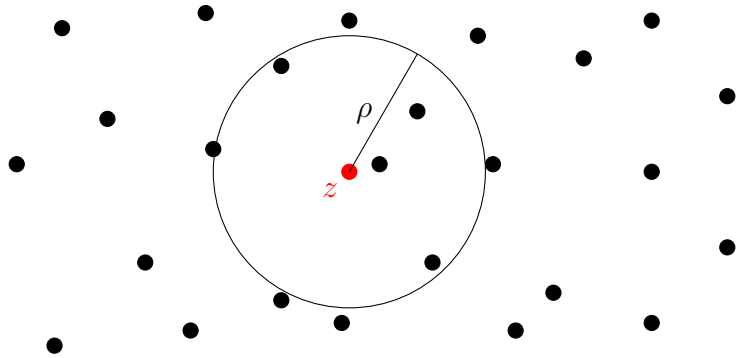

Figure 2.1: An illustration of (ρ, L)-list-decodability. The black dots represent codewords; the red dot is any center z. The guarantee is that any Hamming ball as above contains at most L codewords.

Definition 2.3.2 (List-Decodable Code)**.** Let $\rho > 0$ and $L \in \mathbb{N}$. A code $\mathcal{C} \subseteq \Sigma^n$ is said to be (ρ, L)*-list-decodable* if for all $z \in \Sigma^n$,

$$|B(z, \rho) \cap \mathcal{C}| \leq L \, . \tag{2.2}$$

The parameter L is called the *list size*.

For a code $\mathcal{C}$, the largest ρ such that Eq. (2.2) holds is called the *list-of-L decoding radius* of $\mathcal{C}$. Informally, when we say that $\mathcal{C}$ has list decoding radius ρ, this means that Eq. (2.2) holds for ρ with $L \leq \mathrm{poly}(n)$. For an illustation of list-decodability, see Figure 2.1.

A slight strengthening of this notion is furnished by average-radius list-decodability.

Definition 2.3.3 (Average-Radius List-Decodable Code)**.** Let $\rho > 0$ and $L \in \mathbb{N}$. A code $\mathcal{C} \subseteq \Sigma^n$ is said to be (ρ, L)*-average-radius list-decodable* if for all $z \in \Sigma^n$ and subsets $\Lambda \subseteq \mathcal{C}$ of size $L + 1$,

$$\frac{1}{L+1} \sum_{c \in \Lambda} d(c, z) > \rho \, . \tag{2.3}$$

Note that the condition in Definition 2.3.3 is stricter than that in Definition 2.3.2: if every set of $L + 1$ codewords has average distance greater than ρ from z, it cannot be that some set of $L + 1$ codewords all have distance at most ρ from z. If we wish to emphasize that we are referring to the standard notion of list-decodability (that is, Definition 2.3.2), then we will occasionally add the qualifier *absolute*. Similar to above, the maximum ρ for which (2.3) holds is the *list-of-L average-decoding radius* and, if L is polynomially bounded, just the average-decoding radius. For an illustration of average-radius list-decodability, see Figure 2.2. As motivation for the study of average-radius list-decodability, note that by turning to this concept one is essentially replacing a maximum by an average, which is natural from a mathematical perspective. Furthermore, this viewpoint has helped establish connections between list-decoding and other problems, e.g., compressed sensing [CGV13].

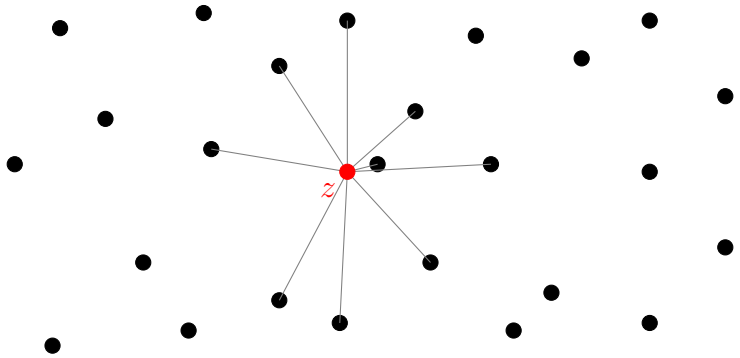

Figure 2.2: An illustration of (ρ, L)-average-radius list-decodability. The black dots represent codewords; the red dot is any center z. The guarantee is that if one chooses $L+1$ codewords, their average distance to z is greater than ρ.

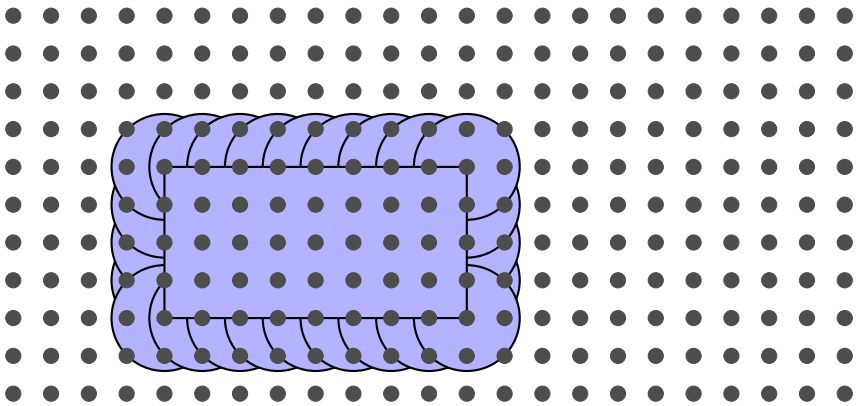

Figure 2.3: An illustration of a "puffed-up rectangle" $B(S, \rho)$. We fix a combinatorial rectangle $S = S_1 \times \cdots \times S_n$, and then put a ball of radius ρ around each point in S.

Another way to generalize Definition 2.3.2 is to study list-recovery. Informally, list-recovery calls for list-decoding with only "soft information" on the coordinates. This property has only been studied in the context of the Hamming metric,[8] so in the remainder of this section we specialize to this setting. To provide the formal definition, we first introduce some notation. For a string $z \in \Sigma^n$ and a tuple $S = (S_1, \ldots, S_n) \in \binom{\Sigma}{\ell}^n$, we define

$$d(x, S) := \min\{d(x, y) : y \in S_1 \times \cdots \times S_n\} = \frac{1}{n} \cdot |\{i \in [n] : x_i \notin S_i\}| \, .$$

For $\rho > 0$, we define $B(S, \rho) = \{x \in \Sigma^n : d(x, S) \leq \rho\}$. Geometrically, one can think of the set $B(S, \rho)$ as the combinatorial rectangle $S_1 \times \cdots \times S_n$ puffed-up by Hamming balls; see Figure 2.3.[9]

[8] However, there is a natural analog for the rank-metric; we comment upon this in Chapter 8.

[9] If $\Sigma = \mathbb{F}_q$ and we define $A + B := \{a + b : a \in A, b \in B\}$ for two subsets $A, B \subseteq \mathbb{F}_q^n$, then $B(S, \rho) = S_1 \times \cdots \times S_n + B(0, \rho)$; this provides more formal justification for the given mental picture.

Definition 2.3.4 (List-Recoverable Code)**.** Let $\rho > 0$ and $\ell, L \in \mathbb{N}$. A code $\mathcal{C} \subseteq \Sigma^n$ is said to be (ρ, ℓ, L)*-list-recoverable* if for all $S \in \binom{\Sigma}{\ell}^n$,

$$|\mathcal{C} \cap B(S, \rho)| \leq L \, .$$

Observe that $(\rho, 1, L)$-list-recoverability is the same as (ρ, L)-list-decodability.

Remark 2.3.5. When $\ell > 1$, $(0, \ell, L)$-list-recovery is still a nontrivial property; for brevity, it is termed (ℓ, L)*-zero-error* list-recovery. Geometrically, the guarantee is that no combinatorial rectangle of bounded size intersects the code too much.

List-recovery was initially introduced as a stepping stone towards list-decodable and uniquely-decodable codes [GI01; GI02; GI03; GI04]. In recent years, it has proved to be a useful primitive in its own right, with a long list of applications outside of coding theory [INR10; NPR12; Gil+13; Hai+15; Dor+19]. Specifically, the connections between codes and pseudorandom objects discussed in Section 1.2 actually typically require list-recoverability.

Finally, we can obtain a common generalization of Definitions 2.3.3 and 2.3.4.

Definition 2.3.6 (Average-Radius List-Recoverable Code)**.** Let $\rho > 0$ and $\ell, L \in \mathbb{N}$. A code $\mathcal{C} \subseteq \Sigma^n$ is said to be (ρ, ℓ, L)*-list-recoverable* if for all $S \in \binom{\Sigma}{\ell}^n$ and $\Lambda \subseteq \mathcal{C}$ of size $L + 1$,

$$\frac{1}{L+1} \sum_{c \in \Lambda} d(c, z) > \rho \, .$$

Again, average-radius list-recoverability is a stronger guarantee than standard list-recoverability, and $(\rho, 1, L)$-average-radius list-recoverability recovers (ρ, L)-average-radius list-decodability. If we wish to emphasize that we are referring to the standard notion of list-recoverability we may add the qualifier *absolute*.

2.4 Combinatorial Bounds on Codes

In this section we collect several well-known combinatorial bounds on codes to which we will make repeated reference in this thesis. All of these results are specialized to the Hamming metric; for analogous results over the rank metric, see Section 5.1.

2.4.1 Rate-Distance Tradeoffs

First, we state the fundamental Singleton bound.

Theorem 2.4.1 (Singleton Bound [Sin64])**.** *Let Σ be a finite set and n a positive integer. Let $\mathcal{C} \subseteq \Sigma^n$ be a code of rate R and distance δ. Then*

$$R \leq 1 - \delta + 1/n \, .$$

To see this, consider *puncturing* the code to $(1-\delta)n+1$ coordinates, i.e., for some $S \subseteq [n]$ of size $(1-\delta)n+1$, consider the code $\mathcal{C}_S = \{c_S = (c_i)_{i\in S} : c \in \mathcal{C}\}$. Observe that $|\mathcal{C}_S| = |\mathcal{C}|$: otherwise, we would have two codewords for which the set of coordinates on which they disagree is confined to $[n] \setminus S$, a set of size $\delta n - 1$, and this contradicts the assumption that $\mathcal{C}$ has distance δ. Hence, $R(\mathcal{C}) = \frac{1}{n}\log_q |\mathcal{C}| = \frac{1}{n}\log_q |\mathcal{C}_{[n]\setminus S}| \leq (1-\delta) + \frac{1}{n}$.

This bound is actually achievable, and any code achieving this tradeoff between rate and distance is called *maximum distance separable*, or MDS for short. In fact, up to isomorphism the only known MDS code is the famous Reed-Solomon code,[10] which we introduce in Example 2.5.1.

The Singleton bound gives an impossibility result for a rate-distance tradeoff: it shows that they cannot both be too large. The following result, known as the Gilbert-Varshamov bound (or GV bound for short), is a possibility result: it asserts that a rate-distance tradeoff is achievable by a code family. Before stating the result, we must introduce the q-ary entropy function, which will make many appearances in this thesis.

Definition 2.4.2 (q-ary entropy function)**.** Let $q \geq 2$ be an integer. Define $h_q : [0,1] \to [0,1]$ by

$$\begin{aligned} h_q(x) &= x\log_q(q-1) + x\log_q\left(\frac{1}{x}\right) + (1-x)\log_q\left(\frac{1}{1-x}\right) \\ &= x\log_q\left(\frac{q-1}{x}\right) + (1-x)\log_q\left(\frac{1}{1-x}\right) . \end{aligned}$$

When $q = 2$, we refer to $h_2(x) = x\log_2\frac{1}{x} + (1-x)\log_2\frac{1}{1-x}$ as the *binary entropy function*, and we typically denote it simply by $h(x)$.

Remark 2.4.3. To justify the name, note that if $\boldsymbol{x} \sim \mathrm{Ber}(p)$, i.e., $\boldsymbol{x}$ is 1 with probability p and 0 with probability $1-p$, then $H(\boldsymbol{x}) = h(p)$, where $H(\cdot)$ is the Shannon entropy. More generally, if $\boldsymbol{x}$ is distributed over $\{0,1,\ldots,q-1\}$ and $\mathbb{P}(\boldsymbol{x}=0) = 1-p$ and $\mathbb{P}(\boldsymbol{x}=x) = \frac{p}{q-1}$ for all $x \neq 0$, then $H_q(\boldsymbol{x}) = h_q(p)$.

Theorem 2.4.4 (Gilbert-Varshamov Bound [Gil52; Var57])**.** *Let q be a positive integer. There exists a family of codes $\mathcal{C} = \{\mathcal{C}_n : n \in \mathbb{N}\}$ such that each $\mathcal{C}_n$ has block length n for which $R = R(\mathcal{C}) = \lim_{n\to\infty} R(\mathcal{C}_n)$ and $\delta = \delta(\mathcal{C}) = \lim_{n\to\infty}\delta(\mathcal{C}_n)$ satisfy*

$$R \geq 1 - h_q(\delta) .$$

There are two main ways to prove this theorem. One can either construct a code greedily by adding codewords so long as the code does not violate the distance constraint, or by observing that random linear codes of rate $1 - h_q(\delta) - \varepsilon$ have distance at least δ with probability $\geq 1 - q^{-\varepsilon n}$. The second of these arguments is most in the spirit of the techniques employed in this thesis, and so we sketch it now. For a random linear code $\boldsymbol{\mathcal{C}}$ to have distance less than δ, there must be a vector $x \in B(0,\delta)$ such that $x \in \boldsymbol{\mathcal{C}}$.

[10] The famous MDS conjecture asserts that this is not due to a lack of ingenuity.

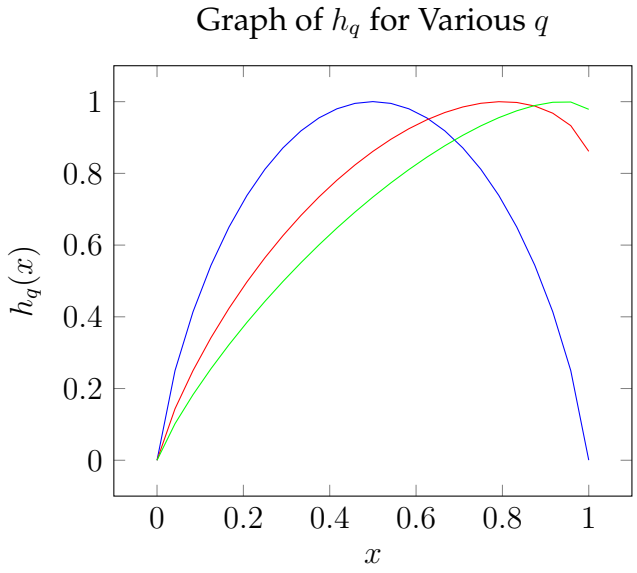

Figure 2.4: Graph of $h_q(x)$ for various values of alphabet size q. In blue, $q = 2$; in red, $q = 5$; in green, $q = 17$. Note that as q increases, $h_q(x) \to x$; we quantify this below.

By a union bound, this probability is at most

$$\sum_{x \in B(0,\delta)} \mathbb{P}\left(x \in \mathcal{C}\right) = |B(0,\delta)| q^{-(1-R)n} \,. \tag{2.4}$$

To bound this final expression, we must understand the quantity

$$|B(0,\delta)| = \sum_{i=0}^{\lfloor \delta n \rfloor} \binom{n}{i} (q-1)^i \,. \tag{2.5}$$

The following estimate is of fundamental importance.

Proposition 2.4.5. *Let q be a positive integer and fix $\delta \in (0, 1 - 1/q)$. For any $z \in [q]^n$,*

$$q^{nh_q(\delta) - o(n)} \leq |B(0,\delta)| \leq q^{nh_q(\delta)} \,.$$

Thus, the exceptional probability in (2.4) is at most $q^{h_q(\delta)n} q^{-(1-R)n}$, which is in turn at most $q^{-\varepsilon n}$ if $R \leq 1 - h_q(\delta) - \varepsilon$.

For a proof of Proposition 2.4.5, one can see, e.g., [GRS12, Proposition 3.3.1]. Alternatively, one can observe that it is the $\ell = 1$ case of Proposition 2.4.11, for which we do provide a proof.

While achieving the Singleton bound exactly is only known to be possible if $q \geq n$, the following estimate shows that we can get ε-close to the Singleton bound assuming $q \geq \exp(\Omega(1/\varepsilon))$.

Proposition 2.4.6 ([GRS12, Proposition 3.3.2]). *For small enough $\varepsilon > 0$, $1-h_q(\delta) \geq 1-\delta-\varepsilon$ for every $\delta \in (0, 1-1/q]$ if and only if $q \geq \exp(\Omega(1/\varepsilon))$.*

As a corollary of Theorem 2.4.4, we conclude that there are (linear) codes over $\mathbb{F}_q$ of rate $1-\delta-\varepsilon$ with distance δ if $q \geq \exp(\Omega(1/\varepsilon))$.

Before concluding this subsection, we note that there do exist explicit families of codes beating the GV bound (at least, for $q \geq 49$ of the form p^{2t} for p a prime and $t \in \mathbb{N}$): these are the famous Goppa codes of Tsfasman, Vlădut and Zink [Gop81; TVZ82]. However, for other values of q (in particular, for $q = 2$), the GV bound essentially represents the best known achievable tradeoff between rate and distance.

2.4.2 List-Decoding Tradeoffs

In this subsection, we discuss the achievable tradeoffs between the rate R, decoding radius ρ and list size L of a list-decodable code. We also discuss generalizations of these results to the case of list-recovery, where we have the additional parameter ℓ, the input list size.

Capacity Theorems

Unlike the situation for rate-distance tradeoffs, the best achievable tradeoff between rate and list-decoding radius is known. Recall that we say a code $\mathcal{C} \subseteq \Sigma^n$ has list-decoding radius ρ if $|\mathcal{C} \cap B(z, \rho)| \leq \text{poly}(n)$ for all $z \in \Sigma^n$. The following theorem precisely determines the largest rate R of a code with list decoding radius ρ.

Theorem 2.4.7 (List Decoding Capacity Theorem). *Let $n \in \mathbb{N}$ and Σ a finite alphabet of size q. Fix $\rho \in (0, 1-1/q)$ and $\varepsilon > 0$.*

- *There exists a code $\mathcal{C} \subseteq \Sigma^n$ of rate $1-h_q(\rho)-\varepsilon$ which is $(\rho, O(1/\varepsilon))$-list decodable.*
- *For any code $\mathcal{C} \subseteq \Sigma^n$ of rate $1-h_q(\rho)+\varepsilon$, there exists a center $z \in \Sigma^n$ such that $|\mathcal{C} \cap B(z, \rho)| \geq q^{\varepsilon n - o(n)}$.*

Thus, for list-decoding up to radius ρ with polynomially-sized lists,[11] $1-h_q(\rho)$ is the *capacity*. If a code $\mathcal{C}$ has rate $1-h_q(\rho)-\varepsilon$ for some small constant $\varepsilon > 0$ and has list-decoding radius at least ρ with L, then we say that $\mathcal{C}$ *achieves list-decoding capacity*, or just *achieves capacity* if list-decoding is clear from the context. By Theorem 2.4.7, such codes achieve the optimal tradeoff between decoding radius and rate. As a final piece of terminology, we refer to $\varepsilon = 1-h_q(\rho)-R$ as the *gap to capacity*.

Remark 2.4.8. Recall that Proposition 2.4.6 states that for large enough q (i.e., $q \geq \exp(\Omega(1/\varepsilon))$), $1-h_q(\rho) \geq 1-\rho-\varepsilon$. Thus, for the large q regime, we refer to $1-\rho$ as the list decoding capacity. Alternatively, recalling Theorem 2.4.1, we might state that a capacity-achieving code is "list decodable up to the Singleton bound", as they are list decodable up to radius $\rho = 1-R-\varepsilon$.

[11] Or even constant-sized lists.

As the List-Decoding Capacity Theorem is of fundamental importance, and because it introduces certain techniques that will recur throughout this thesis, we provide its proof.

Proof of Theorem 2.4.7. The first item follows by considering the performance of a uniform random code $\boldsymbol{\mathcal{C}}$ of rate $1 - h_q(\rho) - \varepsilon$. Observe that $\boldsymbol{\mathcal{C}}$ fails to be (ρ, L)-list decodably if and only if there exists a center $z \in \Sigma^n$ and a subset $\{x_1, \ldots, x_{L+1}\} \subseteq B(z, \rho)$ such that $x_i \in \boldsymbol{\mathcal{C}}$ for all $i \in [L+1]$. By a union bound and the independence of the events "$x_i \in \boldsymbol{\mathcal{C}}$", the probability of failure is at most

$$\sum_{z \in \Sigma^n} \sum_{\{x_1, \ldots, x_{L+1}\} \subseteq B(z,\rho)} \mathbb{P}\left(\forall i \in [L+1],\ x_i \in \boldsymbol{\mathcal{C}}\right) \leq q^n \cdot q^{(L+1)nh_q(\rho)} q^{-(1-R)n(L+1)} \,. \tag{2.6}$$

Substituting $R = 1 - h_q(\rho) - \varepsilon$ into (2.6) and simplifying, we obtain $q^n \cdot q^{-\varepsilon(L+1)n}$. Thus, if $L \geq 1/\varepsilon$, the probability of failure is $q^{-\varepsilon n}$. Finally, note that a Chernoff bound implies that $|\boldsymbol{\mathcal{C}}| \geq q^{Rn}/2$ with probability at least $1 - \exp(-\Omega(n))$. Thus, with exponentially high probability, the code $\boldsymbol{\mathcal{C}}$ is (ρ, L)-list decodable and has rate $\geq 1 - h_q(\rho) - \varepsilon - o_n(1)$.

We now establish the second item. Let $\mathcal{C}$ be a code of rate at least $1 - h_q(\rho) + \varepsilon$ and let $\boldsymbol{z} \sim \Sigma^n$ be uniform. We compute the expectation

$$\begin{aligned} \mathbb{E}|B(\boldsymbol{z}, \rho) \cap \mathcal{C}| &= \sum_{c \in \mathcal{C}} \mathbb{E}\left[\mathbb{I}\left(c \in B(\boldsymbol{z}, \rho)\right)\right] = \sum_{c \in \mathcal{C}} \mathbb{E}\left[\mathbb{I}\left(\boldsymbol{z} \in B(c, \rho)\right)\right] \\ &= \sum_{c \in \mathcal{C}} \frac{|B(c, \rho)|}{q^n} = |\mathcal{C}| \cdot \frac{|B(0, \rho)|}{q^n} \,. \end{aligned} \tag{2.7}$$

Proposition 2.4.5 tells us $\frac{|B(0,\rho)|}{q^n} \geq q^{-(1-h_q(\rho))n - o(n)}$. Hence, using the assumption $|\mathcal{C}| \geq q^{n(1-h_q(\rho)+\varepsilon)}$, we find that (2.7) is at least

$$q^{n(1-h_q(\rho)+\varepsilon)} \cdot q^{-(1-h_q(\rho))n - o(n)} = q^{\varepsilon n - o(n)} \,.$$

Hence, by the probabilistic method, there must exist a $z \in \Sigma^n$ for which $|B(z, \rho) \cap \mathcal{C}| \geq q^{\varepsilon n - o(n)}$, as claimed. □

Next, we remark that there is a similar list-recovery capacity theorem. While we have seen mentions of such a theorem in the literature (e.g., [RW18]), the following form does not appear to be present. In analogy to (2.5), for $S \in \binom{[q]}{\ell}$, we seek an estimate for

$$|B(S, \rho)| = \sum_{i=0}^{\rho n} \binom{n}{i} (q - \ell)^i \ell^{n-i} \,. \tag{2.8}$$

Much as the q-ary entropy function provided an effective estimate for the cardinality of Hamming balls, the following function will prove useful:

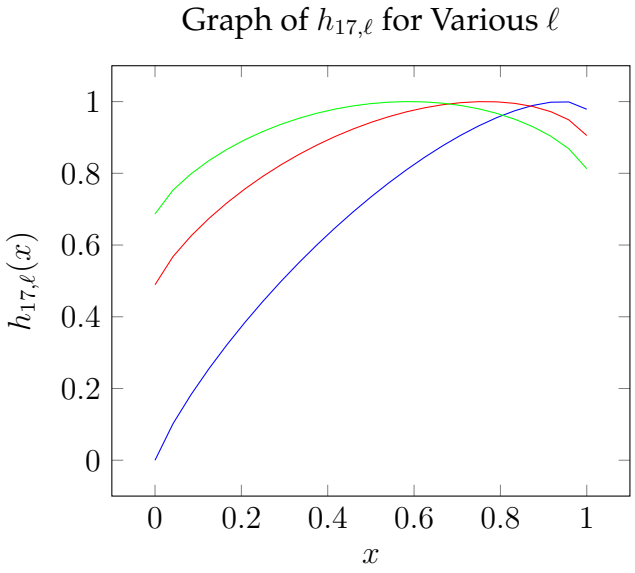

Figure 2.5: Graph of $h_{17,\ell}(x)$ for various values of input list size ℓ. In blue, $\ell = 1$; in red, $\ell = 4$; in green, $\ell = 7$. Note that $h_{q,\ell}(0) = \log_q \ell$ and $h_{q,\ell}(1 - \ell/q) = 1$, and that $h_{q,\ell}$ increases monotonically between these points.

Definition 2.4.9 ((q,ℓ)-ary entropy function)**.** Let $1 \le \ell \le q$ be integers. Define $h_{q,\ell} : [0,1] \to [\min\{\log_q \ell, \log_q(q-\ell)\}, 1]$ by

$$\begin{aligned} h_{q,\ell}(x) &= x \log_q(q-\ell) + (1-x)\log_q \ell - x \log_q x - (1-x)\log_q(1-x) \\ &= x \log_q \left(\frac{q-\ell}{x}\right) + (1-x)\log_q\left(\frac{\ell}{1-x}\right) . \end{aligned}$$

Remark 2.4.10. Akin to the probabilistic interpretation of h_q (cf. Remark 2.4.3), $h_{q,\ell}(p)$ is the q-ary entropy of a random variable $\boldsymbol{x}$ distributed over $[q]$ such that, for some $S \in \binom{[q]}{\ell}$,

$$\mathbb{P}(\boldsymbol{x} = x) = \begin{cases} \frac{1-p}{|S|} & x \in S , \\ \frac{p}{q-|S|} & x \notin S . \end{cases}$$

Furthermore, $h_{q,\ell}$ satisfies the following symmetry:

$$h_{q,\ell}(p) = h_{q,q-\ell}(1-p) .$$

We provide plots of this function in Fig. 2.5 for $q = 17$ and various choices of ℓ. Now, we justify the claim that the (q,ℓ)-entropy function effectively characterizes the cardinality of puffed-up rectangles, i.e., the quantity in (2.8).

Proposition 2.4.11. *For any integers $1 \le \ell \le q$, $\rho \in (0, 1-1/q)$ and $S \in \binom{[q]}{\ell}^n$,*

$$q^{h_{q,\ell}(\rho)n - o(n)} \le |B(S,\rho)| \le q^{h_{q,\ell}(\rho)n}$$

Proof. As a first step, note the identity

$$q^{h_{q,\ell}(\rho)} = (q-\ell)^{\rho}\ell^{(1-\rho)}\left(\frac{1}{\rho}\right)^{\rho}\left(\frac{1}{1-\rho}\right)^{(1-\rho)} . \tag{2.9}$$

Now, we prove the upper bound.

$$\begin{aligned}
1 = (\rho + (1-\rho))^n &= \sum_{i=0}^{n}\binom{n}{i}\rho^i(1-\rho)^{n-i} \geq \sum_{i=0}^{\rho n}\binom{n}{i}\rho^i(1-\rho)^{n-i} \\
&= \sum_{i=0}^{\rho n}\binom{n}{i}(q-\ell)^i\left(\frac{\rho}{q-\ell}\right)^i \ell^{n-i}\left(\frac{1-\rho}{\ell}\right)^{n-i} \\
&= \sum_{i=0}^{\rho n}\binom{n}{i}(q-\ell)^i\ell^{n-i}\left(\frac{\rho\ell}{(q-\ell)(1-\rho)}\right)^i\left(\frac{1-\rho}{\ell}\right)^n \\
&\geq \sum_{i=0}^{\rho n}\binom{n}{i}(q-\ell)^i\ell^{n-i}\left(\frac{1-\rho}{\ell}\right)^n\left(\frac{\rho\ell}{(q-\ell)(1-\rho)}\right)^{\rho n} \\
&= \sum_{i=0}^{\rho n}\binom{n}{i}(q-\ell)^i\ell^{n-i}\left(\frac{\rho}{q-\ell}\right)^{\rho n}\left(\frac{1-\rho}{\ell}\right)^{(1-\rho)n} \\
&= |B(S,\rho)| \cdot q^{-h_{q,\ell}(\rho)n} ,
\end{aligned}$$

where the last equality uses (2.8) and (2.9).

For the lower bound, we use Stirling's approximation to obtain

$$\binom{n}{\rho n} = \frac{1}{\rho^{\rho n}(1-\rho)^{(1-\rho)n}}q^{-o(n)} ,$$

and so

$$\begin{aligned}
|B(S,\rho)| &\geq \binom{n}{\rho n}(q-\ell)^{\rho n}\ell^{(1-\rho)n} \\
&= \frac{(q-\ell)^{\rho n}\ell^{(1-\rho)n}}{\rho^{\rho n}(1-\rho)^{(1-\rho)n}}q^{-o(n)} \\
&= q^{h_{q,\ell}(\rho)n-o(n)} .
\end{aligned}$$

□

The following capacity theorem now follows from Proposition 2.4.11.

Theorem 2.4.12 (List-Recovery Capacity Theorem)**.** *Let $n \in \mathbb{N}$, Σ an alphabet of size q and $\ell \in \mathbb{N}$ satisfying $1 \leq \ell \leq q$. Fix $\rho \in (0, 1-\ell/q)$ and $\varepsilon > 0$.*

- *There exists a code $\mathcal{C} \subseteq \Sigma^n$ of rate $1 - h_{q,\ell}(\rho) - \varepsilon$ which is $(\rho, \ell, O(\ell/\varepsilon))$-list-recoverable.*
- *For any code $\mathcal{C} \subseteq \Sigma^n$ of rate $1-h_{q,\ell}(\rho)+\varepsilon$, there exists $S \in \binom{\Sigma}{\ell}^n$ such that $|\mathcal{C} \cap B(S,\rho)| \geq q^{\varepsilon n - o(n)}$.*

Thus, for list-recovering up to radius ρ with input lists of size ℓ and polynomially-sized output lists, $1 - h_{q,\ell}(\rho)$ is the *(list-recovery) capacity*.

Remark 2.4.13. Recall zero-error list-recovery (Remark 2.3.5); as $h_{q,\ell}(0) = \log_q \ell$, we conclude that the *zero-error list-recovery capacity* is $1 - \log_q \ell$.

Analogously to Proposition 2.4.6, one can show that assuming $q \geq \exp(\Omega(\log(\ell)/\varepsilon))$, the capacity for $(\rho, \ell, \mathrm{poly}(n))$-list recovery is at least $1 - \rho - \varepsilon$.

Proposition 2.4.14. *For small enough $\varepsilon > 0$, $1 - h_{q,\ell}(\rho) \geq 1 - \rho - \varepsilon$ for every $\rho \in (0, 1 - \ell/q)$ if and only if $q \geq \exp(\Omega(\log(\ell)/\varepsilon))$.*

Finally, we remark that for the average-radius variants of list decoding and recovery, the capacities are unchanged. The second bullet-points of Theorem 2.4.7 and 2.4.12 still hold, as average-radius is a stricter requirement. To establish the first bullet-point for Theorem 2.4.7, it suffices to observe that the number of tuples $(x_1, \ldots, x_{L+1}) \in (\Sigma^n)^{L+1}$ with average distance at most ρ from a center $z \in \Sigma^n$ is $|B^{(L+1)n}(z)|$, which is at most $q^{(L+1)nh_q(\rho)}$; this is precisely the bound we used in (2.6), except in that case it was for the number of $(L+1)$-element subsets of $B^n(z, \rho)$.

Other Combinatorial Bounds for List-Decoding and Recovery

Have established the capacity for list-decoding and related notions, we now discuss some other combinatorial bounds.

Johnson bound. First of all, we state the fundamental Johnson bound [Joh62; Joh63].[12] The Johnson bound asserts that *any* code has list-decoding radius strictly larger than half its minimum distance. In fact, the proof applies equally well to *average-radius* list-decoding. The following version is taken from Guruswami's thesis; another relevant citation is [AVZ00].

Theorem 2.4.15 (Johnson Bound, [Gur04, Corollary 3.3])**.** *Let $\mathcal{C} \subseteq [q]^n$ be a code of distance at least δ. If $\rho < 1 - \sqrt{1 - (1-\gamma)\delta}$ for $\gamma \in (0, 1)$, then $\mathcal{C}$ is $(\rho, 1/\gamma)$-average-radius list-decodable.*

We also provide the following variant of the Johnson bound which applies to (average-radius) list-recovery.

Theorem 2.4.16 (Johnson Bound for List Recovery, [Gop+18, Lemma 5.2])**.** *Let $\mathcal{C} \subseteq [q]^n$ be a code of distance at least δ. If $\rho < 1 - \sqrt{\ell(1-\delta)}$, then $\mathcal{C}$ is (ρ, ℓ, L)-average-radius list-recoverable with $L = \frac{\delta\ell}{(1-\rho)^2 - \ell(1-\delta)}$.*

Briefly, all proofs of the Johnson bound proceed by considering the sum

$$\sum_{i<j} d(c_i, c_j)$$

[12] Interestingly, it is known that there is no result analogous to the Johnson bound for the rank-metric. For details, see [WZ13].

in two ways, where $c_1, \ldots, c_L \in \mathcal{C}$ are distinct codewords. The lower bound follows easily from the distance assumption; an upper bound is obtained via a convexity argument.

The important (and perhaps surprising) conclusion to be drawn from the Johnson bound is that every code of positive distance can be list-decoded beyond radius $\delta/2$, even with *constant* list sizes. Moreover, so long as $\delta > 1 - 1/\ell$, list-recovery at positive radius is combinatorially feasible.

Lower bounds on list sizes. Lastly, we wish to highlight certain results that derive lower bounds on list sizes. It is known that a typical code with gap to capacity ε requires lists of size $\Theta(1/\varepsilon)$;[13] it is natural to ask whether *every* list-decodable code requires lists of size $\Omega(1/\varepsilon)$. Blinovsky [Bli86; Bli05] appears to be the first researcher to make progress on this question; amongst other results, he demonstrated that lists of size $\Omega_\rho(\log(1/\varepsilon))$ are necessary. The first work applies only to binary codes while the second applies to general q-ary alphabets.[14] Later, Guruswami and Vadhan [GV10] considered the high-noise regime and deduced that for a code to be $(1-(1+\eta)/q, L)$-list-decodable, L must be at least $\Omega_q(1/\eta^2)$. Later, Guruswami and Narayanan [GN14] showed that for *average-radius* list-decoding of binary codes, $L \geq \Omega_\rho(1/\sqrt{\varepsilon})$ is necessary.

2.5 Code Families

In this section, we briefly survey the field of algorithmic list-decoding, that is, explicit constructions of codes equipped with efficient list-decoding algorithms. In this thesis, unless specified otherwise, an *explicit construction* of a family of codes $\{\mathcal{C}_{n_i}\}_{i\in\mathbb{N}}$ is an algorithm Cons which takes as input n_i written in unary and outputs a description of the code $\mathcal{C}_{n_i}$ in time $\mathrm{poly}(n_i)$, and a code is deemed *explicit* if such an algorithm exists. We are particularly interested in surveying explicit codes equipped with efficient list-decoding algorithms. These results largely provide relevant context for Chapter 7. For explicit constructions of rank metric codes, see Sections 5.1 and 8.1.

It is difficult to conceive of a thesis on error-correcting codes which does not at some point introduce Reed-Solomon codes; this thesis is no exception.

Example 2.5.1 (Reed-Solomon Codes)**.** Reed-Solomon (RS) codes [Ree54; RS60] are defined in terms of polynomials over finite fields. We assume $q \geq n$, and let $\alpha_1, \ldots, \alpha_n \in \mathbb{F}_q$ be distinct field elements. For an integer $k \leq n$, identify $\mathbb{F}_q^k$ with $\mathbb{F}_q[X]_{<k}$, the space of degree $< k$ polynomials with coefficients in $\mathbb{F}_q$. The "message" $p(X) = \sum_{i=0}^{k-1} c_i X^i \in \mathbb{F}_q[X]_{<k}$ is mapped to the codeword $(p(\alpha_1), \ldots, p(\alpha_n))$, where $p(\alpha_j) = \sum_{i=0}^{k-1} c_i \alpha_j^i$ is the

[13] We established the upper bound in proving Theorem 2.4.7 above; for the lower bound, see [GN14, Theorem 20], or even [Rud11].

[14] His argument is in fact precise enough to give nontrivial upper bounds on rate for every finite list size L.

standard polyomial evaluation. That is,

$$\mathrm{RS}[n,k] = \{(p(\alpha_1), \ldots, p(\alpha_n)) : p \in \mathbb{F}_q[X]_{<k}\} .$$

Thanks to the "degree mantra", i.e., the fact that a non-zero polynomial of degree at most $k-1$ can have at most $k-1$ roots, it follows that every non-zero codeword has at least $n-k+1$ non-zero entries. That is, the minimum distance δ is at least $1-R+1/n$ (where $R = \frac{k}{n}$ is the rate), so Reed-Solomon codes achieve the Singleton bound (Theorem 2.4.1).

Furthermore, thanks to the Welch-Berlekamp algorithm [WB86], it is known how to uniquely decode Reed-Solomon codes up to half the minimum distance in polynomial time. Lastly, these codes can be efficiently list-decoded up to the Johnson bound (Theorem 2.4.15) via the celebrated Guruswami-Sudan algorithm [GS99].

The main drawback of Reed-Solomon codes is the requirement that the field size exceed the block-length. We next introduce the basic premise behind algebraic-geometry (AG)/Goppa codes, which address this concern. The precise details of the construction do not really interest us (only in Section 8.4 do we employ similar techniques), but as we will employ AG codes in the codes constructed in Chapter 7 a few words are merited.

Example 2.5.2 (AG/Goppa Codes)**.** Goppa observed that Reed-Solomon codes can be thought of as being obtained as the evaluation of rational functions with bounded poles at infinity (and no poles anywhere else) at every point on the line $\mathbb{F}_q$ (in fact, one could take the projective plane $\mathbb{P}^1(\mathbb{F}_q) = \mathbb{F}_q \cup \{\infty\}$). Goppa [Gop81] suggested one could take other curves in the projective plane $\mathbb{P}^2(\mathbb{F}_q)$ and evaluate functions with bounded poles at all points on the curve; the hope is that the curve will have more than q points, allowing for block lengths which exceed the field size.

Quite spectacularly, this approach has led to the construction of explicit codes with rate-distance tradeoffs exceeding the GV bound [TVZ82]. Furthermore, natural modifications of the Welch-Berlekamp and Guruswami-Sudan algorithms allow for efficient unique decoding up to half the minimum distance and list-decoding up to the Johnson bound, respectively.

Now, returning our focus to list-decoding, note that the Guruswami-Sudan algorithm allows for list-decoding of codes with rate R up to radius about $1-\sqrt{R}$. The next breakthrough in list-decoding comes from the Parvaresh-Vardy codes [PV05], which allowed for non-trivial list-decoding beyond the Johnson bound. As an example of the achievable parameters, for a small $\varepsilon > 0$ one can construct an explicit $(1-\varepsilon, (1/\varepsilon)^{O(\log\log(1/\varepsilon))})$-list-decodable code of rate $\Omega(\varepsilon \log(1/\varepsilon))$; in contrast, the Johnson bound only guarantees that codes of rate $\Omega(\varepsilon^2)$ can be notrivially list-decoded at radius $1-\varepsilon$.

In 2008, the first capacity-achieving list decodable codes were constructed: Guruswami and Rudra [GR08b] analyzed "Folded"[15] Reed-Solomon codes and showed

[15]Briefly, the folding operation groups together consecutive symbols into the same block symbol. That is, for a folding parameter s dividing n, the word $c = (c_1, \ldots, c_n) \in \Sigma^n$ is mapped to $(c|_{[s]}, c|_{[s+1,2s]}, \ldots, c|_{[n-s+1,n]}) \in (\Sigma^s)^{n/s}$.

they are list-decodable up to the Singleton bound, i.e., up to radius $1-R-\varepsilon$ for any $\varepsilon > 0$. (Recall that for large enough q, the capacity approaches $1-\rho$; see Remark 2.4.8.) Later, it was observed that Derivative codes (also known as univariate multiplicity codes)[16] are list-decodable up to the Singleton bound [GW13; Kop15a].

There are two main drawbacks associated to Folded Reed-Solomon Codes and Derivative Codes. First, they require large alphabets, of order roughly $(n/\varepsilon^2)^{O(1/\varepsilon^2)}$ to list-decode up to radius $1-R-\varepsilon$. Second, the list size guaranteed by these works is also $(n/\varepsilon^2)^{O(1/\varepsilon^2)}$: while polynomial, this is still much larger than the $O(1/\varepsilon)$ list size which suffices for a typical code.

Broadly speaking, to address these issues, two main approaches are used. First, just as AG codes allow for similar performance to Reed-Solomon codes with smaller alphabets, authors have replaced polynomials in the underlying codes with rational functions on other curves [GX12; GX13; GK16]. To address the list size issue, the general strategy is to pass to carefully constructed *subcodes* of either Folded Reed-Solomon codes, Derivative codes, or their AG code variants. This is typically done by constructing pseudorandom objects such as *(hierarchical) subspace evasive sets* [Gur11; DL12; GW13; GX12]. More recently, an improved analysis [Kop+18] of the list sizes of Folded Reed-Solomon and Derivative codes showed that the list sizes are actually independent of n.

The other approach, which is more combinatorial in nature, takes some code with good properties and uses a general purpose distance amplification technique employing expander graphs [AEL95; GI04; HW15; Gop+18; HRZW17; Kop+18; Kop+19]. Colloquially, we term this technique "expander tricks". This is very much in the spirit of the work we present in Chapter 7.

We summarize the state-of-the-art for capacity-achieving codes in Table 2.1. In fact, as all of these codes come with efficient list-recovery algorithms, we state the parameters they achieve when the input lists have size ℓ; to recover the list-decoding guarantee, it suffices to set $\ell = 1$ in the expressions.

2.6 Thesis' Contributions and Organization

With this background established, we are in position to discuss our contributions in more detail. We also describe the thesis' organization and the dependencies between the chapters.

As in Section 1.3, we divide our contributions into three categories.

[16] In these codes, instead of just evaluating a polynomial, one also evalutes derivatives of the polynomial and packs all these data into a single symbol.

Code	Alphabet size q	List size L	Decoding time	Notes
FRS Codes [GR08b; Kop+18]	$\left(\frac{n}{\varepsilon^2}\right)^{O(\log \ell/\varepsilon^2)}$	$\left(\frac{\ell}{\varepsilon}\right)^{O(\log(\ell/\varepsilon)/\varepsilon)}$	$n^{O(\log \ell/\varepsilon)}$	List-size initially $\left(\frac{n}{\varepsilon^2}\right)^{O(\log \ell/\varepsilon^2)}$; improved in [Kop+18].
Derivative Codes [GW13; Kop15a]	$\left(\frac{n}{\varepsilon^2}\right)^{O(\log \ell/\varepsilon^2)}$	$\left(\frac{\ell}{\varepsilon}\right)^{O(\log(\ell/\varepsilon)/\varepsilon)}$	$n^{O(\log \ell/\varepsilon)}$	List-size initially $\left(\frac{n}{\varepsilon^2}\right)^{O(\log \ell/\varepsilon^2)}$; improved in [Kop+18] assuming $d < q$.
FRS subcodes via SES [DL12]	$\left(\frac{n\ell}{\varepsilon^2}\right)^{O(\ell/\varepsilon^2)}$	$\left(\frac{\ell}{\varepsilon}\right)^{(\ell/\varepsilon)}$	$O_{\ell,\varepsilon}(n^2)$	[Gur11] suggests using SES's; [DL12] constructs them .
Folded AG subcodes via SES [GX12]	$\exp\left(\frac{\ell \log(\ell/\varepsilon)}{\varepsilon^2}\right)$	$O\left(\frac{\ell}{\varepsilon}\right)$	$\mathrm{poly}_{\ell,\varepsilon}(n)$	Construction is Monte Carlo.
Folded AG subcodes via SD [GX13; GK16]	$\exp\left(\frac{\ell \log(\ell/\varepsilon)}{\varepsilon^2}\right)$	$2^{2^{2^{O_{\ell,\varepsilon}(\log^* n)}}}$	$\mathrm{poly}(n) \cdot (1/\varepsilon)^{O(\ell)}$	
Tensor of above + expander tricks [HRZW17; Kop+19]	$\exp(\ell/\varepsilon^2)$	$O_{\ell,\varepsilon}(g(n))$, where $g(n) = o(\log^{(c)} n)$ for any $c \in \mathbb{N}$	$O_{\ell,\varepsilon}\left(n^{1+o(1)}\right)$	In [HRZW17], decoding is randomized; in [Kop+19], it's deterministic.
FRS Codes + expander tricks [Kop+18]	$2^{O(\log(\ell)/\varepsilon^6)}$	$O_{\varepsilon,\ell}(1)$	$\mathrm{poly}_{\varepsilon,\ell}(n)$	

Table 2.1: A summary of parameters achieved by explicit constructions of capacity-achieving list-recoverable codes. In the above, $R \in (0,1)$ denotes the rate (which we assume is constant) and ℓ is the input list size. Recall that when $q \geq \exp(\log(\ell)/\varepsilon)$ the capacity is $1-\rho-\varepsilon$, where ρ is the decoding radius. In the above, we abbreviate subspace evasive set as SES and subspace design as SD.

2.6.1 Random Ensembles of Codes

In Chapter 3, we describe a novel framework for understanding properties of random linear codes. We define a broad class of properties which we term *local*, and show that they capture list-decodability and recoverability, along with their average-radius variants. Our main result is a demonstration that every local property experiences a *sharp* threshold. Informally, this means that for every local property, there is a rate R^* such that codes of rate less than R^* almost certainly satisfy the property, while codes of rate larger than R^* almost certainly do not. We also provide a characterization of this rate R^*, which leads to the tantalizing possibility that we could precisely compute this quantity and thereby obtain a perfect understanding of the list-decodability of random linear codes. Obtaining such a computation, alas, remains the subject of ongoing work, although we do provide equivalent formulations that might be more amenable to an effective analysis. The results presented in this chapter are a (substantial) expansion upon results of [Mos+19].

In Chapter 4, we study *low-density parity-check* (LDPC) codes, which are an important subclass of linear codes. Building off the results in Chapter 3, we show that random LDPC codes experience a similar threshold phenomenon at roughly the same rate as random linear codes. As an immediate corollary, since random linear codes achieve list-decoding capacity with high probability, we deduce that the same is true for random LDPC codes. Although there is a large volume of literature devoted to LDPC codes, our result appears to be the first to demonstrate that *any* LDPC code has nontrivial list-decodability (i.e., list-decodability at radii beyond the Johnson bound). Along the way, we also provide a proof that random LDPC codes over fields of size larger than q achieve the GV bound with high probability. The results in this chapter are derived from [Mos+19].

In Chapter 5, we investigate the list-decodability of random linear codes over the *rank metric*. We adapt a proof technique of Guruswami, Håstad and Kopparty [GHK11], which proved random linear codes which are ε-away from capacity are $(\rho, O_{\rho,q}(1/\varepsilon))$-list decodable whp. In this way, we show that random linear rank metric codes which ε-away from capacity are $(\rho, O_{\rho,q}(1/\varepsilon))$-list decodable whp. (In this chapter, we also discuss the list-decoding capacity for rank metric codes.) The proof in this chapter appears in [GR18].

Finally, in Chapter 6, we revisit an argument of Li and Wootters [LW18], which is itself an improvement of an argument of Guruswami, Håstad, Sudan and Zuckerman [Gur+02]. Li and Wootters showed that random linear codes over $\mathbb{F}_2$ that are ε-away from capacity are $(\rho, O(1/\varepsilon))$-list-decodable with high probability. Moreover, their approach applies equally to the Hamming and rank metrics. We show how to modify their argument to obtain the same result for *average-radius* list-decoding. The results in this chapter are currently being prepared for submission.

2.6.2 Explicit Constructions of List-Decodable Codes

While most of our results concern randomized constructions of codes, we also provide improved explicit constructions of capacity-achieving list-decodable (in fact, list-recoverable) codes. Specifically, in Chapter 7, we show how to use the *tensoring* operation (along with certain by-now standard distance amplification techniques) to construct capacity-achieving list-recoverable codes with near-linear[17] time decoding algorithms. Prior works had only established *randomized* near-linear time decoding algorithms; ours is completely deterministic. Furthermore, we provide improved *local* list-recovery algorithms, which informally allow for one to obtain nontrivial information about a single coordinate of a codeword given oracle access to a noisy version of the codeword, and moreover the algorithm runs in *sublinear* time. As a corollary, by concatenating our codes with random linear codes we obtain (non-explicit) binary codes approaching the GV bound with efficient unique decoding algorithms. Finally, we demonstrate that in some sense our analysis is tight; even for zero-error list-recovery, we prove a lower bound on the list size of any high-rate tensor code, which implies certain impossibility results for local list-recovery. The material presented in this chapter comes from [Kop+19].

2.6.3 Applications of List-Decodable Codes

We provide a new application of list-decodable codes to the field of pseudorandomness. Specifically, in Chapter 8, we present an explicit construction of dimension expanders, which are a linear-algebraic analog of expander graphs. An (η, β)-dimension expander of degree d is a collection of d linear maps $\Gamma_j : \mathbb{F}^n \to \mathbb{F}^n$ such that for every subspace $U \leq \mathbb{F}^n$ of degree at most ηn, the image of U under all the maps, $\sum_{j=1}^{d} \Gamma_j(U)$, has dimension at least $\beta \dim(U)$. Over a finite field, a random collection of $d = O(1)$ maps offers excellent "lossless" expansion whp: $\beta \approx d$ for $\eta \geq \Omega(1/d)$. By leveraging techniques developed for list-decoding rank metric codes, we develop a framework for explicitly constructing dimension expanders over finite fields. Our approach yields the following:

- *Lossless* expansion over large fields; more precisely $\beta \geq (1-\varepsilon)d$ and $\eta \geq \frac{1-\varepsilon}{d}$ with $d = O_\varepsilon(1)$, when $|\mathbb{F}| \geq n/d$.
- Optimal up to constant factors expansion over fields of arbitrarily small polynomial size; more precisely $\beta \geq \Omega(\delta d)$ and $\eta \geq \Omega(1/\delta d)$ with $d = O_\delta(1)$, when $|\mathbb{F}| \geq n^\delta$.

This chapter's material first appeared in [GRX18].

[17] That is, $n^{1+o(1)}$.

2.6.4 Dependency Between Chapters

The material presented in Chapter 3 is used quite heavily in Chapter 4, and also provides useful background for Chapters 5 and 6. The material presented in Section 6.3 assumes familiarity with rabj metric codes, particularly Section 5.1.

Chapter 7 is independent of the other chapters and can be read immediately. The same is generally true for Chapter 8; however, comfort with rank metric codes will provide useful intuition for our techniques. Thus, we recommend the reader peruse Section 5.1 prior to reading Chapter 8.

Chapter 3

Combinatorial Properties of Random Linear Codes: A New Toolkit

In terms of the quantity of attention paid to it, random linear codes are certainly the most popular random ensemble of codes. A main focus of this thesis is to understand combinatorial properties possessed by a typical linear code. The combinatorial properties of interest are those we introduced in Section 2.3: list-decodability and its generalizations. In this chapter we develop a suite of tools for understanding what properties we can expect a random linear code of a prescribed rate to possess. Our main contribution is a demonstration that every *local* property (a broad class including list-decoding and its relatives) experiences a sharp threshold: there is a rate R^* such that random linear codes of rate less than R^* almost certainly satisfy the property, whereas random linear codes of rate larger than R^* almost certainly do not. Furthermore, we provide a novel characterization of this threshold R^*.

We begin by reviewing the literature on the list-decodability of random linear codes in Section 3.1. In Section 3.2, we precisely define what we mean by a local property of a code and motivate its definition. The sharp threshold phenomenon experienced by random linear codes is described in Section 3.3, and the main technical argument is provided in Section 3.4. Later, we demonstrate a few uses for our main theorem. First, in Section 3.5, we recover known results on combinatorial properties of random linear codes with what we feel are simpler arguments. Secondly, in Section 3.6, we prove what we believe is a new result on list-of-2 decoding for binary random linear codes: specifically, we precisely pin down the maximum rate R such that a random linear code of rate R is $(\rho, 2)$-average-radius list-decodable with high probability.

Unless otherwise noted, in this chapter we always think of q as a constant, independent of the block length. There has been some work studying random linear codes when q may grow with n; however, the local properties formulation that we introduce in Section 3.2 is only really effective when q is constant.

3.1 Prior Work

Zyablov-Pinsker argument. The first researchers to consider the list-decodability of random linear codes were Zyablov and Pinsker [ZP81]. They demonstrated that a random linear code of rate $1 - h_q(\rho) - \varepsilon$ is $(\rho, q^{O(1/\varepsilon)})$-list-decodable. This demonstrates that there exist linear codes achieving list-decoding capacity (Theorem 2.4.7). The argument is based on the observation that any subset $S \subseteq \mathbb{F}_q^n$ has a linearly indendent subset of size at least $\log_q |S|$. Thus, to show that a linear code $\mathcal{C}$ is (ρ, L)-list-decodable, it suffices to show that $\mathcal{C}$ does not contain $\lceil \log_q(L+1) \rceil$ linearly independent vectors from any Hamming ball of radius ρ, and a simple adaptation of the proof of Theorem 2.4.7 guarantees that this is, with high probability, the case for random linear codes so long as the gap to capacity $\varepsilon > \frac{1}{\log_q(L+1)}$. Furthermore, one can adapt this argument of Zyablov and Pinsker to the setting of list-recovery to deduce that random linear codes of rate $1 - h_{q,\ell}(\rho) - \varepsilon$ are with high probability $(\rho, \ell, q^{O(\ell/\varepsilon)})$-list-recoverable.[1] Thus, we know that random linear codes achieve list-decoding and recovery capacity with high probability; however, the dependence of L on ε is exponentially worse than what is achievable by uniformly random codes. Elias [Eli91] was the first to raise the question of whether lists of size $O(1/\varepsilon)$ are sufficient for a typical linear code; the analogous question for list-recovery asks whether lists of size $O(\ell/\varepsilon)$ suffice. Despite many partial results, some gaps remain in our knowledge, and answering this question in full generality remains an active area of research. [2]

GHSZ and LW: optimal for $q = 2$. The next progress on the list-decodability of random linear codes was made by Guruswami, Håstad, Sudan and Zuckerman [Gur+02]. Via a very slick potential-function based argument (upon which we elucidate further in Chapter 6), they show that with positive probability, a random linear code over $\mathbb{F}_2^n$ of rate $1 - h_2(\rho) - \varepsilon$ is $(\rho, O(1/\varepsilon))$-list-decodable. Later, Li and Wootters [LW18] revisited their techniques and observed that the argument can be adapted to show that the same conclusion holds with high probability. Furthermore, they even determine the constant in the list size: one can take $L \sim h_2(\rho)/\varepsilon$. Thus, we have a complete understanding of the list-decodability of random linear codes over $\mathbb{F}_2$. In Chapter 6, we extend this to *average-radius* list-decoding.

GHK: optimal for $\rho \ll 1 - 1/q$. In light of the above, the remaining task is to comprehend the list-decodability and recoverability of random linear codes when the field size is larger than 2. The first result in this direction was provided by Guruswami, Håstad and Kopparty [GHK11]. Therein it is shown that there exists a constant $C = C_{\rho,q}$

[1] We remark that it is not clear how to adapt this argument for the average-radius variants: even if $\{x_1, \ldots, x_{L+1}\}$ is on average ρ-close to z, it need not be the case that some linearly independent subset is on average ρ-close to z.

[2] We remark that it is known that, with high probability, lists of size $\Omega_{q,\rho}(1/\varepsilon)$ are *required* for list-decoding a random linear code: see, e.g., [GN14, Theorem 20].

such that a random linear code of rate $1 - h_q(\rho) - \varepsilon$ is $(\rho, C/\varepsilon)$-list-decodable with high probability. The argument makes use of a Ramsey-theoretic argument to deduce that sets of vectors have some nice combinatorial structure which can be exploited to bound the number of low-rank subsets of Hamming balls. We will provide more discussion of this technique in Chapter 5; indeed, the material presented in that section is largely an adaptation of the [GHK11] method to the setting of rank metric codes. Unfortunately, the constant C blows up if either $\rho \to 1 - 1/q$ or if $q \to \infty$. Moreover, it is unclear how to generalize the argument to list-recovery, or to average-radius list-decoding.

The high-noise regime. The next series of works attempt to obtain better control of the list size when ρ is close to $1 - 1/q$. Define $\eta := q - q\rho - 1$ (so $\rho = 1 - \frac{1+\eta}{q}$); in this setting, by examining the Taylor expansion of h_q centered at $1 - 1/q$ we obtain the estimate

$$h_q\left(1 - \tfrac{1+\eta}{q}\right) = 1 - \frac{1}{2(q-1)\ln(q)}\eta^2 + O_q(\eta^3)\,. \tag{3.1}$$

More generally, we have the estimate

$$h_{q,\ell}\left(1 - \tfrac{\ell+\eta}{q}\right) = 1 - \frac{1}{2(q-\ell)\ell\ln(q)}\eta^2 + O_q(\eta^3)\,. \tag{3.2}$$

In either case, one can conclude from the capacity theorems (Theorems 2.4.7 and 2.4.12) that there exist codes of rate $\Omega_q(\eta^2)$ which are $(1 - \frac{1+\eta}{q}, O(1/\eta^2))$-list-decodable, or $(1 - \frac{\ell+\eta}{q}, \ell, O(\ell/\eta^2))$-list-recoverable.

The first work to make progress in this regime is by Cheraghchi, Guruswami and Velinker [CGV13]. Therein, it is shown a random linear code of rate

$$\Omega\left(\frac{\eta^2}{q^2\ln^3(q/\eta)\ln^4(q)}\right)$$

is $\left(1 - \frac{1+\eta}{q}, O(1/\eta^2)\right)$-average-radius list-decodable with probability 0.99. Wootters [Woo13] improved their argument to show that the same conclusion holds with probability $1 - o(1)$ for random linear codes of rate

$$\Omega\left(\frac{\eta^2}{q^2\ln(q)}\right)\,.$$

Later, Rudra and Wootters [RW14] managed to show (amongst other things) that the same conclusion holds when the rate is

$$\Omega\left(\frac{\eta^2}{q\ln(q)\ln^5(1/\eta)}\right)\,,$$

although the success probability is again only constant.

Source	Radius	Rate	List Size	Notes
[GHK11]	ρ	$1 - h_q(\rho) - \varepsilon$	$C_{\rho,q}/\varepsilon$	Constant $C_{\rho,q} \to \infty$ as $\rho \to 1 - 1/q$ or $q \to \infty$.
[Woo13; RW14]	$1 - \frac{1+\eta}{q}$	$\Omega\left(\frac{\eta^2}{q\ln(q)} \cdot \max\left\{\frac{1}{q}, \frac{1}{\ln^5(1/\eta)}\right\}\right)$	$O(1/\eta^2)$	Rate should be $\Theta\left(\frac{\eta^2}{q\ln(q)}\right)$.

Table 3.1: Brief snapshot of state-of-the-art for list-decoding. The first result is effective in the constant-noise regime; the latter in the high-noise regime.

The first two of these works use a simplex encoding and thereby obtain a problem concerning random vectors in complex vector space. [CGV13] observes that it is sufficient to prove that a certain matrix is a *restricted isometry*; [Woo13] observes that a simpler condition is sufficient. In both cases, the arguments boil down to analyzing a certain *gaussian process*, i.e., they bound the maximum of a set of random gaussian vectors; the standard technique used for these problems is a *chaining argument*. In [RW14], in order to obtain a better dependence on q,[3] a chaining argument was directly applied to random vectors over $\mathbb{F}_q$ which appear naturally in constructing a random linear code.

A structure vs. pseudorandomness approach. The final work we wish to highlight is again by Rudra and Wootters [RW18]. In this work, a "structure vs. pseudorandomness" argument is used to directly study the average-radius list-recoverability of random linear codes. Their results are quite general and apply in many different parameter regimes; however, none of the results exactly match the results attainable by uniformly random codes, and are often incomparable to previous results. As an example, for sufficiently large alphabets the authors are able to show that codes of rate $0.99(1 - h_{q/\ell}(1 - \ell/q - \eta) - \log_q(\ell))$ are $(1 - \ell/q - \eta, \ell, q^{O(\ln^2(\ell/\eta))})$-average-radius list-recoverable; that is, the rate is closer to optimal[4] and the list size is quasi-polynomial in ℓ and η.

For a mostly comprehensive summary of this literature review, see Table 3.2; a brief snapshot of the state of the art is provided in Table 3.1. In this thesis, we extend this literature in multiple ways. Firstly, in this chapter, we introduce a novel framework for understanding combinatorial properties of random linear codes. We now turn to the development of this framework.

[3]They were particularly interested in the case when $q \geq n$, as then their results would apply to random puncturings of Reed-Solomon codes.

[4]The quantity $1 - h_{q/\ell}(1 - \ell/q - \eta) - \log_q(\ell)$ is slightly smaller than $1 - h_{q,\ell}(1 - \ell/q - \eta)$.

Source	Radius	List-Recovery?	Rate	List Size	Average-radius?
[ZP81]	ρ	✓	$1 - h_{q,\ell}(\rho) - \varepsilon$	$q^{\ell/\varepsilon}$	✗
[LW18]	ρ	✗	$1 - h_2(\rho) - \varepsilon$	$h_2(\rho)/\varepsilon$	✗
[GHK11]	ρ	✗	$1 - h_q(\rho) - \varepsilon$	$O_{\rho,q}(1/\varepsilon)$	✗
[CGV13]	$1 - \frac{1+\eta}{q}$	✗	$\Omega\left(\frac{\eta^2}{q^2 \ln^3(q/\eta) \ln^4(q)}\right)$	$O(1/\eta^2)$	✓
[Woo13]	$1 - \frac{1+\eta}{q}$	✗	$\Omega\left(\frac{\eta^2}{q^2 \ln(q)}\right)$	$O(1/\eta^2)$	✓
[RW14]	$1 - \frac{1+\eta}{q}$	✗	$\Omega\left(\frac{\eta^2}{q \ln(q) \ln^5(1/\eta)}\right)$	$O(1/\eta^2)$	✓
[RW18]	$\alpha = 1 - \frac{\ell}{q} - \eta$	✓	$0.99(1 - h_{q/\ell}(\alpha) - \log_q(\ell))$	$q^{O(\ln^2(\ell/\eta))}$	✓

Table 3.2: A summary of state-of-the arts results concerning combinatorial properties of random linear codes. The [LW18] result builds off [Gur+02] and only applies when $q = 2$.

3.2 Local Properties of Codes

One contribution of this thesis is the development of a theory that characterizes the sorts of subsets that one can expect to lie in a random linear code. That is, suppose $\mathcal{C}$ is a random linear code of rate R, and $\mathcal{B}$ is a collection of subsets of $\mathbb{F}_q^n$. As R increases, it will become only more and more likely that a subset $B \in \mathcal{B}$ will lie in the code. The hope is to nail down the value of R for which a random linear code goes from almost certainly not containing a subset of $\mathcal{B}$ to almost certainly containing a subset.

To properly develop this theory, we are inspired by the literature devoted to local properties of random graphs: that is, a property of a graph that is characterized by the graph containing, or not containing, a certain constant-sized subgraph H.[5] Note that such a property is invariant with respect to permutations on the vertices.

Now, consider a graph drawn from the Erdős-Rényi model $G(n, p)$. As p increases, such a graph will be more-and-more likely to satisfy the property. It is well-known (see, e.g., [Bol01, Sec. 4.2]) that for any constant-sized graph H there is some threshold p_0^H such that the expected number of appearances of H as a subgraph of $G(n, p)$ either tends to 0 or ∞ depending on whether p is smaller or larger than p_0^H, and moreover if $p > p_0^H$ then H will indeed appear as a subgraph of $G(n, p)$ with high probability.[6]

We now seek a reasonable definition for a local property of a linear code. Guided by the corresponding definition for graphs, this definition should be (i) defined by constant sized subcodes of a code, and (ii) invariant with respect to permutations of the coordinates. Our definition is inspired by the theory of types developed in information

[5]There is also interest in graphs containing vertex-induced subgraphs; however, for the analogy with random linear codes we wish to draw, it is best to think of edge-induced subgraphs.

[6]Note that this does not follow trivially from the assertion that the expected number of appearances of H is $\omega(1)$: it could be that there are many copies of H with small probability and 0 copies with large probability.

theory [CS+04; CT12].

3.2.1 Definitions

A *(length n code) property* $\mathcal{P}$ is simply a set of linear codes in $\mathbb{F}_q^n$. We say a code $\mathcal{C}$ *satisfies* the property $\mathcal{P}$ if $\mathcal{C} \in \mathcal{P}$. We will only be concerned with *monotone decreasing* properties, i.e., properties for which $\mathcal{C} \in \mathcal{P}$ and $\mathcal{D} \leq \mathcal{C}$ imply $\mathcal{D} \in \mathcal{P}$. Furthermore, we assume properties are *nontrivial*, which means $\{0\} \in \mathcal{P}$. We will typically be concerned with *property families* $\mathcal{P} = (\mathcal{P}_{n_i})_{i \in \mathbb{N}}$, where each $\mathcal{P}_{n_i}$ is a length n_i code property and $n_1 < n_2 < \ldots$ is an increasing sequence of integers.

Informally, a *local property* of a code is a property that can be defined by the exclusion of certain constant-sized sets. This is hopefully reminiscent the property of a graph being H-free. However, we find it more convenient think of the excluded sets being defined in terms of *types*, which we introduce next.

Types. Types are a basic object of study in information theory; see, e.g., [CT12, Chapter 11] or [CS+04]. We provide the definition specialized to our situation.

Definition 3.2.1. Let q be a prime power and $\ell, n \in \mathbb{N}$ with $\ell < n$. A *type over $\mathbb{F}_q^\ell$ with denominator n* is a distribution $\tau \sim \mathbb{F}_q^\ell$ for which $\tau(u) \in \{0, \frac{1}{n}, \frac{2}{n}, \ldots, \frac{n-1}{n}, 1\}$. When $\mathbb{F}_q^\ell$ or n are clear from context, we will simply refer to τ as a *type*.

Next, we define the rank of a type.

Definition 3.2.2 (Rank of a Type)**.** Let τ be a type. The *rank of a type* is

$$\operatorname{rank}(\tau) := \operatorname{rank}(\operatorname{supp}(\tau)) \, .$$

Let $\mathcal{D}_{n,\ell}$ denote the set of all possible types over $\mathbb{F}_q^\ell$ with denominator n. Note that types are in one-to-one correpondence with partitions of $[n]$ into q^ℓ sets, and so we have

$$|\mathcal{D}_{n,\ell}| \leq \binom{n + q^\ell - 1}{q^\ell - 1} \leq (n+1)^{q^\ell} \, . \tag{3.3}$$

In particular, note that for constant q and ℓ this quantity is *polynomial* in n.

Next, we associate a type to a matrix in $\mathbb{F}_q^{n \times \ell}$ as follows.

Definition 3.2.3 (Type of a Matrix)**.** Let q be a prime power and $\ell, n \in \mathbb{N}$ with $\ell < n$. Let $M \in \mathbb{F}_q^{n \times \ell}$. The *type of* M, denoted τ_M, is the probability distribution of a uniformly sampled row of M. That is,

$$\tau_M(u) = \frac{\#\{i \in [n] : M_{i,*} = u\}}{n} \, .$$

Note that the support of τ is precisely the set of M's rows, and ergo $\operatorname{rank}(\tau_M) = \operatorname{rank}(M)$. Denote by $\mathcal{M}_\tau$ the set of matrices in $\mathbb{F}_q^{n \times \ell}$ that have row distribution τ; we refer to this set as the *type class* of τ.

Observe that there are $|\mathbb{F}_q^{n\times\ell}| = q^{n\ell}$ total matrices, which is exponentially large in n. However, as observed previously, there are only polynomially many types. Thus, at least one of the $\mathcal{M}_\tau$'s must have exponential size. In fact, we have the following identity: if $u_1, \ldots, u_{q^\ell}$ is an enumeration of $\mathbb{F}_q^\ell$,

$$|\mathcal{M}_\tau| = \binom{n}{n\tau(u_1), n\tau(u_2), \ldots, n\tau(u_{q^\ell})}.$$

This expression is quite unwieldy in practice though. Fortunately, we have the following well-known estimate. Recall that $H_q(\tau) = H(\tau)/\log(q)$ denotes the base q entropy of τ.

Proposition 3.2.4 ([CS+04, Lemma 2.2], [CT12, Theorem 11.1.3])**.** *Let τ be a type over $\mathbb{F}_q^\ell$ with denominator n. Then*

$$\binom{n+q^\ell-1}{q^\ell-1}^{-1} q^{nH_q(\tau)} \le |\mathcal{M}_\tau| \le q^{nH_q(\tau)}.$$

That is,

$$\log_q |\mathcal{M}_\tau| = (1-o(1))H_q(\tau)\cdot n.$$

3.2.2 Local Properties

Having established the definition of a type, we are able to discuss local properties of codes. Let $\mathcal{C} \le \mathbb{F}_q^n$ be a code and τ a type. We say that $\mathcal{C}$ *contains the type* τ, written $\tau \in \mathcal{C}$, if there exists a matrix $M \in \mathcal{M}_\tau$ such that $\text{col-span}(M) \subset \mathcal{C}$, which we denote slightly abusively by "$M \subseteq \mathcal{C}$".

Definition 3.2.5. Let $\ell \in \mathbb{N}$. An *ℓ-local property* is a property defined by excluding a set of types $\mathcal{T} \subseteq \bigcup_{1\le\ell'\le\ell} \mathcal{D}_{n,\ell'}$:

$$\{\mathcal{C} \le \mathbb{F}_q^n : \forall \tau \in \mathcal{T}, \tau \notin \mathcal{C}\}.$$

We refer to this property as *$\mathcal{T}$-freeness* and denote it by $\mathcal{P}^{\mathcal{T}}$. We shorthand $\mathcal{P}^\tau := \mathcal{P}^{\{\tau\}}$.

For a family of properties $\mathcal{P} = (\mathcal{P}_{n_i})_{i\in\mathbb{N}}$, $\mathcal{P}$ is called an *ℓ-local property* if $\mathcal{P}_{n_i}$ is an ℓ-local property for each $i \in \mathbb{N}$.

Remark 3.2.6. It is natural to assume that the family of properties is defined in a "uniform" manner. As a first example, observe that if we have an ℓ-local type τ with denominator n, one can naturally view it as an ℓ-local type with denominator m whenever $n|m$. Thus, if we fix some type $\tau \in \mathcal{D}_{n_1,\ell}$ for integers n_1 and ℓ, we can consider the family of properties of τ-freeness for all block-lengths $n_i := i \cdot n_1$ for $i \in \mathbb{N}$.

More generally, when we describe popular code properties (such as list-decodability) as a local property, the description will be uniform in the following sense: for each n, we will take the set of all types $\tau \in \mathcal{D}_{n,\ell}$ for which the vector $(\tau(u))_{u\in\mathbb{F}_q^\ell} \in \mathbb{R}^{q^\ell}$ satisfies some finite, fixed set of linear inequalities. Notably, the set of linear inequalities will not depend on n.

Nonetheless, our theorem will be general enough to apply to local properties that are not uniform in any sense. That is, sequence of forbidden types $(\mathcal{T}_{n_i})_{i\in\mathbb{N}}$ need not be "consistent" in any meaningful sense. In Chapter 9, we discuss potential improvements to our results if we assume the property family is uniform.

Intuitively, as the rate R of a random linear code $\boldsymbol{\mathcal{C}}$ increases (equivalently, as the number of rows in the random parity-check matrix $\boldsymbol{H}$ decreases), it will become increasingly unlikely that $\boldsymbol{\mathcal{C}}$ will be $\mathcal{T}$-free. This can be proved formally via a standard coupling argument, akin to [Bol01, Theorem 2.1]; see Remark 3.3.3.

We will demonstrate that any such local property family experiences a *sharp threshold*. The formal definition is given later (Definition 3.3.4), but the intuition is that there is a fixed $R^* \in [0,1]$ such that codes of rate less than R^* almost certainly satisfy the property whereas codes of rate greater than R^* almost certainly do not.

Before proceeding, we demonstrate that many widely studied properties of codes are local properties in this sense.

Example 3.2.7 (Distance is a 1-Local Property)**.** Consider the property of a code having distance greater than δ. Let $\mathcal{T} \subseteq \mathcal{D}_{n,1}$ be the set of all types $\tau \sim \mathbb{F}_q^1$ for which $1 > \tau(0) \geq 1-\delta$. Then a code $\mathcal{C}$ is $\mathcal{T}$-free if and only if it has distance greater than δ.

Example 3.2.8 (List-Decodability is an $(L+1)$-Local Property)**.** Consider the property of a code being (ρ, L)-list-decodable. Let $\mathcal{T} \subseteq \mathcal{D}_{n,L+1}$ be the set of all types τ such that for some (correlated) type $\tau' \in \mathcal{D}_{n,1}$,

$$\forall i \in [L+1], \quad \mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{x})\sim(\tau,\tau')}(\boldsymbol{u}_i \neq \boldsymbol{x}) \leq \rho\,. \tag{3.4}$$

We furthermore stipulate

$$\forall i \neq j \in [L+1], \quad \mathop{\mathbb{P}}_{\boldsymbol{u}\sim\tau}(\boldsymbol{u}_i \neq \boldsymbol{u}_j) < 1\,. \tag{3.5}$$

We claim that a code $\mathcal{C}$ is (ρ, L)-list-decodable if and only if $\mathcal{C}$ is $\mathcal{T}_1$-free.

We briefly provide the justification. $\mathcal{C}$ fails to be (ρ, L)-list-decodable iff there exists a center $z \in \mathbb{F}_q^n$ and a set of $L+1$ distinct codewords $\{c^1, \ldots, c^{L+1}\} \in B(z,\rho) \cap \mathcal{C}$. Let $M \in \mathbb{F}_q^{n\times(L+1)}$ be the matrix whose columns are given by $c^1, \ldots, c^{L+1}$ (in this order) and enumerate its rows $u^1, \ldots, u^n \in \mathbb{F}_q^{L+1}$. (Thus, $u_i^j = c_j^i$ for $i \in [L+1]$ and $j \in [n]$.) Define the pair (τ, τ') by sampling $\boldsymbol{j} \sim [n]$ uniformly and outputting $(u^{\boldsymbol{j}}, z_{\boldsymbol{j}}) \in \mathbb{F}_q^{L+1} \times \mathbb{F}_q$. As $\mathcal{C}$ contains $\mathrm{col\text{-}span}(M)$, $\mathcal{C}$ contains a matrix in $\mathcal{M}_\tau$, i.e., $\mathcal{C}$ contains τ. Moreover note that Condition 3.4 holds for (τ, τ') as $c^1, \ldots, c^{L+1} \in B(z,\rho)$. Indeed, for any $i \in [L+1]$,

$$\begin{aligned}\mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{x})\sim(\tau,\tau')}(\boldsymbol{u}_i \neq \boldsymbol{x}) &= \mathop{\mathbb{P}}_{\boldsymbol{j}\sim[n]}\left(u_i^{\boldsymbol{j}} \neq z_{\boldsymbol{j}}\right) = \frac{1}{n}\sum_{j=1}^{n}\mathbb{I}\left(u_i^j \neq z_j\right)\\ &= \frac{1}{n}\sum_{j=1}^{n}\mathbb{I}\left(c_j^i \neq z_j\right) = d(c^i, z_j) \leq \rho\,.\end{aligned}$$

Furthermore, the fact that the codewords are distinct guarantees that Condition 3.5 is satisfied. The converse can be proved in the analogous way (using the assumption that the type τ has denominator n).

If one is interested in average-radius list-decodability, one can replace Condition 3.4 by

$$\frac{1}{L+1}\sum_{i=1}^{L+1}\mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{x})\sim(\tau,\tau')}(\boldsymbol{u}_i \neq \boldsymbol{x}) \leq \rho\,. \tag{3.6}$$

Example 3.2.9 (List-Recovery is an $(L+1)$-Local Property)**.** Generalizing the previous example, we can consider the property of a code being (ρ,ℓ,L)-list-recoverable. Let $\mathcal{T} \subseteq \mathcal{D}_{n,L+1}$ be the set of all types τ such that for some type $\tau' \in \mathcal{D}_{n,\ell}$,

$$\forall i \in [L+1],\quad \mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{z})\sim(\tau,\tau')}(\forall j \in [\ell],\ \boldsymbol{u}_i \neq \boldsymbol{x}_j) \leq \rho\,, \tag{3.7}$$

and moreover the type τ is required to satisfy Condition 3.5. It follows that $\mathcal{C}$ is (ρ,ℓ,L)-list-recoverable if and only if $\mathcal{C}$ is $\mathcal{T}$-free. The justification is analogous to that given for the previous example and we omit it. If one is concerned with average-radius list-recoverability, one may replace Condition 3.7 by

$$\frac{1}{L+1}\sum_{i=1}^{L+1}\mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{x})\sim(\tau,\tau')}(\forall j \in [\ell],\ \boldsymbol{u}_i \neq \boldsymbol{x}_j) \leq \rho\,. \tag{3.8}$$

Thus, the previous examples demonstrate that local properties capture most interesting properties of linear codes. The following proposition generalizes these examples. In brief, it states that local properties are precisely those properties defined by excluding a family of "bad" subsets, provided those subsets are closed under coordinate permutations. To state the proposition, we use the following notation and terminology. For a permutation $\pi \in S_n$ and a string $x \in \Sigma^n$, $\pi(x)$ denotes the string obtained by permuting the coordinates according to π, i.e., $\pi(x) := (x_{\pi(1)},\ldots,x_{\pi(n)})$. For a subset $B \subseteq \Sigma^n$, $\pi(B) := \{\pi(x) : x \in B\}$. A family of subsets $\mathcal{B} \subseteq 2^{\Sigma^n}$ is called *permutation-invariant* if for all $B \in \mathcal{B}$ and $\pi \in S_n$, $\pi(B) \in \mathcal{B}$. The family $\mathcal{B}$ is furthermore deemed *ℓ-bounded* if $|B| \leq \ell$ for all $B \in \mathcal{B}$.

Proposition 3.2.10 (Characterization of Local Properties)**.** *Let $\mathcal{C}$ be a linear code. For any ℓ-local property $\mathcal{P}$, there exists an ℓ-bounded permutation-invariant family $\mathcal{B} \in 2^{\Sigma^n}$ such that $\mathcal{C}$ satisfies $\mathcal{P}$ if and only if for all $B \in \mathcal{B}$, $B \not\subseteq \mathcal{C}$.*

Conversely, given any ℓ-bounded permutation-invariant family $\mathcal{B}$, there is an ℓ-local type $\mathcal{P}$ such that $\mathcal{C}$ satisfies $\mathcal{P}$ if and only if for all $B \in \mathcal{B}$, $B \not\subseteq \mathcal{C}$.

Proof. For the forward implification, take

$$\mathcal{B} = \bigcup_{\tau\in\mathcal{T}}\bigcup_{M\in M_\tau}\mathrm{cols}(M)\,,$$

where $\mathrm{cols}(M)$ denotes the columns of the matrix.

For the other direction, for each $B \in \mathcal{B}$, take any matrix $M \in \mathbb{F}_q^{n \times |B|}$ with $\mathrm{cols}(M) = B$. By permutation-invariance of $\mathcal{B}$, every $M' \in \mathcal{M}_{\tau_M}$ has $\mathrm{cols}(M') = B'$ for some $B' \in \mathcal{B}$. Thus, we take $\mathcal{T}$ to be the union of all the $\mathcal{M}_{\tau_M}$'s obtained in this manner. □

3.3 Characterizing the Threshold of Local Properties

In this section we demonstrate that every local property of a random linear code experiences a sharp threshold. Moreover, we provide a characterization of the threshold in terms of an explicitly computable quantity. First, we define what we mean by a threshold for a monotone property. In the following, for $n \in \mathbb{N}$ and $R \in [0,1]$, $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R)$ denotes a random linear code of rate R.

Definition 3.3.1 (Threshold of a Property)**.** Let $\mathcal{P}$ be a nontrivial property of length n codes. The *threshold* of $\mathcal{P}$ is defined to be

$$R_{\mathrm{RLC}}(\mathcal{P}) := \sup\{R \in [0,1] : \mathbb{P}\left(\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R) \text{ satisfies } \mathcal{P}\right) \geq 1/2\} .$$

Remark 3.3.2. As $\mathcal{P}$ is nontrivial (i.e., $\{0\} \in \mathcal{P}$), $R_{\mathrm{RLC}}(\mathcal{P}) \geq 0$.

Remark 3.3.3. Note that every property has a threshold in the sense of Definition 3.3.1. This is not entirely obvious, as one must justify that $\mathbb{P}\left(\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R) \text{ satisfies } \mathcal{P}\right)$ is decreasing in R. To do this, a standard coupling argument ([Bol01, Theorem 2.1]) works: if $R_1 < R_2$ then one can imagine sampling $n - R_2 n$ vectors at random and setting $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R_2)$ to be the orthogonal complement of these vectors, and then sampling an additional $(R_2 - R_1)n$ vectors and taking the orthogonal complement of all the sampled vectors to obtain $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R_1)$. Both $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R_1)$ and $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R_2)$ have the correct distribution, and $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R_1)$ is only more likely to satisfy $\mathcal{P}$ than $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R_2)$.

To say that a property experiences a *sharp* threshold is to say that there is a small interval of rates over which a random linear code goes from almost certainly satisfying the property to almost certainly not satisfying the property. To define sharpness formally, we should speak of a family of properties for an increasing sequence of blocklengths $n_1 < n_2 < \dots$. The following definition makes this notion precise. In this chapter, $o(1)$ always denotes a quantity $f(n)$ for which $\lim_{i\to\infty} f(n_i) = 0$.

Definition 3.3.4 (Sharp Threshold of a Property Family)**.** Let $\mathcal{P} = (\mathcal{P}_{n_i})_{i\in\mathbb{N}}$ be a property family. The property family $\mathcal{P}$ is said to be *sharp for random linear codes* if for any $\varepsilon > 0$ the following holds:

- if $R_{n_i} \leq R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) - \varepsilon$, a length n_i random linear code of rate R_{n_i} satisfies $\mathcal{P}_{n_i}$ with probability $1 - o(1)$;
- for any $R_{n_i} \geq R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) + \varepsilon$, a length n_i random linear code of rate R_{n_i} satisfies $\mathcal{P}_{n_i}$ with probability $o(1)$.

We now specialize our discussion to local properties. Our main result in this chapter is a proof that *every* local property is sharp for random linear codes. This will follow from Theorem 3.3.9 which additionally characterizes the sequence $(R_{\mathrm{RLC}}(\mathcal{P}_{n_i}))_{i\in\mathbb{N}}$.

Given a type τ with denominator n and $M \in \mathcal{M}_\tau$, a random linear code $\boldsymbol{\mathcal{C}}$ of rate R contains M with probability $q^{-n(1-R)\operatorname{rank}(M)} = q^{-n(1-R)\operatorname{rank}(\tau)}$; see Proposition 2.2.2. Hence, in expectation, $\boldsymbol{\mathcal{C}}$ contains roughly $q^{n(H_q(\tau)-(1-R)\operatorname{rank}(\tau))}$ matrices from $\mathcal{M}_\tau$. In particular, this expectation grows (resp. decays) exponentially in n when R is larger (resp. smaller) than $1 - \frac{H_q(\tau)}{\operatorname{rank}(\tau)}$. This motivates the following definition.

Definition 3.3.5 (Expectation Threshold)**.** Given a distribution τ over $\mathbb{F}_q^\ell$, let

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) := 1 - \frac{H_q(\tau)}{\operatorname{rank}(\tau)} \, .$$

It follows from a standard first-moment argument that if $R < R^{\mathbb{E}}_{\mathrm{RLC}}(\tau)$ then $\boldsymbol{\mathcal{C}}$ satisfies $\mathcal{P}^\tau$ with probability $1 - \exp(-\Omega(n))$. In particular, as n grows we get the lower bound

$$R_{\mathrm{RLC}}(\mathcal{P}^\tau) \geq R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) - o(1) \, . \tag{3.9}$$

However, as the following example shows, this bound is not tight.

Example 3.3.6. Let $q = 2$, $\ell = 3$ and consider the distribution τ over $\mathbb{F}_2^3$ given by the following table:

u	$\tau(u)$
$(1,0,0)$	$1/4$
$(0,1,0)$	$1/4$
$(1,0,1)$	$1/4$
$(0,1,1)$	$1/4$
Every other vector	0

Note that such a type may be viewed as having denominator n for any n divisible by 4. It is straightforward to compute $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) = 1 - \frac{H_2(\tau)}{\operatorname{rank}(\tau)} = 1 - \frac{2}{3} = \frac{1}{3}$.

We claim that $R_{\mathrm{RLC}}(\mathcal{P}^\tau)$ is strictly larger than $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau)$. Let $A := \left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix}\right) \in \mathbb{F}_2^{2\times 3}$ represent the linear map which projects a vector onto its first two coordinates. Let τ' denote the distribution of $A\boldsymbol{u}$, where $\boldsymbol{u}$ is a random vector sampled from τ. Thus, τ' is distributed as follows:

u	$\tau'(u)$
$(1,0)$	$1/2$
$(0,1)$	$1/2$
Every other vector	0

Note that a code $\mathcal{C}$ which contains a matrix M from $\mathcal{M}_\tau$ must contain the first two columns of M: that is, the matrix MA^T. Consequently, every code which satisfies $\mathcal{P}^{\tau'}$ also satisfies $\mathcal{P}^\tau$, and so $R_{\mathrm{RLC}}(\mathcal{P}^\tau) \geq R_{\mathrm{RLC}}(\mathcal{P}^{\tau'})$.

Finally, (3.9) yields

$$R_{\mathrm{RLC}}(\mathcal{P}^{\tau'}) \geq R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') - o(1) = 1 - \frac{H_2(\tau')}{\operatorname{rank}(\tau')} - o(1) = 1 - \frac{1}{2} - o(1) = \frac{1}{2} - o(1) \, ,$$

and we conclude that, for sufficiently large n,

$$R_{\mathrm{RLC}}(\mathcal{P}^\tau) \geq \frac{1}{2} - o(1) > \frac{1}{3} = R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) \,.$$

Example 3.3.6 motivates the following definition.

Definition 3.3.7 (Implied Type)**.** Let $\tau \in \mathcal{D}_{n,\ell}$ and let $A \in \mathbb{F}_q^{m\times\ell}$ be a full-rank matrix for some $m \leq \ell$. The type τ' of the random vector $A\boldsymbol{u}$, where $\boldsymbol{u}$ is sampled according to τ, is said to be τ*-implied*. Note that $\tau' \in \mathcal{D}_{n,m}$, i.e., it is an m-local type with denominator n. We denote the set of τ-implied distributions by $\mathcal{I}_\tau$.

Note that if $\tau \in \mathcal{C}$, then $\tau' \in \mathcal{C}$ for any $\tau' \in \mathcal{I}_\tau$. Indeed, suppose that $\mathcal{C}$ contains a matrix $M \in \mathcal{M}_\tau$. As all the columns of MA^T lie in $\text{col-span}(M)$ (where $A \in \mathbb{F}_q^{m\times\ell}$ is as in Definition 3.3.7), the linearity of $\mathcal{C}$ guarantees that it also contains the matrix MA^T, which belongs to $\mathcal{M}_{\tau'}$. That is to say, $\tau' \in \mathcal{C}$. This justifies the terminology: an appearance of the type τ' is "implied" by the appearance of the type τ.

Stated in terms of the contrapositive, this amounts to saying that a linear code satisfying $\mathcal{P}^{\tau'}$ must also satisfy $\mathcal{P}^\tau$. Consequently, $R_{\mathrm{RLC}}(\mathcal{P}^\tau) \geq R_{\mathrm{RLC}}(\mathcal{P}^{\tau'})$. Combining these observations with Inequality (3.9) implies the stronger lower bound

$$R_{\mathrm{RLC}}(\mathcal{P}^\tau) \geq \max_{\tau'\in\mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') - o(1) \,. \tag{3.10}$$

Lemma 3.3.8 essentially says that (3.10) is tight, and that $\mathcal{P}^\tau$ is sharp for random linear codes.

Lemma 3.3.8. *Let $\ell \in \mathbb{N}$ and $\tau \in \mathcal{D}_{n,\ell}$. Denote $R^*_\tau = \max_{\tau'\in\mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau')$. The threshold $R_{\mathrm{RLC}}(\mathcal{P}^\tau)$ satisfies*

$$R_{\mathrm{RLC}}(\mathcal{P}^\tau) = R^*_\tau \pm o(1) \,.$$

Furthermore, suppose $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ is a random linear code of rate R.

1. *If $R \leq R^*_\tau - \varepsilon$ then*
$$\mathbb{P}\left(\tau \in \boldsymbol{\mathcal{C}}\right) \leq q^{-\varepsilon n} \,.$$
2. *Conversely, if $R \geq R^*_\tau + \varepsilon$ then*
$$\mathbb{P}\left(\tau \in \boldsymbol{\mathcal{C}}\right) \geq 1 - \binom{n + q^{2\ell} - 1}{q^{2\ell} - 1}^3 \cdot q^{-\varepsilon n} \,.$$

We defer the proof of Lemma 3.3.8 to Section 3.4. Assuming this result, we are able to conclude that local property families have a sharp threshold in the sense of Definition 3.3.4.

Theorem 3.3.9. *[Sharpness of Local Properties for Random Linear Codes] Fix $\ell \in \mathbb{N}$. Any ℓ-local property family $\mathcal{P} = (\mathcal{P}_{n_i})_{i\in\mathbb{N}}$ is sharp for random linear codes.*

Futhermore, if $\mathcal{T}_{n_i} \subseteq \mathcal{D}_{n,\ell}$ is a set of types such that $\mathcal{P}_{n_i} = \mathcal{P}^{\mathcal{T}_{n_i}}$, then

$$R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) = \min_{\tau\in\mathcal{T}_{n_i}} \max_{\tau'\in\mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \pm o(1) \,. \tag{3.11}$$

Proof. For $i \in \mathbb{N}$, denote

$$R^*_{n_i} = \min_{\tau \in \mathcal{T}_{n_i}} \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \, .$$

For $R \in [0,1]$ let $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R)$ denote a random linear code in $\mathbb{F}_q^n$ of rate R. To prove the theorem, it suffices to prove the following:

I. For any $\varepsilon > 0$, if $R_{n_i} \leq R^*_{n_i} - \varepsilon$ for all $i \in \mathbb{N}$, $\lim_{i\to\infty} \mathbb{P}\left(\boldsymbol{\mathcal{C}}^{n_i}_{\mathrm{RLC}}(R_{n_i}) \text{ satisfies } \mathcal{P}_{n_i}\right) = 1$.

II. For any $\varepsilon > 0$, if $R_{n_i} \geq R^*_{n_i} + \varepsilon$ for all $i \in \mathbb{N}$, $\lim_{i\to\infty} \mathbb{P}\left(\boldsymbol{\mathcal{C}}^{n_i}_{\mathrm{RLC}}(R_{n_i}) \text{ satisfies } \mathcal{P}_{n_i}\right) = 0$.

To prove Statement I, observe that Item 1 of Lemma 3.3.8 guarantees that for each $\tau \in \mathcal{T}_{n_i}$, $\mathbb{P}\left(\tau \in \boldsymbol{\mathcal{C}}^{n_i}_{\mathrm{RLC}}(R_{n_i})\right) \leq q^{-\varepsilon n_i}$. Note that

$$|\mathcal{T}_{n_i}| \leq |\mathcal{D}_{n_i,\ell}| \leq \binom{n_i + q^\ell - 1}{q^\ell - 1} \leq (n_i+1)^{q^\ell} \, ,$$

where we have recalled Eq. (3.3). Thus, by taking a union bound over all $\tau \in \mathcal{T}_{n_i}$, we find

$$\mathbb{P}\left(\boldsymbol{\mathcal{C}}^{n_i}_{\mathrm{RLC}}(R_{n_i}) \text{ satisfies } \mathcal{P}_{n_i}\right) \leq (n_i+1)^{q^\ell} q^{-\varepsilon n_i} \overset{i\to\infty}{\longrightarrow} 0 \, .$$

For Statement II, take any $\tau \in \mathcal{T}_{n_i}$ such that $\max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\mathcal{P}^{\tau'}) = R^*_{n_i}$. By Item 2 of Lemma 3.3.8, $\boldsymbol{\mathcal{C}}^{n_i}_{\mathrm{RLC}}(R_{n_i})$ almost surely contains τ, which is a sufficient condition for $\boldsymbol{\mathcal{C}}^{n_i}_{\mathrm{RLC}}(R_{n_i})$ to not satisfy $\mathcal{P}_{n_i}$. □

Remark 3.3.10. Note that this theorem actually promises that the $o(1)$ terms in Definition 3.3.4 are of the form $\exp(-\Omega(\varepsilon n))$, i.e., they are exponentially small in n.

Remark 3.3.11. Returning to the setting of Remark 3.2.6, suppose we fix a type $\tau \in \mathcal{D}_{n_1,\ell}$ and that the property family $(\mathcal{P}_{n_i})_{i\in\mathbb{N}}$ is defined uniformly as the property of τ-freeness, where τ is viewed as an ℓ-local property with denominator $i \cdot n_1$ for any $i \in \mathbb{N}$. Note that the quantity $R^*_\tau = \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau')$ is independent of n_i. Thus, we see that the sequence of thresholds $(R_{\mathrm{RLC}}(\mathcal{P}_{n_i}))$ converges to a fixed value; namely, R^*_τ.

More generally, if we have a family of local properties $(\mathcal{P}_{n_i})$ defined uniformly via a set of linear inequalities (as is the case for, e.g., list-decoding), it is natural to suspect that the sequence $(R_{\mathrm{RLC}}(\mathcal{P}_{n_i}))_{i\in\mathbb{N}}$ will converge to a fixed value. We leave it as an interesting open problem to determine if this is indeed the case.

3.4 Proof of Lemma 3.3.8

In this section we prove Lemma 3.3.8. The first part uses a simple first-moment argument. The real challenge is the second part, where we use the second-moment method.[7]

Proof of Item 1 of Lemma 3.3.8. We begin by proving Item 1 of the lemma. Assume that τ is such that $R^*_\tau = \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau')$ satisfies

$$R \leq R^*_\tau - \varepsilon \, .$$

[7]How appropriate that the first argument uses the first moment and the second argument uses the second moment. You would almost think that was intentional.

Choose $\tau' \in \mathcal{I}_\tau$ achieving $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') = R^*_\tau$ and let $A \in \mathbb{F}_q^{m\times \ell}$ be such that τ' is the distribution of $A\boldsymbol{v}$ where $\boldsymbol{v} \sim \tau$. By Proposition 2.2.2, a matrix $M' \in \mathcal{M}_{n,\tau'}$ is contained in $\boldsymbol{\mathcal{C}}$ with probability $q^{-(1-R)\operatorname{rank}(M')n} = q^{-(1-R)\operatorname{rank}(\tau')n}$, and so

$$\mathbb{P}\left(\exists M \in \mathcal{M}_{\tau'}, M \subset \boldsymbol{\mathcal{C}}\right) \le |\mathcal{M}_{\tau'}| \cdot q^{-(1-R)\operatorname{rank}(\tau')n} \le q^{(H_q(\tau')-(1-R)\operatorname{rank}(\tau'))n} \le q^{-\varepsilon n}\,.$$

The first inequality uses a union bound, the second uses Proposition 3.2.4, and the final uses $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') = 1 - \frac{H_q(\tau')}{\operatorname{rank}(\tau')} \ge R + \varepsilon$.

Finally, note that if $\boldsymbol{\mathcal{C}}$ contains some matrix $M \in \mathcal{M}_\tau$, then by linearity, $M' := MA^T \in \mathcal{M}_{\tau'}$ is also contained in $\boldsymbol{\mathcal{C}}$. So we conclude

$$\mathbb{P}\left(\exists M \in \mathcal{M}_{\tau}, M \subset \boldsymbol{\mathcal{C}}\right) \le q^{-\varepsilon n}\,. \qquad \square$$

We now proceed to the second part of the Lemma, which is more involved.

Proof of Item 2 of Lemma 3.3.8. We wish to show that when the rate R is too large, then a random linear code of rate will contain a matrix of type τ with high probability. Suppose $\tau \in \mathcal{D}_{n,\ell}$ is such that $R^*_\tau = \max_{\tau'\in\mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau')$ satisfies $R \ge R^*_\tau + \varepsilon$.

First, we argue that we may assume $\operatorname{rank}(\tau) = \ell$. By the definition of $\operatorname{rank}(\tau)$, there is some matrix $B \in \mathbb{F}_q^{\operatorname{rank}(\tau)\times\ell}$ of rank $\operatorname{rank}(\tau)$ so that the distribution $\tilde{\tau}$ given by $B\boldsymbol{v}$, $\boldsymbol{v} \sim \tau$ has $\operatorname{rank}(\tilde{\tau}) = \operatorname{rank}(\tau)$. We claim that

$$\max_{\tau'\in\mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \le R - \varepsilon$$

implies that

$$\max_{\tilde{\tau}'\in\mathcal{I}_{\tilde{\tau}}} R^{\mathbb{E}}_{\mathrm{RLC}}(\tilde{\tau}') \le R - \varepsilon.$$

To demonstrate, we prove the contrapositive. Suppose that there is some $\tilde{\tau}' \in \mathcal{I}_{\tilde{\tau}}$ so that $R^{\mathbb{E}}_{\mathrm{RLC}}(\tilde{\tau}') > R - \varepsilon$. Then by the definition of $\mathcal{I}_{\tilde{\tau}}$, there is some matrix $A \in \mathbb{F}_q^{m\times\operatorname{rank}(\tilde{\tau})}$ with $m \le \operatorname{rank}(\tilde{\tau})$ so that $\tilde{\tau}'$ is given by $A\boldsymbol{w}$, $\boldsymbol{w} \sim \tilde{\tau}$. But this is the same as the distribution of $AB\boldsymbol{v}$, $\boldsymbol{v} \sim \tau$, using the definition of $\tilde{\tau}$. Thus, $\tilde{\tau}' \in \mathcal{I}_\tau$, and this implies that $\max_{\tau'\in\mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') > R - \varepsilon$. Finally, we observe that $\binom{n+q^{2\ell}-1}{q^{2\ell}-1}$ is increasing in ℓ, so we conclude that to prove Item 2, we may as well work with the distribution $\tilde{\tau}$ on $\mathbb{F}_q^{\operatorname{rank}(\tilde{\tau})}$. Thus, in the sequel we will assume $\operatorname{rank}(\tau) = \ell$.

For a matrix $M \in \mathbb{F}_q^{n\times\ell}$, let $\boldsymbol{X}_M$ be the indicator variable for the event that $M \subseteq \boldsymbol{\mathcal{C}}$, and let $\boldsymbol{X} = \sum_{M\in\mathcal{M}_\tau} \boldsymbol{X}_M$. Our goal then is to show that $\boldsymbol{X} > 0$ with high probability, and we do so by showing that $\operatorname{Var}(\boldsymbol{X}) = o(\mathbb{E}^2[\boldsymbol{X}])$.

We first show a lower bound on $\mathbb{E}[\boldsymbol{X}]$. Using Propositions 2.2.2 and 3.2.4,

$$\mathbb{E}[\boldsymbol{X}] = |\mathcal{M}_\tau| \cdot q^{-(1-R)\cdot\ell\cdot n} \ge q^{(H_q(\tau)-(1-R)\cdot\ell)\cdot n} \cdot \binom{n+q^\ell-1}{q^\ell-1}^{-1}. \tag{3.12}$$

Next we show an upper bound on $\mathrm{Var}(\boldsymbol{X})$. Given a pair of matrices $M, M' \in \mathcal{M}_\tau$, we let $(M|M')$ denote the $(n \times (2\ell))$-matrix consisting of a left $n \times \ell$ block equal to M, and a right $n \times \ell$ block equal to M'. Then in this notation we have

$$\begin{aligned}\mathrm{Var}(\boldsymbol{X}) &= \sum_{M,M' \in \mathcal{M}_\tau} \mathbb{E}[\boldsymbol{X}_M \cdot \boldsymbol{X}_{M'}] - \mathbb{E}[\boldsymbol{X}_M] \cdot \mathbb{E}[\boldsymbol{X}_{M'}] \\ &= \sum_{M,M' \in \mathcal{M}_\tau} \mathbb{P}((M|M') \subseteq \boldsymbol{\mathcal{C}}) - \mathbb{P}(M \subseteq \boldsymbol{\mathcal{C}}) \cdot \mathbb{P}(M' \subseteq \boldsymbol{\mathcal{C}}) \\ &= \sum_{M,M' \in \mathcal{M}_\tau} q^{-(1-R)\cdot \mathrm{rank}(M|M')\cdot n} - q^{-2\cdot(1-R)\cdot \ell \cdot n}\,.\end{aligned}$$

Notice that in the above sum, terms for which $\mathrm{rank}(M|M') = 2\ell$ vanish. Let

$$\mathcal{M} := \{(M|M') : M, M' \in \mathcal{M}_\tau \text{ and } \mathrm{rank}(M|M') < 2\ell\}$$

and

$$\mathcal{D} := \{\tau_M : M \in \mathcal{M}\}\,.$$

Then we have

$$\begin{aligned}\mathrm{Var}(\boldsymbol{X}) &\le \sum_{M \in \mathcal{M}} q^{-(1-R)\,\mathrm{rank}(M)n} \\ &= \sum_{\tau' \in \mathcal{D}} \sum_{M \in \mathcal{M}_{\tau'}} q^{-(1-R)\,\mathrm{rank}(M)n} \\ &= \sum_{\tau' \in \mathcal{D}} |\mathcal{M}_{\tau'}| \cdot q^{-(1-R)\,\mathrm{rank}(\tau')n} \\ &\le \sum_{\tau' \in \mathcal{D}} q^{(H_q(\tau') - (1-R)\,\mathrm{rank}(\tau'))n}\,, \end{aligned} \tag{3.13}$$

where the inequality used Proposition 3.2.4. We seek a bound on $H_q(\tau') - (1-R)\,\mathrm{rank}(\tau')n$, which is provided by the following claim. Recall that $\varepsilon > 0$ is such that $\varepsilon \le R - R^*_\tau$, i.e., it lower bounds the amount by which the rate of the random linear code exceeds R^*_τ.

Claim 3.4.1. *For any $\tau' \in \mathcal{D}$,*

$$H_q(\tau') - (1-R)\cdot \mathrm{rank}(\tau') \le 2(H_q(\tau) - (1-R)\cdot \ell) - \varepsilon\,.$$

We show how to complete the proof assuming the claim. Continuing from (3.13),

$$\mathrm{Var}(\boldsymbol{X}) \le |\mathcal{D}| q^{(2(H_q(\tau) - (1-R)\cdot \ell) - \varepsilon)n} \le \binom{n + q^{2\ell} + 1}{q^{2\ell} - 1} \cdot q^{(2(H_q(\tau) - (1-R)\cdot \ell)n} \cdot q^{-\varepsilon n}\,. \tag{3.14}$$

Above, we used the fact that $\mathcal{D} \subseteq \mathcal{D}_{n,\ell}$ and applied (3.3). Combining (3.12) and (3.14), by Chebyshev's inequality we conclude that

$$\mathbb{P}(\boldsymbol{X} = 0) \le \frac{\mathrm{Var}(\boldsymbol{X})}{\mathbb{E}^2[\boldsymbol{X}]} \le \binom{n + q^{2\ell} - 1}{q^{2\ell} - 1}^3 q^{-\varepsilon n}\,.$$

To complete the proof, we prove Claim 3.4.1 which we used above.

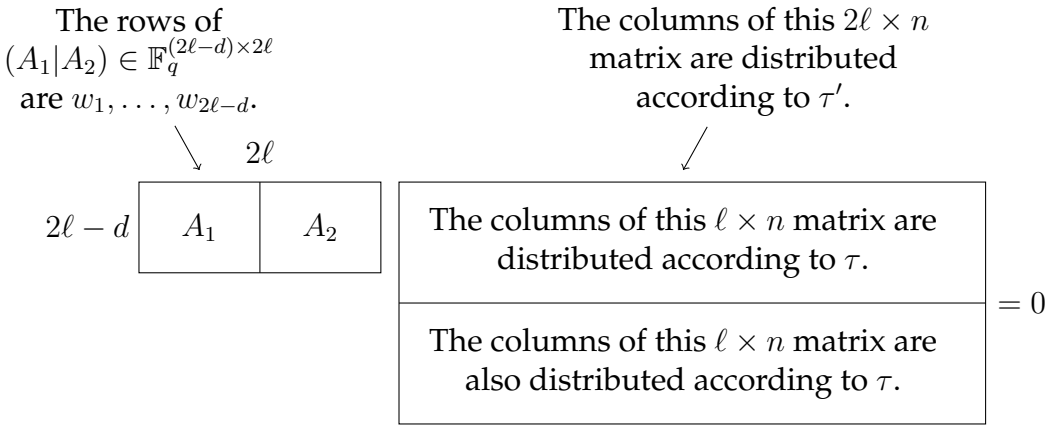

Figure 3.1: Notation in the proof of Claim 3.4.1.

Proof of Claim 3.4.1. In what follows, let $d := \mathrm{rank}(\tau')$, and $V := \mathrm{span}(\mathrm{supp}(\tau')) \leq \mathbb{F}_q^{2\ell}$. Let $w_1, \ldots, w_{2\ell-d} \in \mathbb{F}_q^{2\ell}$ be a basis for $V^\perp$. Let $\pi_1 : \mathbb{F}_q^{2\ell} \to \mathbb{F}_q^{\ell}$ (respectively, π_2) denote the projection of a vector $w \in \mathbb{F}_q^{2\ell}$ to the first (respectively, last) ℓ coordinates. Finally, let A be the matrix whose rows are $w_1, \ldots, w_{2\ell-d}$, and let $A_1 \in \mathbb{F}_q^{(2\ell-d)\times \ell}$ (A_2, respectively) denote the matrix whose rows are $\pi_1(w_1), \ldots, \pi_1(w_{2\ell-d})$ ($\pi_2(w_1), \ldots, \pi_2(w_{2\ell-d})$, respectively). See Fig. 3.1 for a diagram of this notation.

We claim that all rows of A_1 are linearly independent, and so $\mathrm{rank}(A_1) = 2\ell - d$. To see this, suppose for a contradiction that $\pi_1(w_1), \ldots, \pi_1(w_{2\ell-d})$ are linearly dependent. Then there exists a non-trivial linear combination of $w_1, \ldots, w_{2\ell-d}$ that sums to a non-zero vector of the form $(0, w)$. But this means that $\pi_2(\mathrm{supp}(\tau')) = \mathrm{supp}(\tau)$ is orthogonal to w, in contradiction to our assumption that $\mathrm{span}(\mathrm{supp}(\tau)) = \mathbb{F}_q^{\ell}$. Consequently, recalling that $\mathrm{rank}(\tau) = \ell$, the distribution τ'' given by $A_1\boldsymbol{w}$ for $\boldsymbol{w} \sim \tau$ has $\mathrm{rank}(\tau'') = 2\ell - d$. As $\tau'' \in \mathcal{I}_\tau$, $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau'') \leq R - \varepsilon$.

Recall that $I_q(\boldsymbol{x}; \boldsymbol{y})$ denotes the base-q mutual information of the random variables $\boldsymbol{x}$ and $\boldsymbol{y}$. For $\boldsymbol{v} \sim \tau'$ we have

$$\begin{aligned}
H_q(\tau') &= H_q(\boldsymbol{v}) \\
&= H_q(\pi_1(\boldsymbol{v})) + H_q(\pi_2(\boldsymbol{v})) - I_q(\pi_1(\boldsymbol{v}); \pi_2(\boldsymbol{v})) && (3.15)\\
&= 2H_q(\tau) - I_q(\pi_1(\boldsymbol{v}); \pi_2(\boldsymbol{v})) && (3.16)\\
&\leq 2H_q(\tau) - I_q(A_1\pi_1(\boldsymbol{v}); -A_2\pi_2(\boldsymbol{v})) && (3.17)\\
&= 2H_q(\tau) - H_q(A_1\pi_1(\boldsymbol{v})) && (3.18)\\
&\leq 2H_q(\tau) - (1 - R + \varepsilon) \cdot \mathrm{rank}(\tau'') && (3.19)\\
&= 2H_q(\tau) - (1 - R + \varepsilon) \cdot (2\ell - d)\,.
\end{aligned}$$

The equality (3.15) follows from the definition of mutual information, using $\boldsymbol{v} = (\pi_1(\boldsymbol{v}), \pi_2(\boldsymbol{v}))$. The equality (3.16) follows from the fact that π_1 and π_2 are injective on $\text{row-span}(A)$. The inequality (3.17) follows from the data-processing inequality. The equality (3.18) follows since $A_1\pi_1(\boldsymbol{v}) + A_2\pi_2(\boldsymbol{v}) = A\boldsymbol{v} = 0$. Finally, inequality (3.19) follows because

$1 - \frac{H_q(\tau'')}{\mathrm{rank}(\tau'')} = R^{\mathbb{E}}_{\mathrm{RLC}}(\tau'') \leq R - \varepsilon$. Rearranging, and recalling the assumption that $2\ell > d$, gives the desired conclusion. □

The proof of Item 2 of Lemma 3.3.8 is thereby completed. □

3.5 New Derivations of Known Results

In order to demonstrate the power of the types framework, we show how to use it to rederive some known results concerning random linear codes.

3.5.1 Showing Random Linear Codes Achieve the GV Bound

Recall that it is known that random linear codes achieve the GV bound (Theorem 2.4.4) with high probability. That is, for any $\delta \in (0, 1 - 1/q)$, a random linear code of rate roughly $1 - h_q(\delta)$ has distance δ with high probability. We show how to prove this fact using our types framework. Recalling Example 3.2.7, if $\mathcal{T} \subseteq \mathcal{D}_{n,1}$ denotes the set of all types τ for which $1 > \tau(0) \geq 1 - \delta$, showing that a code $\mathcal{C} \leq \mathbb{F}_q^n$ does not contain a vector x with $0 < \mathrm{wt}(x) \leq \delta$ is the same as showing that it doesn't contain a vector[8] of type $\tau \in \mathcal{T}$. We wish to compute the threshold for $\mathcal{T}$-freeness, $R_{\mathrm{RLC}}(\mathcal{P}^{\mathcal{T}})$. By Theorem 3.3.9,

$$R_{\mathrm{RLC}}(\mathcal{P}^{\mathcal{T}}) = \min_{\tau \in \mathcal{T}} \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \pm o(1) \ .$$

Thus, given a $\tau \in \mathcal{T}$, we must show that there exists an implied type $\tau' \in \mathcal{I}_\tau$ for which $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') = 1 - \frac{H_q(\tau')}{\mathrm{rank}(\tau')} \leq 1 - h_q(\delta)$. To establish this, we just compute $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau)$, the expectation threshold of τ. Note first that $\tau(0) < 1$ guarantees $\mathrm{supp}(\tau) \subseteq \mathbb{F}_q$ contains a nonzero element, and therefore $\mathrm{rank}(\tau) = 1$. To upper bound the q-ary entropy, we use the following proposition.

Proposition 3.5.1. *Let $\delta \in (0, 1 - 1/q)$ and let $\tau \sim \mathbb{F}_q$ be a distribution which, for some $x_0 \in \mathbb{F}_q$, satisfies $\tau(x_0) \geq 1 - \delta$. Then $H_q(\tau) \leq h_q(\delta)$.*

Proof. We compute

$$\begin{aligned} H_q(\tau) &= \sum_{x \in \mathbb{F}_q} \tau(x) \log_q \left(\tfrac{1}{\tau(x)} \right) = \tau(x_0) \log_q \left(\tfrac{1}{\tau(x_0)} \right) + \sum_{x \neq x_0} \tau(x) \log_q \left(\tfrac{1}{\tau(x)} \right) \\ &\leq \tau(x_0) \log_q \left(\tfrac{1}{\tau(x_0)} \right) + \left(\sum_{x \neq x_0} \tau(x) \right) \cdot \log_q \left(\frac{q-1}{\sum_{x \neq x_0} \tau(x)} \right) \\ &= h_q \left(\sum_{x \neq x_0} \tau(x) \right) \leq h_q(\delta) \ . \end{aligned}$$

[8]Typically we speak of matrices of a given type, but as a matrix of type $\tau \in \mathcal{D}_{n,1}$ is $n \times 1$ such a matrix is more naturally thought of as a vector.

In the above computations, the first inequality follows from the concavity of the function $y \mapsto y \log \frac{1}{y}$, and the second uses the assumption $\tau(x_0) \geq 1 - \delta$ and the fact that $h_q(\delta)$ increases with δ for $\delta \in (0, 1 - 1/q)$. □

Thus,

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) = 1 - \frac{H_q(\tau)}{\mathrm{rank}(\tau)} = 1 - h_q(\delta) \, .$$

Since the previous argument was valid for any $\tau \in \mathcal{D}_{n,1}$ with $1 > \tau(0) > 1 - \delta$, we conclude

$$R_{\mathrm{RLC}}(\mathcal{P}^{\mathcal{T}}) \pm o(1) = \min_{\tau \in \mathcal{T}} \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \geq 1 - h_q(\delta) \, .$$

If desired, one can also prove the corresponding upper bound on $R_{\mathrm{RLC}}(\mathcal{P}^{\mathcal{T}})$ by considering the specific type $\tau^* \in \mathcal{D}_{n,1}$ which assigns probability mass $1-\delta$ to 0 and probability mass $\frac{\delta}{q-1}$ to each $x \in \mathbb{F}_q \setminus \{0\}$; such a type has entropy $h_q(\delta)$ and rank 1, so its expectation threshold is $1 - h_q(\delta)$. As τ^* does not contain any nontrivial implied types, we conclude that $R_{\mathrm{RLC}}(\tau^*) = R^{\mathbb{E}}_{\mathrm{RLC}}(\tau^*) = 1 - h_q(\delta)$, so

$$R_{\mathrm{RLC}}(\mathcal{P}^{\mathcal{T}}) \pm o(1) = \min_{\tau \in \mathcal{T}} \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \leq \max_{\tau' \in \mathcal{I}_{\tau^*}} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') = 1 - h_q(\delta) \, .$$

Remark 3.5.2. Of course, the distribution τ^* defined as above need not be a type with denominator n, but one can adjust the probability masses slightly so that it does have denominator n. By continuity, this only affects $H_q(\tau^*)$ by $o(1)$ terms, which we may safely ignore. In the sequel we will ignore this technicality.

Now, suppose we had been a bit less clever in defining the types we must forbid in order to prove that a random linear code has distance δ. Specifically, suppose that we had chosen to forbid all types $\tau \in \mathcal{D}_{n,2}$ for which

$$0 < \mathop{\mathbb{P}}_{(\boldsymbol{x},\boldsymbol{y}) \sim \tau} (\boldsymbol{x} = \boldsymbol{y}) \leq \delta \, . \tag{3.20}$$

Note that this amounts to saying that the code must not contain an $n \times 2$ matrix whose columns are at Hamming distance δ from one another. If one were to compute the expectation threshold of such a τ, one would not in general obtain a better lower bound than $1 - \frac{1+h_q(\delta)}{2} < 1 - h_q(\delta)$. Indeed, consider the following τ: it samples $(\boldsymbol{x}, \boldsymbol{y}) \sim \mathbb{F}_q^2$ such that $\boldsymbol{x}$ and $\boldsymbol{y}$ are marginally uniform over $\mathbb{F}_q$, and that $\mathbb{P}_{(\boldsymbol{x},\boldsymbol{y}) \sim \tau} (\boldsymbol{x} = \boldsymbol{y}) = \delta$.[9] Now, observe that

$$H_q(\tau) = H_q(\boldsymbol{x}, \boldsymbol{y}) = H_q(\boldsymbol{x}) + H_q(\boldsymbol{y} | \boldsymbol{x}) = 1 + h_q(\delta) \, ,$$

and so

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) = 1 - \frac{H_q(\tau)}{2} = 1 - \frac{1 + h_q(\delta)}{2} \, ,$$

as claimed.

[9]This is slightly at odds with the previous assumption that neither of the columns of a matrix of type τ may be 0; however, this discrepancy would be nullified when we enforce the assumption that the type have denominator n.

However, we claim that any such τ has an implied type $\tau' \in \mathcal{I}_\tau$ with $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \geq 1 - h_q(\delta)$. Specifically, take the type implied by the linear map which maps $(x, y) \mapsto x - y$. Note that this $\tau' \in \mathcal{D}_{n,1}$ satisfies

$$1 > \mathop{\mathbb{P}}_{\boldsymbol{x} \sim \tau'} (\boldsymbol{x} = 0) \geq 1 - \delta \, ,$$

and Proposition 3.5.1 demonstrates that such a τ' has entropy at least $h_q(\delta)$. Since $\mathrm{rank}(\tau') = 1$ (as $\mathbb{P}_{(\boldsymbol{x},\boldsymbol{y}) \sim \tau} (\boldsymbol{x} = \boldsymbol{y}) > 0$, it follows that the image of $\mathrm{supp}(\tau)$ under the map $(x, y) \mapsto x - y$ contains a nonzero point), the expectation threshold $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \geq 1 - h_q(\delta)$. Thus, if $\mathcal{T} \subseteq \mathcal{D}_{n,2}$ denotes the set of all types satisfying Condition (3.20), we still find

$$R_{\mathrm{RLC}}(\mathcal{P}^{\mathcal{T}}) \pm o(1) = \min_{\tau \in \mathcal{T}} \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \geq 1 - h_q(\delta) \, ,$$

thereby again establishing that random linear codes achieve the GV bound with high probability.

3.5.2 Recovering Known Results on the List-Decodability of Random Linear Codes

In this section, we show how two known results on the list-decodability of random linear codes can be obtained in our types framework. First, we study the Zyablov-Pinsker ([ZP81]) argument; later, we show how to adapt the more sophisticated Guruswami-Håstad-Kopparty ([GHK11]) argument. Throughout this section, we let $\mathcal{T} \subseteq \mathcal{D}_{n,L+1}$ denote the set of types defined in Example 3.2.8: that is, $\mathcal{T}$ consists of all $\tau \in \mathcal{D}_{n,L+1}$ such that for some type $\tau' \in \mathcal{D}_{n,1}$,

$$\forall i \in [L+1], \quad \mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{x}) \sim (\tau,\tau')} (\boldsymbol{u}_i \neq \boldsymbol{x}) \leq \rho \tag{3.21}$$

and

$$\forall i \neq j \in [L+1], \quad \mathop{\mathbb{P}}_{\boldsymbol{u} \sim \tau} (\boldsymbol{u}_i \neq \boldsymbol{u}_j) > 0 \, . \tag{3.22}$$

The Zyablov-Pinsker Argument

We now provide a new proof of the following result of Zyablov and Pinsker.

Theorem 3.5.3 ([ZP81])**.** *A random linear code of rate $1 - h_q(\rho) - \frac{1}{\lceil \log_q(L+1) \rceil}$ is with high probability (ρ, L)-list-decodable.*

Proof. Let $\tau \in \mathcal{T}$ be a type satisfying Conditions 3.21 and 3.22. We claim that $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) \leq 1 - \frac{h_q(\rho)}{\lceil \log_q(L+1) \rceil}$. On the one hand, Condition 3.22 guarantees that any matrix sampled according to τ has rank at least $\lceil \log_q(L+1) \rceil$, as any such matrix must have distinct

columns. That is to say, $\mathrm{rank}(\tau) \geq \lceil \log_q(L+1) \rceil$. On the other hand, we may upper bound its q-ary entropy as

$$H_q(\tau) = H_q(\tau, \tau') - H_q(\tau|\tau') = H_q(\tau|\tau') + H_q(\tau') - H_q(\tau'|\tau) \leq H_q(\tau|\tau') + 1 \,. \tag{3.23}$$

Hence, we seek an effective upper bound on $H_q(\tau|\tau')$. Let $(\boldsymbol{u}, \boldsymbol{x}) \sim (\tau, \tau')$, and choose a subset $I \subseteq [L+1]$ of size $\mathrm{rank}(\tau)$ such that $(\boldsymbol{u}_i)_{i \in I}$ determines the entire vector $\boldsymbol{u}$. That is, I is an information set for the subspace $\mathrm{span}(\mathrm{supp}(\tau)) \leq \mathbb{F}_q^{L+1}$. We compute

$$H_q(\tau|\tau') = H_q(\boldsymbol{u}|\boldsymbol{x}) = H_q((\boldsymbol{u}_i)_{i \in I}|\boldsymbol{x}) \leq \sum_{i \in I} H_q(\boldsymbol{u}_i|\boldsymbol{x}) = \sum_{i \in I} \sum_{x \in \mathbb{F}_q} \mathbb{P}\left(\boldsymbol{x} = x\right) \cdot H_q(\boldsymbol{u}_i|\boldsymbol{x} = x) \,.$$

Now, note that Condition 3.21 guarantees that the random variable $\boldsymbol{u}_i \sim \mathbb{F}_q$ conditioned on $\boldsymbol{x} = x$ takes on the value x with probability at least $1-\rho$, and takes on a different value with probability at most ρ. Appealing to Proposition 3.5.1, we find $H_q(\boldsymbol{u}_i|\boldsymbol{x}) \leq h_q(\rho)$. Putting everything together, we conclude

$$H_q(\tau) \leq |I| \cdot h_q(\rho) + 1 = \mathrm{rank}(\tau) \cdot h_q(\rho) + 1 \,.$$

Hence,

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) = 1 - \frac{H_q(\tau)}{\mathrm{rank}(\tau)} \geq 1 - \frac{\mathrm{rank}(\tau) \cdot h_q(\rho) + 1}{\mathrm{rank}(\tau)} \geq 1 - h_q(\rho) - \frac{1}{\lceil \log_q(L+1) \rceil} \,.$$

Since the previous argument applies equally well to any $\tau \in \mathcal{T}$, Theorem 3.3.9 now yields

$$R_{\mathrm{RLC}}(\mathcal{P}^\tau) \pm o(1) = \min_{\tau \in \mathcal{T}} \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \geq 1 - h_q(\rho) - \frac{1}{\lceil \log_q(L+1) \rceil} \,.$$

This demonstrates that a random linear code of rate at most $1 - h_q(\rho) - \frac{1}{\lceil \log_q(L+1) \rceil}$ is, with high probability, (ρ, L)-list-decodable. □

The Guruswami-Håstad-Kopparty Argument

Next, we show how to recover the Guruswami-Håstad-Kopparty ([GHK11]) result, assuming a technical combinatorial result of theirs. Specifically, we use:

Lemma 3.5.4 ([GHK11, Theorem 10])**.** *For every prime power q and $\rho \in (0, 1 - 1/q)$, there is a constant $C = C_{\rho,q} > 1$ such that for all n and all $\ell = o(\sqrt{n})$, the following holds. Suppose $L \geq C \cdot \ell$ and, for $j \in [L]$, let $b_j = (b_{j1}, \ldots, b_{j\ell}) \in \mathbb{F}_q^\ell$ be distinct vectors. If $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell$ are sampled independently and uniformly from $B(0, \rho)$, then*

$$\mathbb{P}\left(\sum_{i=1}^{\ell} b_{ji} \boldsymbol{x}_i \in B(0, \rho) \;\; \forall j \in [L] \right) \leq q^{-(6 - o(1))n} \,.$$

Remark 3.5.5. This is actually a slight weakening of Theorem 10 from [GHK11]; there, they conclude that $\mathbb{P}\left(|\operatorname{span}\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell\} \cap B(0,\rho)| \geq L\right) \leq q^{-(6-o(1))n}$. However, in fact their argument (implicitly) proceeds by proving the above result and then taking a union bound over the choice of vectors b_j (of which there are at most $q^{\ell L}$). So this is not really a weaker result.

We are then able to rederive:

Theorem 3.5.6 ([GHK11, Theorem 6])**.** *Let q be a prime power and $\rho \in (0, 1-1/q)$. Then there exists a constant $C = C_{\rho,q} > 0$ such that for all $\varepsilon > 0$, if $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ is a random linear code of rate $1 - h_q(\rho) - \varepsilon$, then $\boldsymbol{\mathcal{C}}$ is $(\rho, C/\varepsilon)$-list-decodable with high probability.*

Proof. Let $\tau \in \mathcal{T}$, i.e., $\tau \in \mathcal{D}_{n,L+1}$ satisfies Conditions 3.21 and 3.22. Appealing to Theorem 3.3.9, it will suffice to show that

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) \geq 1 - h_q(\rho) - \varepsilon \, .$$

We assume n is large enough so that the $o(1)$ term in the exponent of Lemma 3.5.4 is at most 1. We take $C = C_{\rho,q}$ to be the constant from Lemma 3.5.4 and assume $L \geq 2C/\varepsilon$. Abbreviate $r := \operatorname{rank}(\tau)$. We now consider two cases, depending on r.

Case 1: $r \geq 1/\varepsilon$**.** We assume first that the rank of τ is large. As in the proof of Theorem 3.5.3, we may show that

$$H_q(\tau) \leq r \cdot h_q(\rho) + 1 \, .$$

Hence,

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) = 1 - \frac{H_q(\tau)}{r} \geq 1 - h_q(\rho) - \frac{1}{r} \geq 1 - h_q(\rho) - \varepsilon \, .$$

Case 2: $r < 1/\varepsilon$**.** Suppose now that the rank of τ is small. Note that in this case, $L+1-r \geq Cr$, where we recall $C = C_{\rho,q}$ is the constant from Lemma 3.5.4. We endeavor to prove a better upper bound on $H_q(\tau)$. As in (3.23),

$$H_q(\tau) \leq H_q(\tau|\tau') + 1 \, .$$

If $(\boldsymbol{u}, \boldsymbol{x}) \sim (\tau, \tau')$,

$$H_q(\tau|\tau') = H_q(\boldsymbol{u}|\boldsymbol{x}) \leq H_q(\boldsymbol{u} - \boldsymbol{x} \cdot \mathbb{1}) \, ,$$

where $\mathbb{1}$ denotes the all-ones vector. Let $\sigma \in \mathcal{D}_{n,L+1}$ denote the type corresponding to the random variable $\boldsymbol{u} - \boldsymbol{x} \cdot \mathbb{1}$, i.e., for each $v \in \mathbb{F}_q^{L+1}$,

$$\sigma(v) = \mathbb{P}\left(\boldsymbol{u} - \boldsymbol{x} \cdot \mathbb{1} = v\right) \, .$$

Note that Condition 3.21 implies

$$\forall i \in [L+1], \quad \mathop{\mathbb{P}}_{\boldsymbol{v} \sim \sigma}\left(\boldsymbol{v}_i \neq 0\right) \leq \rho \, . \tag{3.24}$$

By Proposition 3.2.4, we know that

$$H_q(\sigma) \le \frac{1+o(1)}{n} \log_q |\mathcal{M}_\sigma| \, .$$

Finally, observe that Lemma 3.5.4 guarantees that $|\mathcal{M}_\sigma| \le q^{nrh_q(\rho)} \cdot q^{-5n}$. In justification of this, consider the columns $x_1, \ldots, x_{L+1}$ of any $M \in \mathcal{M}_\sigma$, and asumme without loss of generality that $x_1, \ldots, x_r$ are linearly independent and $x_j \in \operatorname{span}\{x_1, \ldots, x_r\}$ for all $j \ge r+1$. Fix scalars $b_{ji} \in \mathbb{F}_q$ such that $x_j = \sum_{i=1}^r b_{ji} x_i$ for each $j \ge r+1$, and observe that as the vectors $x_{r+1}, \ldots, x_{L+1}$ are distinct and the vectors $x_1, \ldots, x_r$ are linearly independent, the vectors $b_j = (b_{j1}, \ldots, b_{j\ell})$ are distinct. Also, by Condition 3.24, $M \in \mathcal{M}_\sigma$ implies that every column $x_1, \ldots, x_{L+1} \in B(0,\rho)$. Hence, we may upper bound $|\mathcal{M}_\sigma|$ by

$$q^{rh_q(\rho)n} \cdot \mathop{\mathbb{P}}_{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_r \sim B(0,\rho)} \left(\sum_{i=1}^r b_{ji} \boldsymbol{x}_i \in B(0,\rho) \ \ \forall r+1 \le j \le L+1 \right) \, .$$

By Lemma 3.5.4, as $L+1-r \ge Cr$, we obtain the bound $q^{nrh_q(\rho)} \cdot q^{-5n}$, as desired. Hence,

$$H_q(\tau|\tau') \le H_q(\sigma) \le \frac{1+o(1)}{n} \log_q |\mathcal{M}_\sigma| \le (1+o(1)) \cdot (rh_q(\rho) - 5) \le rh_q(\rho) - 4 \, ,$$

where the last inequality holds for large enough n. Therefore

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau) = 1 - \frac{H_q(\tau)}{r} \ge 1 - h_q(\rho) + \frac{4}{r} \ge 1 - h_q(\rho) - \varepsilon \, . \qquad \square$$

3.6 An Application to List-of-2 Decoding

Finally, in this section, we study the list-of-2-decoding radius of random linear codes in $\mathbb{F}_2^n$. It is known (see [Bli86], also [ABP18]) that whenever $\rho < 1/4$ there exist $(\rho, 2)$-list-decodable codes with positive rate, but whenever $\rho > 1/4$ the only $(\rho, 2)$-list-decodable codes are of bounded size, independent of n. See also [ABL00], where an improved upper bound on the capacity for list-of-2 decoding is established.

For mathematical simplicity, we focus upon $(\rho, 2)$-*average-radius* list-decoding. Also, recall that if subscripts are omitted, $h(x)$ and $\log x$ are computed to the base 2.

Theorem 3.6.1. *Let $\rho \in (0, 1/4)$ and let $\boldsymbol{\mathcal{C}} \le \mathbb{F}_2^n$ be a random linear code of rate R.*

1. *If*

$$R < 1 - \frac{h(3\rho) + 3\rho \log_2 3}{2} \, ,$$

then a random linear code of rate R is $(\rho, 2)$-average-radius list-decodable with high probability.

2. *If*

$$R > 1 - \frac{h(3\rho) + 3\rho \log_2 3}{2} \, ,$$

then a random linear code of rate R fails to be $(\rho, 2)$-average-radius list-decodable with high probability.

Proof. For $\mathcal{C}$ to be $(\rho, 2)$-average-radius list-decodable, it must be the case that it does not contain any type $\tau \in \mathcal{D}_{n,3}$ for which there is some $\tilde{\tau} \in \mathcal{D}_{n,1}$ such that

$$\frac{1}{3}\sum_{i=1}^{3} \mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{x})\sim(\tau,\tilde{\tau})} (\boldsymbol{u}_i \neq \boldsymbol{x}) \leq \rho \tag{3.25}$$

and

$$\forall 1 \leq i < j \leq 3, \quad \mathop{\mathbb{P}}_{\boldsymbol{u}\sim\tau} (\boldsymbol{u}_i \neq \boldsymbol{u}_j) > 0 \,. \tag{3.26}$$

Denote the set of all such τ by $\mathcal{T}$. Note that if $\tilde{\tau}$ is defined to always sample $\mathrm{MAJ}(\boldsymbol{u})$, then the left-hand side of (3.25) can only decrease. Hence, we may assume $\boldsymbol{x} = \mathrm{MAJ}(\boldsymbol{u})$, in which case the condition (3.25) may be replaced by

$$\frac{1}{3}\sum_{i=1}^{3} \mathop{\mathbb{P}}_{\boldsymbol{u}\sim\tau} (\boldsymbol{u}_i \neq \mathrm{MAJ}(\boldsymbol{u})) \leq \rho \,. \tag{3.27}$$

Now, let[10]

$$A = \{000, 111\} \quad \text{and} \quad B = \mathbb{F}_2^3 \setminus A \,.$$

Then, defining $x = \tau(B)$, Condition 3.27 becomes

$$x \leq 3\rho \,. \tag{3.28}$$

Now, we establish Statement 1 of the theorem. Consider the implied type $\tau^* \in \mathcal{I}_\tau$ defined by the linear map $(a, b, c) \mapsto (a + b, a + c)$. Note that the kernel of this map is $\{000, 111\}$, and so $\tau^*(00) = 1 - x$. Hence, $\tau^*(10) + \tau^*(01) + \tau^*(11) = x$. We may therefore upper bound the entropy as

$$\begin{aligned} H_q(\tau^*) &= \tau^*(00) \cdot \log \tfrac{1}{\tau^*(00)} + \tau^*(01) \cdot \log \tfrac{1}{\tau^*(01)} + \tau^*(10) \cdot \log \tfrac{1}{\tau^*(10)} + \tau^*(11) \cdot \log \tfrac{1}{\tau^*(11)} \\ &\leq (1 - x) \log \tfrac{1}{1-x} + x \log \tfrac{3}{x} = h(x) + x \log 3 \,. \end{aligned}$$

The above inequality uses the concavity of the function $y \mapsto y \log \frac{1}{y}$. Note that in the range $[0, 3/4)$, the function $x \mapsto h(x) + x \log 3$ is increasing: clearly $h(0) + 0 \cdot \log 3 = 0$, and moreover the derivative of $h(x) + x \log 3$ with respect to x is $\log\left(\frac{3(1-x)}{x}\right)$ which is positive assuming $\frac{3(1-x)}{x} > 1$, which rearranges to $x < 3/4$. Hence, as $x \leq 3\rho$ and $\rho < 1/4$, we conclude

$$H_q(\tau^*) \leq h(3\rho) + 3\rho \log 3 \,.$$

Now, we note that $\mathrm{rank}(\tau^*) = 2$. Let $U = \mathrm{span}(\mathrm{supp}(\tau))$. If $\mathrm{rank}(\tau^*) \leq 1$, then we would require $\dim(U) \leq 2$ and $111 \in U$ (recall that the kernel of $(a, b, c) \mapsto (a + b, b + c)$ is $\{000, 111\}$). This implies

$$U \in \{\{000\}, \{000, 111\}, \{000, 111, 001, 110\}, \{000, 111, 010, 101\}, \{000, 111, 100, 011\}\} \,.$$

[10] In this proof, we denote vectors by the corresponding string for readability.

In any case, we find that τ contradicts (3.26). For example, if $U = \{000, 111, 001, 110\}$, then $\mathbb{P}_{\boldsymbol{u}\sim\tau}(\boldsymbol{u}_1 \neq \boldsymbol{u}_2) = 0$.

Putting everything together, we conclude

$$R_{\mathrm{RLC}}(\mathcal{P}^\tau) \pm o(1) = \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \geq R^{\mathbb{E}}_{\mathrm{RLC}}(\tau^*) = 1 - \frac{H(\tau^*)}{\mathrm{rank}(\tau^*)} \geq 1 - \frac{h(3\rho) + 3\rho \log 3}{2} \,.$$

As the previous argument applies equally well to any type $\tau \in \mathcal{T}$, applying Theorem 3.3.9 we conclude that a random linear code of rate less than $1 - \frac{h(3\rho)+3\rho\log 3}{2}$ is $(\rho, 2)$-average-radius list-decodable with high probability.

We now turn to establishing the second statement of the theorem. To do this, we provide a specific type $\tau_0 \in \mathcal{T}$ that is bad for $(\rho, 2)$-average-radius list-decoding and show that all of its implied types have expectation threshold at most $1 - \frac{h(3\rho)+3\rho\log 3}{2}$.

Specifically, take the type $\tau_0 \sim \mathbb{F}_2^3$ with

$$\begin{aligned}
&\tau_0(000) = \tau_0(111) = \frac{1-3\rho}{2} \text{ and} \\
&\tau_0(001) = \tau_0(010) = \tau_0(100) = \tau_0(011) = \tau_0(101) = \tau_0(110) = \frac{\rho}{2} \,.
\end{aligned}$$

First, one can compute that

$$H(\tau_0) = h(3\rho) + 1 + 3\rho \log 3$$

and so

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_0) = 1 - \frac{H(\tau_0)}{\mathrm{rank}(\tau_0)} = 1 - \frac{h(3\rho) + 1 + 3\rho \log 3}{3} \,.$$

Next, consider the type τ_1 implied by the map $(a, b, c) \mapsto (a + b, a + c)$; it has entropy

$$H(\tau_1) = h(3\rho) + 3\rho \log 3$$

and so

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_1) = 1 - \frac{H(\tau_1)}{\mathrm{rank}(\tau_1)} = 1 - \frac{h(3\rho) + 3\rho \log 3}{2} \,.$$

Next, consider any type τ_2 implied by a map $\mathbb{F}_2^3 \to \mathbb{F}_2^2$ with a vector from B in the kernel. In this case, the entropy $H(\tau_2) = h(2\rho) + 1$, and so

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_2) = 1 - \frac{H(\tau_2)}{\mathrm{rank}(\tau_2)} = 1 - \frac{h(2\rho) + 1}{2} \,.$$

Next, consider any type τ_3 implied by a map $\mathbb{F}_2^3 \to \mathbb{F}_2$ with 111 in the kernel. In this case, the entropy $H(\tau_3) = h(2\rho)$, and so

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_3) = 1 - \frac{H(\tau_3)}{\mathrm{rank}(\tau_3)} = 1 - h(2\rho) \,.$$

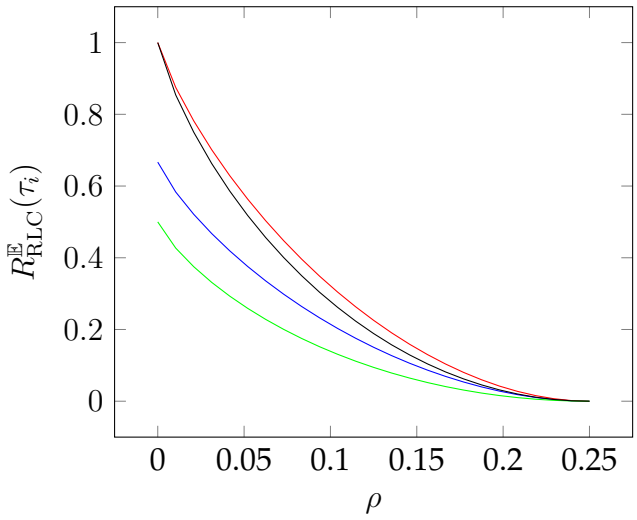

Figure 3.2: Plots of $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_i)$ for each $i \in \{0,1,2,3\}$. $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_0)$ is in blue; $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_1)$ is in red; $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_2)$ is in green; and $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_3)$ is in black. One can see that, uniformly over $\rho \in [0, 0.25]$, the maximum is obtained by $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_1)$.

Finally, consider any type τ_4 implied by a map $\mathbb{F}_2^3 \to \mathbb{F}_2$ without 111 in the kernel. In this case, the entropy $H(\tau_4) = 1$, and so

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_4) = 1 - \frac{H(\tau_4)}{\mathrm{rank}(\tau_4)} = 1 - \frac{1}{1} = 0\,.$$

This completes the computation of the expectation threshold of all of τ_0's implied types. It is now just a finite check to verify that all of the expectation thresholds are at most $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_1) = 1 - \frac{h(3\rho)+\log 3}{2}$. For example, to show that

$$R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_0) = 1 - \frac{h(3\rho) + 1 + 3\rho\log 3}{3} \leq 1 - \frac{h(3\rho)+\log 3}{2}\,,$$

one can note that both the left-hand side and the right-hand side are 0 at $\rho = 1/4$, and moreover the derivative of the right-hand side minus the left-hand side is $\frac{1}{6}\left(\log_2\left(\frac{3\rho}{1-3\rho}\right) - \log_2 3\right)$, which is negative if $\rho < 1/4$. We omit the remaining computations, which are completely routine; for a pictorial proof, see Figure 3.2. Thus, we conclude that

$$R_{\mathrm{RLC}}(\mathcal{P}^{\tau_0}) \leq \max_{i\in\{0,\ldots,4\}} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau_i) + o(1) = 1 - \frac{h(3\rho)+\log 3}{2} + o(1)\,,$$

and so, recalling $\mathcal{T} \subseteq \mathcal{D}_{n,3}$ is the set of forbidden types,

$$R_{\mathrm{RLC}}(\mathcal{P}^{\mathcal{T}}) = \min_{\tau\in\mathcal{T}} R_{\mathrm{RLC}}(\tau) \leq R_{\mathrm{RLC}}(\tau_0) = 1 - \frac{h(3\rho)+\log 3}{2} + o(1)\,. \qquad \square$$

Chapter 4

LDPC Codes Achieve List-Decoding Capacity

In this section, we study Gallager's ensemble of Low-Density Parity-Check (LDPC) codes. Our main contribution is to show that a random code drawn from this ensemble achieves list decoding capacity with high probability. These are the first graph-based codes shown to have this property: prior codes known to achieve list decoding capacity were either uniformly random, random linear, or inherently algebraic.

Our result on list decoding follows from a much more general result: any *local* property (q.v. Definition 3.2.5) satisfied with high probability by a random linear code is also satisfied with high probability by a random LDPC code from Gallager's distribution.

4.1 LDPC Codes

Originally introduced by Gallager in the 1960's [Gal62], codes defined from graphs have become a class of central importance in the past 30 years. There are multiple ways to obtain a code from a graph; for now, we consider the following procedure. Suppose that $G = (V, W, E)$ is a bipartite graph with $|V| = n$ and $|W| = m$. Then G naturally defines a linear code $\mathcal{C} \leq \mathbb{F}_q^n$ of rate at least $1 - m/n$ as follows:

$$\mathcal{C} = \left\{ c \in \mathbb{F}_q^n : \sum_{i \in \Gamma(j)} \alpha_{i,j} c_i = 0 \; \forall j \in W \right\},$$

where $\Gamma(i)$ denotes the neighbors of i in G and $\alpha_{i,j} \in \mathbb{F}_q$ are fixed coefficients. That is, each vertex in W serves as a *parity check*, and the code is defined as all possible labelings of vertices in V which obey all of the parity checks. When the right-degree of G is small (ideally constant, independent of n), the resulting code is called a *low-density parity-check (LDPC)* code. Quantifying this, if every parity-check node has degree at most s, we say the code is s-LDPC.

4.1.1 Prior Work on LDPC Codes

LDPC Codes have been studied extensively in the context of unique decoding, especially in models of random errors. Informally, a code is said to achieve capacity on the Binary Symmetric Channel (BSC) if there is some algorithm which can, with high probability, uniquely decode a code of rate $R = 1 - h_2(\rho) - \varepsilon$ from a ρ-fraction of random bit-flips. It is known that Gallager's LDPC codes nearly achieve capacity on the BSC as n gets large, under maximum-likelihood decoding [Gal62; Gur06], and recently it was shown that related codes (specifically, *spatially-coupled* LDPC codes) achieve capacity for smaller block lengths under efficient decoding algorithms as well [KRU13]. Achieving capacity on the BSC appears to be related to achieving list-decoding capacity (in particular, the capacities are the same, $R = 1 - h_2(\rho)$). However, there is no formal connection along these lines, and to the best of our knowledge these results about the BSC do not imply anything about the list-decodability of LDPC codes.

As for the Binary Erasure Channel (BEC), where codes are said to achieve capacity if there is some algorithm which can, with high probability, uniquely decode a code of rate $R = 1 - \rho - \varepsilon$ from a ρ-fraction of bit erasures, a special kind of LDPC code with varying (but still constant) parity-check degree are known to achieve capacity with a linear-time decoding algorithm [Lub+01].

Another pleasing property of LDPC codes is that any such code can be encoded in linear time (assuming one is given a sparse parity-check matrix defining it) [LM09; KS12].

In the model of worst-case errors, LDPC codes (in particular, Tanner codes [Tan81] and expander codes [SS96; Zém01]) are notable for their efficient algorithms for unique decoding. In fact, the only asymptotically good codes with linear-time encoding and decoding algorithms we have are based on such codes.

However, despite our wealth of knowledge on LDPC codes, very little is known concerning their list-decodability.[1] For example, to the best of our knowledge we currently do not know of any purely combinatorial constructions of capacity-achieving list-decodable codes (q.v. Table 2.1 and the discussion in Section 2.5). Graph-based techniques have been used to modify a fixed underlying codes to obtain capacity-achieving codes, e.g., the distance amplification method of [AEL95] used by [HRZW17; Kop+19] employs an expander graph.

We also currently do not know of any linear-time algorithms to list-decode any code to capacity. Since graph-based codes and LDPC codes in particular are notable for their linear-time algorithms, they provide a natural candidate for a code which could have such efficient decoding.

This state of affairs motivates the following question:

Question 4.1.1. Are there (families of) LDPC codes that achieve list-decoding capacity?

[1] Beyond what can be deduced from the distance of such a code, i.e., the Johnson bound (Theorem 2.4.15).

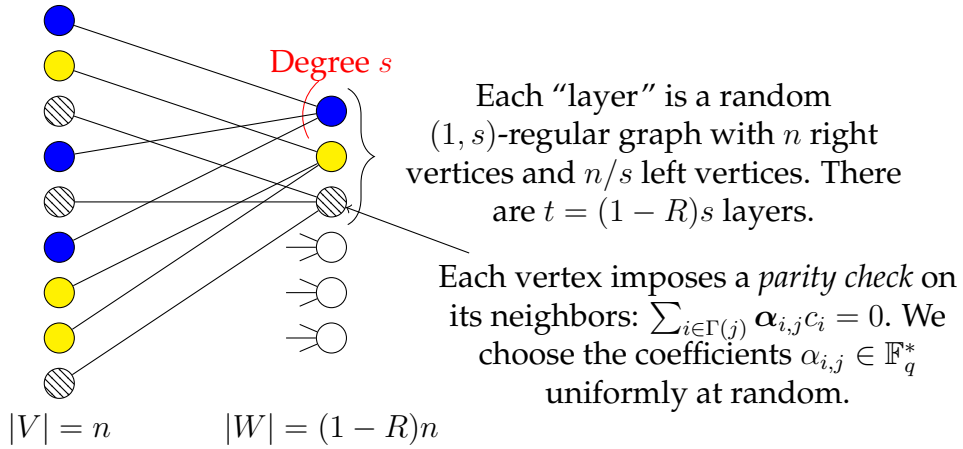

Figure 4.1: A random (t, s)-regular bipartite graph that gives rise to a random s-LDPC code of rate R. Here, we set $t := s(1 - R)$.

We show that the answer to Question 4.1.1 is a resounding yes: a random LDPC code sampled from (a generalization[2] of) Gallager's ensemble [Gal62] achieves list-decoding capacity with high probability. Before formally stating our results, we pause to introduce the random ensemble of LDPC codes we study.

Gallager's ensemble. Fix a rate $R \in (0, 1)$ and a sparsity parameter s, and let $t = (1 - R)s$.[3] To define the ensemble of random s-LDPC codes of rate R, we need to specify a distribution on the underlying bipartite graphs and a distribution on the coefficients $\alpha_{i,j}$. We define the distribution on graphs as follows. Let $G_i = (V, W_i, \boldsymbol{E}_i)$ for $i = 1, \ldots, t$ be independent uniformly random $(1, s)$-biregular bipartite graphs with a shared left vertex set V of size n and disjoint right vertex sets W_i: thus, $|W_i| = n/s$ for each i. Then let $\boldsymbol{G} = (V, W, \boldsymbol{E})$ be the union of these graphs, where $W = \bigsqcup_{i=1}^{t} W_i$. Finally, we choose the coefficients $\boldsymbol{\alpha}_{i,j}$ for $(i, j) \in \boldsymbol{E}$ to be uniformly random in $\mathbb{F}_q^*$. The ensemble of s-random LDPC codes of rate R is illustrated in Figure 4.1.

4.2 Our Results

Our main theorem about the list-decodability of random LDPC codes is a reduction from the list-decodability of random linear codes:

Theorem 4.2.1. *For any $R \in (0, 1)$, $\varepsilon > 0$, prime power q, and $L \geq 1$ there exists $s_0 \geq 1$ such that the following holds for any odd $s \geq s_0$. Suppose that a random linear code of rate R over $\mathbb{F}_q$ is (ρ, L)-list-decodable with high probability. Then a random s-LDPC code of rate $R - \varepsilon$ over $\mathbb{F}_q$ is (ρ, L)-list-decodable with high probability.*

[2]When $q = 2$ our definitions coincide. However, for larger q, our definitions differ in that we allow that $\alpha_{i,j}$'s to take random values in $\mathbb{F}_q^*$, whereas Gallager set each $\alpha_{i,j} = 1$.

[3]We assume that t is an integer.

Remark 4.2.2. All of our results actually hold for even s as well. In fact, when $q > 2$ the proofs work in this case equally well. To adapt the proof to deal with even s when $q = 2$ is doable, but tedious.

We can instatiate the above theorem with any of the results concerning the list-decodability of random linear codes discussed in Section 3.1. First of all, applying the Zyablov-Pinsker argument [ZP81] which shows that random linear codes are with high probability list-decodable up to capacity with constant list sizes, we conclude that LDPC codes also achieve list-decoding capacity with constant list sizes (note that as R, ε, q and L are all constant, the sparsity s is also constant). The title of this chapter is therefore justified.

Moreover, we can obtain better control of the list size. Recall that [GHK11] shows that random linear codes of rate $1-h_q(\rho)-\varepsilon$ are with high probability $(\rho, O_{\rho,q}(1/\varepsilon))$-list-decodable. Combining this result with Theorem 4.2.1 implies that for sufficiently large s, a random s-LDPC code of rate $1 - h_q(\rho) - \varepsilon$ is also $(\rho, O_{\rho,q}(1/\varepsilon))$-list-decodable with high probability.

Random LDPC codes achieve any local property that random linear codes achieve. In fact, Theorem 4.2.1 follows as a corollary of a much more general theorem. We show that any local property which is satisfied by random linear codes with high probability is also satisfied by random LDPC codes with high probability. For the definition of a local property, see Section 3.2, and in particular, Definition 3.2.5.

Our main theorem essentially states that every local property is approximately sharp for random s-LDPC codes, with approximately the same threshold as for random linear codes. This approximation improves as s grows. Recall the notation $\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R_n)$ from Chapter 3 for a random linear code of block length n and rate R_n. Below, we use the analogous notation $\boldsymbol{\mathcal{C}}^n_{s\mathrm{LDPC}}(R_n)$ for a random s-LDPC code of block length n and rate R_n. Also, recall that if $\mathcal{P}_n$ is a local property of length n codes, its threshold (q.v. Definition 3.3.1) is

$$R_{\mathrm{RLC}}(\mathcal{P}_n) := \sup\{R \in [0,1] : \mathbb{P}\left(\boldsymbol{\mathcal{C}}^n_{\mathrm{RLC}}(R) \text{ satisfies } \mathcal{P}_n\right) \geq 1/2\} .$$

Theorem 4.2.3 (Main). *Let $\mathcal{P} = (P_{n_i})_{i\in\mathbb{N}}$ be any ℓ-local property family such that*

$$\bar{R} := \limsup_{i\to\infty} R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) < 1 .$$

For any $\varepsilon > 0$ and prime power q, there exists $s_0 = s_0(\varepsilon, \bar{R}, q, \ell) \geq 1$ such that the following holds for any odd $s \geq s_0$. If $R_{n_i} \leq R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) - \varepsilon$ for all $i \in \mathbb{N}$, then

$$\lim_{i\to\infty} \mathbb{P}\left(\boldsymbol{\mathcal{C}}^{n_i}_{s\mathrm{LDPC}}(R_{n_i}) \text{ satisfies } P_{n_i}\right) = 1 .$$

Remark 4.2.4. Recall that for random linear codes of rate ε below the threshold, the probability the code failed to satisfy $\mathcal{P}$ was only $\exp(-\Omega(\varepsilon n))$ (q.v. Remark 3.3.10). One can ask if the same is true for s-LDPC codes. As it turns out, we will only establish failure probability which is inverse polynomial. However, this is tight (at least for the property of distance). See Remark 4.5.4.

Furthermore, as list-recovery is also a local property, as are average-radius list-decoding and recovery, we can instantiate Theorem 4.2.3 with any of the results discussed in Section 3.1. In particular, one can port over any result from Table 3.2.

For example, for the special case of $q = 2$, we will show in Chapter 6 (following [Gur+02; LW18]) that random linear codes of rate $1-h_2(\rho)-\varepsilon$ are $(\rho, h_2(\rho)/\varepsilon+1)$-average-radius list-decodable with high probability; see Theorem 6.1.1. Applying Theorem 4.2.1 with this result, making the sparsity s sufficiently large (but still independent of n), it follows that random s-LDPC codes of rate $1 - h_2(\rho) - \varepsilon$ are (ρ, L)-average-radius list-decodable with high probability, where (say) $L = 1.01h_2(\rho)/\varepsilon$.

For completeness, we now provide the formal justification for Theorem 4.2.1, assuming Theorem 4.2.3.

Proof of Theorem 4.2.1. Let $\mathcal{P} = (\mathcal{P}_{n_i})$ denote the property of (ρ, L)-list-decodability. Recall from Example 3.2.8 that this is an $(L+1)$-local property. Now, by the List-Decoding Capacity Theorem (Theorem 2.4.7), we know that for sufficiently large n_i there are no (ρ, L)-list-decodable codes of rate $1 - h_q(\rho) + \varepsilon$. In particular, this implies $\bar{R} := \limsup_{i\to\infty} R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) \le 1 - h_q(\rho) < 1$.

Now, suppose $R \in (0, 1)$ is such that a random linear code $\boldsymbol{\mathcal{C}}^{n_i}_{\mathrm{RLC}}(R)$ is (ρ, L)-list-decodable with high probability. By Theorem 3.3.9, it follows that $R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) \le R + o_{i\to\infty}(1)$. As $R - \varepsilon \le R_{\mathrm{RLC}}(\mathcal{P}_{n_i})$ for large enough n_i, Theorem 4.2.3 then guarantees that there exists $s_0 = s_0(\varepsilon, \bar{R}, q, \ell)$ such that for any odd $s \ge s_0$, a random s-LDPC code $\boldsymbol{\mathcal{C}}^{n_i}_{s\mathrm{LDPC}}(R-\varepsilon)$ satisfies $\mathcal{P}_{n_i}$ with high probability. That is, a random s-LDPC code of rate $R - \varepsilon$ is (ρ, L)-list-decodable with high probability. □

Before concluding this section, we pause to highlight the surprising nature of Theorem 4.2.3. There is a lot more structure in a random LDPC code than in a random linear code. For example, as mentioned earlier, random LDPC codes possess linear-time algorithms for unique-decoding,[4] but it is unlikely that any efficient unique decoding algorithm exists for random linear codes.[5] Thus it is quite shocking that this much more structured ensemble would share many properties—in a black box way—with random linear codes.

Remark 4.2.5 (A converse to Theorem 4.2.3?)**.** One may be tempted to conjecture that the converse of Theorem 4.2.3 holds as well. Namely, in the setting of Theorem 4.2.3, if $R_{n_i} \ge R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) + \varepsilon$ for all i, then the code ensemble $\boldsymbol{\mathcal{C}}^{n_i}_{s\mathrm{LDPC}}(R_{n_i})$ almost surely does

[4]This follows, for example, from [SS96] because the underlying random graph is with high probability a good expander.

[5]Unique decoding of random linear codes is related to the problem of Learning Noisy Parities (LNP) and Learning With Errors (LWE), which are thought to be hard.

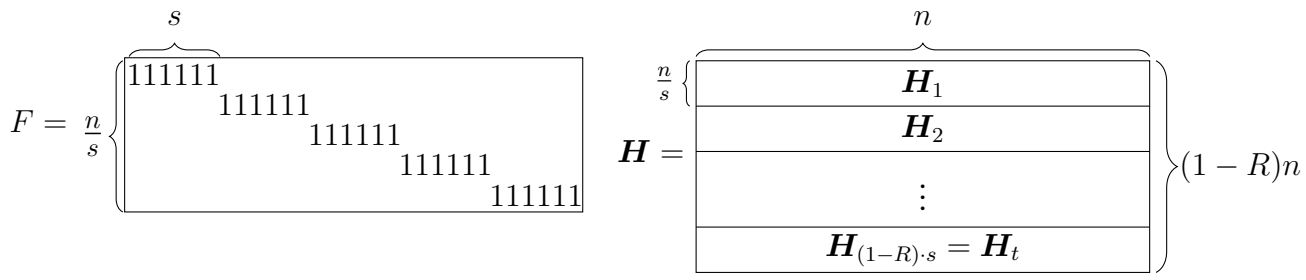

Figure 4.2: The matrices F and H. Each layer $\boldsymbol{H}_j$ of $\boldsymbol{H}$ is drawn independently according to the distribution $F\boldsymbol{\Pi}\boldsymbol{D}$, where $\boldsymbol{\Pi} \in \{0,1\}^{n\times n}$ is a random permutation and $\boldsymbol{D} \in \mathbb{F}_q^{n\times n}$ is a diagonal matrix with diagonal entries that are uniform in $\mathbb{F}_q^*$.

not satisfy $\mathcal{P}_{n_i}$. However, this turns out to be false. Consider the following example: let $q = 2$ and consider the 1-local property $\mathcal{P} := (\mathcal{P}_{n_i})_{i\in\mathbb{N}}$, where $\mathcal{P}_{n_i}$ is the set of all length n_i linear codes that only contain even weight codewords. It is not hard to see (e.g., using Theorem 3.3.9) that $R_{\mathrm{RLC}}(\mathcal{P}_{n_i})$ tends to 0 as $n \to \infty$. On the other hand, if $\frac{n_i}{s}$ is even, then every s-LDPC code of length n_i (including, say, a code of rate $\frac{1}{2}$) satisfies $\mathcal{P}_{n_i}$. Thus, $\limsup_{i\to\infty} R_{s\mathrm{LDPC}}(\mathcal{P}_{n_i}) > 0 = \limsup_{i\to\infty} R_{\mathrm{RLC}}(\mathcal{P}_{n_i})$, contradicting this conjecture.

However, the above counter-example relies on a technicality involving divisibility criteria. It is an interesting question whether a natural converse of Theorem 4.2.3 holds if we additionally assume that $\mathcal{P}$ belongs to some natural class of "nicely behaved" properties that precludes counter-examples of this sort.

4.3 The Proof, Modulo Two Technical Lemmas

In this section we give an overview of our proof strategy. First, despite previously defining our ensemble of LDPC codes in terms of a random graph, it will be more convenient to use the following parity-check matrix viewpoint. We introduce some notation and terminology to talk about the structure of $\boldsymbol{H}$ which we use throughout this chapter.

Let $F \in \{0,1\}^{(n/s)\times n}$ be the matrix $F = [\ F_1 \mid F_2 \mid \cdots \mid F_{n/s}\ ]$, where each $F_i \in \{0,1\}^{(n/s)\times n}$ has all 1's in its ith row, and the remaining rows are all 0's. Let $\boldsymbol{\Pi} \in \{0,1\}^{n\times n}$ be a uniformly random permutation matrix and let $\boldsymbol{D} \in \mathbb{F}_q^{n\times n}$ be a diagonal matrix whose entries are uniform and independent elements in $\mathbb{F}_q^*$. Let $\boldsymbol{H}_1, \ldots, \boldsymbol{H}_{(1-R)\cdot s}$ be independent samples from the distribution of $F\boldsymbol{\Pi}\boldsymbol{D}$. Then let $\boldsymbol{H}$ be the matrix obtained by stacking $\boldsymbol{H}_1, \ldots, \boldsymbol{H}_{(1-R)\cdot s}$ on top of each other. Then $\boldsymbol{H}$ is a parity-check matrix for a random s-LDPC code of rate R. We refer to each $\boldsymbol{H}_j$ as a "layer" of $\boldsymbol{H}$. This notation is summarized in Figure 4.2.

Our main result roughly follows by combining three building blocks. In the next three subsections we describe these components. In Section 4.3.4, we show how to deduce Theorem 4.2.3 from these parts.

4.3.1 Sharpness of Local Properties for Random Linear Codes

Fortunately, the first building block was already established in Chapter 3. Namely, we require the fact that local properties defined by types are sharp for random linear codes. We restate Lemma 3.3.8 for convenience.

Lemma 3.3.8. *Let $\ell \in \mathbb{N}$ and $\tau \in \mathcal{D}_{n,\ell}$. Denote $R^*_\tau = \max_{\tau' \in \mathcal{I}_\tau} R^{\mathbb{E}}_{\mathrm{RLC}}(\tau')$. The threshold $R_{\mathrm{RLC}}(\mathcal{P}^\tau)$ satisfies*

$$R_{\mathrm{RLC}}(\mathcal{P}^\tau) = R^*_\tau \pm o(1) \, .$$

Furthermore, suppose $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ is a random linear code of rate R.

1. *If $R \leq R^*_\tau - \varepsilon$ then*

$$\mathbb{P}\left(\tau \in \boldsymbol{\mathcal{C}}\right) \leq q^{-\varepsilon n} \, .$$

2. *Conversely, if $R \geq R^*_\tau + \varepsilon$ then*

$$\mathbb{P}\left(\tau \in \boldsymbol{\mathcal{C}}\right) \geq 1 - \binom{n + q^{2\ell} - 1}{q^{2\ell} - 1}^3 \cdot q^{-\varepsilon n} \, .$$

In fact, we only require the part of the lemma that characterizes $R_{\mathrm{RLC}}(\mathcal{P}^\tau)$ in terms of the set of implicit types $\mathcal{I}_\tau$. That is, the part of the lemma prior to the word "Furthermore".

4.3.2 Probability that a Matrix is Contained in a Random s-LDPC Code

The second component shows that, given a matrix $M \in \mathbb{F}_q^{n \times \ell}$, the probability that M is contained in a random s-LDPC code is not much larger than that of appearing in a random linear code, provided that M is sufficiently *smooth*.

Definition 4.3.1. Let $\delta > 0$. We say that a type τ over $\mathbb{F}_q^\ell$ is *δ-smooth* if

$$\mathop{\mathbb{P}}_{\boldsymbol{v} \sim \tau} \left(\langle u, \boldsymbol{v} \rangle \neq 0\right) \geq \delta \text{ for all } u \in \mathbb{F}_q^\ell \setminus \{0\} \, .$$

If $M \in \mathbb{F}_q^{\ell \times n}$ is such that τ_M is δ-smooth, we also say that M is *δ-smooth*.

Remark 4.3.2. In coding-theoretic terms, τ_M is δ-smooth if and only if the code $\{Mu : u \in \mathbb{F}_q^\ell\}$ has distance at least δ and M is full-rank. Indeed, the weight of any codeword Mu in this code is

$$\frac{1}{n} \sum_{i \in [n]} \mathbb{I}\left(\langle u, e_i^T M \rangle \neq 0\right) = \mathop{\mathbb{P}}_{\boldsymbol{v} \sim \tau} \left(\langle u, \boldsymbol{v} \rangle \neq 0\right) \, .$$

Remark 4.3.3 (Parity of s, again)**.** In order to adapt the proof to deal with even s when $q = 2$, one should also insist that $\mathbb{P}_{\boldsymbol{v} \sim \tau}\left(\langle u, \boldsymbol{v} \rangle = 0\right) \geq \delta$. Also, one needs the observation that a binary s-LDPC code $\mathcal{C}$ with s even always contains the all-1's vector, and hence, assuming $\mathcal{C}$ has distance δ, for all codewords c except for the all-1's vector, $\mathrm{wt}(c) \leq 1 - \delta$.

The following lemma bounds the probability that a smooth type is contained in a random LDPC code with sufficiently large sparsity parameter. The lemmas assumes that is a sufficiently large gap between the rate of the random s-LDPC code and the expectation threshold of τ (q.v. Definition 3.3.5). We prove this lemma in Section 4.4.

Lemma 4.3.4. *For any $\varepsilon, \delta > 0$, prime power q and $\ell \geq 1$ there exists $s_0 \geq 1$ such that the following holds for any odd $s \geq s_0$ and sufficiently large n. Suppose that $\tau \in \mathcal{D}_{n,\ell}$ is δ-smooth and $R \leq R_{\mathrm{RLC}}(\mathcal{P}^\tau) - \varepsilon$. Then, if $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ is a random s-LDPC code of rate R,*

$$\mathbb{P}\left(\tau \in \boldsymbol{\mathcal{C}}\right) \leq q^{-\varepsilon n/8} .$$

If we ignore the constraint that τ must be smooth, then together with Theorem 3.3.9 the above would imply Theorem 4.2.3. Indeed, if a type τ is unlikely to appear in a random linear code then Theorem 3.3.9 shows that some τ-implied type τ' appears $o(1)$ times in expectation in the random linear code. By Lemma 4.3.4, τ' also appears $o(1)$ times in a random LDPC code as well, so an LDPC code is unlikely to contain τ'. Thus, it is also unlikely to contain τ.

The proof of Lemma 4.3.4 proceeds by Fourier analysis, and we introduce the necessary tools in Section 4.4. The basic idea is as follows: since $\boldsymbol{\mathcal{C}}$ is a random s-LDPC code, each parity-check corresponds (essentially) to an independent and uniformly random set of s coordinates in $[n]$.[6] Thus, the probability that a matrix $M \in \mathcal{M}_\tau$ is in $\boldsymbol{\mathcal{C}}$ can be derived from the probability that s vectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_s \sim \tau$ sampled independently sum to zero. This probability is given by a convolution $\tau^{*s}(0) = \tau * \tau * \cdots * \tau(0)$ of τ with itself s times. The convolution is in turn controlled by sâĂŹth powers of the Fourier coefficients $\hat{\tau}(\xi)$ of τ. As we will see, the condition that τ be δ-smooth implies that the nonzero Fourier coefficients $\hat{\tau}(0)$ are bounded away from 1, and this means that if s is large enough, the contributions $\hat{\tau}(\xi)^s$ of the nonzero coefficients to $\tau^{*s}(0)$ will become small.

4.3.3 Distance of Random s-LDPC Codes

As noted above, the first two building blocks show that for any δ-smooth type $\tau \sim \mathbb{F}_q^\ell$, a random LDPC code of rate slightly below $R^n_{\mathrm{RLC}}(\mathcal{P}_\tau)$ is unlikely to contain τ. The third and final building block shows that we may restrict our attention to δ-smooth types.

As noted in Remark 4.3.2, the condition that M be δ-smooth is the same as the condition that the code generated by M has relative distance at least δ. Thus, if $\mathcal{C} \leq \mathbb{F}_q^n$ has relative distance at least δ, it does not contain any matrices that are not δ-smooth. Fortunately, it was already proved by Gallager [Gal62] that random *binary* s-LDPC codes have good distance; in fact, the distance approaches the Gilbert-Varshamov (GV) bound (Theorem 2.4.4) with high probability. Theorem 4.3.5 generalizes this result to s-LDPC codes over any alphabet.

[6] This is not exactly true because the parity checks that belong to the same layer are not independent; however, we show that this does not significantly affect the probability of the event of interest.

Theorem 4.3.5 (Random LDPC codes achieve the GV bound)**.** *For any $\delta \in (0, 1 - 1/q)$, $\varepsilon > 0$, and prime power q there exists $s_0 \geq 1$ such that the following holds for any $s \geq s_0$. A random s-LDPC code of rate $R \geq 1 - h_q(\delta) - \varepsilon$ over $\mathbb{F}_q$ has relative distance at least δ with high probability.*

Remark 4.3.6 (Comparison to Gallager's Proof)**.** Gallager's proof for binary random s-LDPC codes in [Gal62] uses generating functions. We give an alternative proof using ideas from exponential families, which follows the approach of recent work by Linial and Mosheiff [LM20]. Our proof extends to random s-LDPC codes over any alphabet. We note that Gallager left it as an open problem in [Gal62] to obtain a result like this for larger alphabets, but his definition was slightly different than ours: the coefficients $\alpha_{i,j}$ in his parity checks were all 1's, while ours are sampled uniformly from $\mathbb{F}_q^*$.

Despite having different frameworks, our proof and that of [Gal62] turn out to yield similar equations. In particular our proof of Lemma 4.5.2 is very similar to the corresponding proof in [Gal62] at a technical level. We highlight where the proofs diverge in Remark 4.5.9.

4.3.4 Proof of Theorem 4.2.3, Assuming the Building Blocks

Theorem 4.2.3, which we restate below, now follows as an immediate consequence of the building blocks above.

Theorem 4.2.3 (Main)**.** *Let $\mathcal{P} = (P_{n_i})_{i \in \mathbb{N}}$ be any ℓ-local property family such that*

$$\bar{R} := \limsup_{i \to \infty} R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) < 1 \, .$$

For any $\varepsilon > 0$ and prime power q, there exists $s_0 = s_0(\varepsilon, \bar{R}, q, \ell) \geq 1$ such that the following holds for any odd $s \geq s_0$. If $R_{n_i} \leq R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) - \varepsilon$ for all $i \in \mathbb{N}$, then

$$\lim_{i \to \infty} \mathbb{P}\left(\boldsymbol{\mathcal{C}}^{n_i}_{s\mathrm{LDPC}}(R_{n_i}) \text{ satisfies } P_{n_i}\right) = 1 \, .$$

Proof. Fix a sufficiently large integer s (depending on $\bar{R}$, ε, q and ℓ). Abbreviate $\boldsymbol{\mathcal{C}}_{n_i} = \boldsymbol{\mathcal{C}}^{n_i}_{s\mathrm{LDPC}}(R_{n_i})$ and let $\mathcal{T}_{n_i}$ be a collection of types defining $\mathcal{P}_{n_i}$. Let

$$\delta := \frac{h_q^{-1}(1 - \bar{R})}{2} > 0 \, .$$

Let $\mathcal{E}_{n_i}$ denote the event that the distance of $\boldsymbol{\mathcal{C}}_{n_i}$ is at most δ. As $\bar{R} = \limsup_{i \to \infty} R_{\mathrm{RLC}}(\mathcal{P}_{n_i})$, for sufficiently large n_i we have

$$R_{n_i} \leq R_{\mathrm{RLC}}(\mathcal{P}_{n_i}) - \varepsilon \leq 1 - h_q(\delta) - \varepsilon \, ,$$

and so Theorem 4.3.5 guarantees that $\lim_{i \to \infty} \mathbb{P}\left(\mathcal{E}_{n_i}\right) = 0$. Thus, to conclude the theorem, it suffices to show that

$$\mathbb{P}\left(\exists \tau \in \mathcal{T}_{n_i} \text{ s.t. } \tau \in \boldsymbol{\mathcal{C}}_{n_i} | \neg \mathcal{E}_{n_i}\right) = o(1) \, .$$

Partition $\mathcal{T}_{n_i} = \mathcal{T}_{n_i}^1 \sqcup \mathcal{T}_{n_i}^2$ such that $\mathcal{T}_{n_i}^1$ consists of δ-smooth types and $\mathcal{T}_{n_i}^2$ consists of the remaining types.

For any $\tau \in \mathcal{T}_{n_i}^2$, conditioned on $\neg\mathcal{E}_{n_i}$ it is guaranteed that $\tau \notin \boldsymbol{\mathcal{C}}_{n_i}$. Thus, we have

$$\begin{aligned}\mathbb{P}\left(\exists \tau \in \mathcal{T}_{n_i} \text{ s.t. } \tau \in \boldsymbol{\mathcal{C}}_{n_i} | \neg\mathcal{E}_{n_i}\right) &= \mathbb{P}\left(\exists \tau \in \mathcal{T}_{n_i}^1 \text{ s.t. } \tau \in \boldsymbol{\mathcal{C}}_{n_i} | \neg\mathcal{E}_{n_i}\right) \\ &\leq \frac{\mathbb{P}\left(\exists \tau \in \mathcal{T}_{n_i}^1 \text{ s.t. } \tau \in \boldsymbol{\mathcal{C}}_{n_i}\right)}{\mathbb{P}\left(\neg\mathcal{E}_{n_i}\right)} ;\end{aligned}$$

as $\mathbb{P}\left(\neg\mathcal{E}_{n_i}\right) = 1 - o(1)$, it will suffice to show $\mathbb{P}\left(\exists \tau \in \mathcal{T}_{n_i}^1 \text{ s.t. } \tau \in \boldsymbol{\mathcal{C}}_{n_i}\right) = o(1)$ to conclude the theorem. Take any $\tau \in \mathcal{T}_{n_i}^1$. Noting that $R_{\mathrm{RLC}}(\mathcal{P}^\tau) \geq R_{\mathrm{RLC}}(\mathcal{P}_{n_i})$, for large enough n_i we may apply Lemma 4.3.4 to conclude $\mathbb{P}\left(\tau \in \boldsymbol{\mathcal{C}}_{n_i}\right) \leq q^{-\varepsilon n_i/8}$.

Hence, applying a union bound and recalling (3.3),

$$\mathbb{P}\left(\exists \tau \in \mathcal{T}_{n_i}^1 \text{ s.t. } \tau \in \boldsymbol{\mathcal{C}}_{n_i}\right) \leq |\mathcal{T}_{n_i}^1| \cdot q^{-\varepsilon n_i/8} \leq \binom{n_i + q^\ell - 1}{q^\ell} q^{-\varepsilon n_i/8}$$

which indeed tends to 0 as $i \to \infty$. □

4.4 Probability Smooth Types Appear in LDPC Codes

In this section, we prove that smooth types are unlikely to appear in a random LDPC code, assuming they are sufficiently rare. We restate Lemma 4.3.4 for convenience.

Lemma 4.3.4. *For any $\varepsilon, \delta > 0$, prime power q and $\ell \geq 1$ there exists $s_0 \geq 1$ such that the following holds for any odd $s \geq s_0$ and sufficiently large n. Suppose that $\tau \in \mathcal{D}_{n,\ell}$ is δ-smooth and $R \leq R_{\mathrm{RLC}}(\mathcal{P}^\tau) - \varepsilon$. Then, if $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ is a random s-LDPC code of rate R,*

$$\mathbb{P}\left(\tau \in \boldsymbol{\mathcal{C}}\right) \leq q^{-\varepsilon n/8} .$$

The main technical lemma in the proof of Lemma 4.3.4 shows that the probability that a smooth matrix is contained in a random LDPC with sufficiently large sparsity parameter is roughly the same as in a random linear code (cf., Proposition 2.2.2).

Lemma 4.4.1. *For any $\varepsilon, \delta > 0$, prime power q and $\ell \geq 1$ there exists $s_0 \geq 1$ such that the following holds for any odd $s \geq s_0$ and sufficiently large n. Let $M \in \mathbb{F}_q^{n \times \ell}$ be a δ-smooth matrix. Then, if $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ is a random s-LDPC code of rate R,*

$$\mathbb{P}\left(M \subset \boldsymbol{\mathcal{C}}\right) \leq q^{-(1-\varepsilon/4)(1-R)n\ell} .$$

That is, up to the multiplicative $(1 - \varepsilon/4)$ term in the exponent, the probability the matrix lies in the random LDPC matrix is no greater than the probability it lies in a random linear code. First, we show how Lemma 4.4.1 implies Lemma 4.3.4.

Proof that Lemma 4.4.1 implies Lemma 4.3.4. Applying Lemma 3.3.8, we may choose $\tau' \in \mathcal{I}_\tau$ such that $R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') \geq R_{\mathrm{RLC}}(\mathcal{P}_\tau) - \eta(n)$ where $\eta(n) \to 0$ as $n \to \infty$. We assume n is large enough so that $\eta(n) \leq \frac{\varepsilon}{2}$.

Without loss of generality, we may assume that $\mathrm{rank}(A) = m$, i.e., $d(\tau') = m$. Otherwise, we can replace A with a submatrix $A' \in \mathbb{F}_q^{\mathrm{rank}(A)\times \ell}$ whose rows are a basis for $\text{row-span}(A)$, noting that the distribution τ'' given by $A'\boldsymbol{v}$ for $\boldsymbol{v} \sim \tau$ satisfies $H_q(\tau'') = H_q(\tau')$ and $d(\tau'') = d(\tau')$.

Next, observe that the distribution τ' is δ-smooth. Indeed, for any $u \in \mathbb{F}_q^m \setminus \{0\}$ we have

$$\mathop{\mathbb{P}}_{\boldsymbol{v}'\sim\tau'}\left(\langle u, \boldsymbol{v}'\rangle \neq 0\right) = \mathop{\mathbb{P}}_{\boldsymbol{v}\sim\tau}\left(\langle u, A\boldsymbol{v}\rangle \neq 0\right) = \mathop{\mathbb{P}}_{\boldsymbol{v}\sim\tau}\left(\langle A^\top u, \boldsymbol{v}\rangle \neq 0\right) .$$

As A has rank m, $A^\top u \neq 0$, so $\mathbb{P}_{\boldsymbol{v}\sim\tau}\left(\langle A^\top u, \boldsymbol{v}\rangle \neq 0\right) \geq \delta$.

By Lemma 4.4.1, for any matrix $M' \in \mathcal{M}_{\tau'}$, we have

$$\mathbb{P}\left(M' \subseteq \boldsymbol{\mathcal{C}}\right) \leq q^{-(1-\varepsilon/4)(1-R)nm} ,$$

and so the probability $\boldsymbol{\mathcal{C}}$ contains some matrix in $\mathcal{M}_{\tau'}$ is at most

$$\begin{aligned} |\mathcal{M}_{\tau'}| \cdot q^{-(1-\varepsilon/4)(1-R)mn} &\leq q^{(H_q(\tau')-(1-\varepsilon/4)(1-R)m)n} \\ &\leq q^{((1-R-\varepsilon/2)-(1-\varepsilon/4)(1-R))mn} \leq q^{-\varepsilon n/8} , \end{aligned}$$

where the first inequality uses Proposition 3.2.4 and the second inequality follows from

$$R \leq R_{\mathrm{RLC}}(\mathcal{P}^\tau) - \varepsilon \leq R^{\mathbb{E}}_{\mathrm{RLC}}(\tau') - \frac{\varepsilon}{2} = 1 - \frac{H_q(\tau')}{m} - \frac{\varepsilon}{2} .$$

Thus, we find that $\boldsymbol{\mathcal{C}}$ contains τ' with probability at most $q^{-\varepsilon n/8}$. Since containing the type τ implies containing the type τ', the lemma follows. □

We now provide the proof of Lemma 4.4.1. This proof employs Fourier analysis over finite fields; we provide the necessary definitions and facts below.

4.4.1 Fourier Analysis over Finite Fields

We refer the reader to, for example, [LN97; O'D14] for more details and proofs of these facts. In what follows assume that $q = p^h$ for a prime p. Recall the definition of the *trace map* of $\mathbb{F}_q$ over $\mathbb{F}_p$:

$$\mathrm{Tr}(\alpha) := \alpha + \alpha^p + \alpha^{p^2} + \cdots + \alpha^{p^{h-1}} .$$

For a function $f : \mathbb{F}_q^n \to \mathbb{C}$, we define the *Fourier transform* $\hat{f} : \mathbb{F}_q^n \to \mathbb{C}$ of f by

$$\hat{f}(\xi) := \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathbb{F}_q^n}\left[f(\boldsymbol{x}) \cdot \overline{\chi_\xi(\boldsymbol{x})}\right] ,$$

where $\chi_\xi(x) = \omega_p^{\mathrm{Tr}(\langle \xi, x \rangle)}$ and $\omega_p = e^{2\pi i/p}$ is a primitive p-th root of unity. Then we have the decomposition

$$f(x) = \sum_{\xi \in \mathbb{F}_q^n} \hat{f}(\xi)\chi_\xi(x) \, .$$

For two functions $f, g : \mathbb{F}_q^n \to \mathbb{C}$, we define their inner product by

$$\langle f, g \rangle := \mathop{\mathbb{E}}_{\boldsymbol{x} \sim \mathbb{F}_q^n} \left[f(\boldsymbol{x}) \overline{g(\boldsymbol{x})} \right] \, .$$

Plancherel's identity then asserts that

$$\langle f, g \rangle = \sum_{\xi \in \mathbb{F}_q^n} \hat{f}(\xi) \overline{\hat{g}(\xi)} \, .$$

An important special case is Parseval's identity:

$$\langle f, f \rangle = \sum_{\xi \in \mathbb{F}_q^n} \left| \hat{f}(\xi) \right|^2 \, .$$

The *convolution* of a pair of functions $f, g : \mathbb{F}_q^n \to \mathbb{C}$ is given by

$$(f * g)(x) = \mathop{\mathbb{E}}_{\boldsymbol{y} \sim \mathbb{F}_q^n} [f(\boldsymbol{y}) g(x - \boldsymbol{y})]$$

Convolution interacts nicely with the Fourier transform:

$$\widehat{f * g}(x) = \hat{f}(x) \cdot \hat{g}(x) \, .$$

As a useful piece of notation, we define inductively $f^{*1} := f$ and $f^{*s} := f^{*(s-1)} * f$.

Finally, we state the following claim and, for lack of a suitable reference, provide the proof (although this fact is certainly well-known). It allows us to write the probability that a sum of i.i.d. random variables from $\mathbb{F}_q^\ell$ takes a certain value in terms of the convolution its density function.

Claim 4.4.2. *Let $P \sim \mathbb{F}_q^\ell$ be a distribution, and let $f(x) = q^\ell \cdot P(x)$ be the density of P with respect to the uniform distribution. For any $y \in \mathbb{F}_q^\ell$ and $s \geq 1$, if $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s \sim P$ are independent,*

$$\mathbb{P}\left(\sum_{i=1}^{s} \boldsymbol{u}_i = y \right) = q^{-\ell} \cdot f^{*s}(y) \, .$$

Proof. By induction on s. The case $s = 1$ follows by the definition of $f = f^{*1}$, so we now

assume $s > 1$. Let $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_s$ be independent samples from P.

$$\begin{aligned}
\mathbb{P}\left(\sum_{i=1}^{s} \boldsymbol{u}_i = y\right) &= \sum_{u \in \mathbb{F}_q^\ell} \mathbb{P}\left(\boldsymbol{u}_s = u\right) \cdot \mathbb{P}\left(\sum_{i=1}^{s-1} \boldsymbol{u}_i = y - u \middle| \boldsymbol{u}_s = u\right) \\
&= \sum_{u \in \mathbb{F}_q^\ell} q^{-\ell} f(u) \cdot q^{-\ell} f^{*(s-1)}(y-u) \\
&= q^{-\ell} \cdot \mathop{\mathbb{E}}_{\boldsymbol{u} \sim \mathbb{F}_q^\ell} \left[f(\boldsymbol{u}) \cdot f^{*(s-1)}(y - \boldsymbol{u})\right] \\
&= q^{-\ell} \cdot \left(f * f^{*(s-1)}\right)(y) \\
&= q^{-\ell} \cdot f^{*s}(y) \,.
\end{aligned}$$

In the second equality, we used the induction hypothesis. □

4.4.2 Proof of Lemma 4.4.1

Having established the necessary definitions and notations, we may now prove Lemma 4.4.1.

Proof. Let $\boldsymbol{H} \in \mathbb{F}_q^{(1-R)n \times n}$ be the parity-check matrix of $\boldsymbol{\mathcal{C}}$ with layers $\boldsymbol{H}_1, \ldots, \boldsymbol{H}_{(1-R)s}$. Recall that each layer $\boldsymbol{H}_j$ is an independent sample of $F\boldsymbol{D}\boldsymbol{\Pi}$, where F is as in Figure 4.2, $\boldsymbol{\Pi} \in \{0,1\}^n$ is a uniformly random permutation matrix, and $\boldsymbol{D} \in \mathbb{F}_q^{n \times n}$ is a uniformly random diagonal matrix of rank n (i.e., all the diagonal entries are i.i.d. uniform samples from $\mathbb{F}_q^*$). Let $\boldsymbol{\Lambda}$ be a random matrix sampled according to the distribution $\boldsymbol{D}\boldsymbol{\Pi}M$. Then by independence of the layers,

$$\begin{aligned}
\mathbb{P}\left(M \subset \boldsymbol{\mathcal{C}}\right) &= \mathbb{P}\left(\boldsymbol{H}M = 0\right) \\
&= \mathbb{P}\left(\boldsymbol{H}_1 M\right)^{(1-R)s} \\
&= \mathbb{P}\left(F\boldsymbol{\Pi}\boldsymbol{D}M = 0\right)^{(1-R)s} \\
&= \mathbb{P}\left(F\boldsymbol{\Lambda} = 0\right)^{(1-R)s} \,. \qquad (4.1)
\end{aligned}$$

So it suffices to bound the probability that $F\boldsymbol{\Lambda} = 0$.

Now, observe that the marginal distribution of each row of $\boldsymbol{\Lambda}$ is given by $\boldsymbol{\lambda}\boldsymbol{v}$ for $\boldsymbol{v} \sim \tau_M$ and uniform $\boldsymbol{\lambda} \sim \mathbb{F}_q^*$. Denote this distribution on $\mathbb{F}_q^\ell$ by P. Let $\boldsymbol{\Lambda}' \in \mathbb{F}_q^{n \times \ell}$ denote a random matrix obtained by sampling each row independently according to P. We claim that

$$\mathbb{P}\left(F\boldsymbol{\Lambda} = 0\right) \leq O\left(n^{\frac{q^\ell - 1}{2}}\right) \cdot \mathbb{P}\left(F\boldsymbol{\Lambda}' = 0\right) \,. \qquad (4.2)$$

Indeed,

$$\mathbb{P}\left(F\boldsymbol{\Lambda} = 0\right) = \mathbb{P}\left(F\boldsymbol{\Lambda}' = 0 | \tau_{\boldsymbol{\Lambda}'} = \tau_{\boldsymbol{M}}\right) = \frac{\mathbb{P}\left(F\boldsymbol{\Lambda}' = 0 \wedge \tau_{\boldsymbol{\Lambda}'} = \tau_{\boldsymbol{M}}\right)}{\mathbb{P}\left(\tau_{\boldsymbol{\Lambda}'} = \tau_{\boldsymbol{M}}\right)} \leq \frac{\mathbb{P}\left(F\boldsymbol{\Lambda}' = 0\right)}{\mathbb{P}\left(\tau_{\boldsymbol{\Lambda}'} = \tau_{\boldsymbol{M}}\right)} \,.$$

Now, enumerating $\mathbb{F}_q^\ell = \{v_1, \ldots, v_{q^\ell}\}$, we have

$$\begin{aligned}\mathbb{P}\left(\tau_{\boldsymbol{M}} = \tau_{\boldsymbol{\Lambda}'}\right) &= \binom{n}{\tau(v_1)n, \ldots, \tau(v_{q^\ell})n} \cdot \prod_{j=1}^{q^\ell} \tau_M(v_j)^{\tau_M(v_j)n} \\ &= \binom{n}{\tau(v_1)n, \ldots, \tau(v_{q^\ell})n} \cdot q^{-H_q(\tau)n} \,,\end{aligned}$$

and so (4.2) follows from our estimate for multinomial coefficients, Proposition 3.2.4.

Thus, we are reduced to bounding $\mathbb{P}\left(F\boldsymbol{\Lambda}' = 0\right)$. Let $f(x) = q^\ell \cdot P(x)$ be the density of P with respect to the uniform distribution on $\mathbb{F}_q^\ell$. By the independence of the rows of $\boldsymbol{\Lambda}'$ and Claim 4.4.2:

$$\mathbb{P}\left(F\boldsymbol{\Lambda}' = 0\right) = \left(\mathop{\mathbb{P}}_{\boldsymbol{u}_1,\ldots,\boldsymbol{u}_s \sim P}\left(\sum_{i=1}^{s} \boldsymbol{u}_i = 0\right)\right)^{n/s} = \left(q^{-\ell} \cdot f^{*s}(0)\right)^{n/s} \,. \tag{4.3}$$

We are therefore reduced to bounding $P^{*s}(0)$. In terms of the Fourier transform, we can write

$$f^{*s}(0) = \sum_{\xi \in \mathbb{F}_q^\ell} \widehat{f^{*s}}(\xi) \cdot \overline{\chi_\xi(0)} = \sum_{\xi \in \mathbb{F}_q^\ell} \left(\hat{f}(\xi)\right)^s \,.$$

We now proceed to bound each term in the above sum. As f is a density, $\hat{f}(0) = 1$. For $\xi \neq 0$, we claim the following:

Claim 4.4.3. *For any $\xi \in \mathbb{F}_q^\ell \setminus \{0\}$, $\hat{f}(\xi) \in \mathbb{R}$ and*

$$\hat{f}(\xi) \leq 1 - \frac{q-1}{q} \cdot \delta \,.$$

Proof of Claim 4.4.3. We have

$$\begin{aligned}\hat{f}(\xi) &= \mathop{\mathbb{E}}_{\boldsymbol{x} \sim \mathbb{F}_q^\ell}\left[f(\boldsymbol{x})\chi_\xi(\boldsymbol{x})\right] \\ &= \mathop{\mathbb{E}}_{\boldsymbol{x} \sim P}\left[\chi_\xi(\boldsymbol{x})\right] = \mathop{\mathbb{E}}_{\boldsymbol{v} \sim \tau_M, \boldsymbol{\lambda} \sim \mathbb{F}_q^*}\left[\omega_p^{\mathrm{Tr}(\langle \boldsymbol{\lambda}\boldsymbol{v}, \xi\rangle)}\right] \\ &= \mathop{\mathbb{P}}_{\boldsymbol{v} \sim \tau_M}\left(\langle \boldsymbol{v}, \xi\rangle = 0\right) \mathop{\mathbb{E}}_{\boldsymbol{\lambda} \sim \mathbb{F}_q^*}\left[\omega_p^0\right] + \mathop{\mathbb{P}}_{\boldsymbol{v} \sim \tau_M}\left(\langle \boldsymbol{v}, \xi\rangle = 0\right) \mathop{\mathbb{E}}_{\boldsymbol{\lambda} \sim \mathbb{F}_q^*}\left[\omega_p^{\boldsymbol{\lambda}}\right] \\ &= \mathop{\mathbb{P}}_{\boldsymbol{v} \sim \tau_M}\left(\langle \boldsymbol{v}, \xi\rangle = 0\right) \cdot 1 + \mathop{\mathbb{P}}_{\boldsymbol{v} \sim \tau_M}\left(\langle \boldsymbol{v}, \xi\rangle = 0\right) \cdot \frac{-1}{q-1} \\ &\leq (1-\delta) - \delta \cdot \frac{1}{q-1} = 1 - \frac{q-1}{q} \cdot \delta \,,\end{aligned}$$

where the inequality follows from the assumption that M is δ-smooth. To justify the identity $\mathbb{E}_{\boldsymbol{\lambda} \sim \mathbb{F}_q^*}\left[\omega_p^{\boldsymbol{\lambda}}\right] = -\frac{1}{q-1}$, recall that $\mathrm{Tr} : \mathbb{F}_q \to \mathbb{F}_p$ is an h-to-1 map[7] and $\mathrm{Tr}(0) = 0$.

[7]Recall that we have $q = p^h$ in this section.

Therefore,

$$\mathop{\mathbb{E}}_{\boldsymbol{\lambda}\sim\mathbb{F}_q^*}\left[\omega_p^{\boldsymbol{\lambda}}\right] = \frac{1}{q-1}\sum_{\lambda\in\mathbb{F}_q^*}\omega_p(\lambda) = \frac{1}{q-1}\cdot\left((h-1)\omega_p^0 + \sum_{j=1}^{p-1} h\omega_p^j\right)$$
$$= \frac{1}{q-1}\cdot\left(\sum_{j=1}^{p-1}\omega^j\right) = \frac{1}{q-1}\cdot(-1)\,. \qquad \square$$

Returning to the proof of the lemma, recalling the assumption that s is odd, we obtain

$$f^{*s}(0) = \left(\hat{f}(0)\right)^s + \sum_{\xi\in\mathbb{F}_q^\ell\setminus\{0\}}\left(\hat{f}(\xi)\right)^s \le 1 + \left(1 - \frac{q-1}{q}\cdot\delta\right)^s\,,$$

and so, recalling Equations Eq. (4.1), (4.2) and (4.3), we conclude

$$\mathbb{P}\left(M\subseteq\boldsymbol{\mathcal{C}}\right) \le O\left(n^{\frac{q^\ell-1}{2}\cdot(1-R)\cdot s}\right)\cdot\left(q^{-\ell} + \left(1-\frac{q-1}{q}\cdot\delta\right)^s\right)^{(1-R)\cdot n} \le q^{-(1-\varepsilon/4)(1-R)\ell n}\,,$$

where the last inequality holds for large enough s depending on δ, ε, q and ℓ, as well as sufficiently large n. $\square$

Remark 4.4.4 (The choice of s)**.** An inspection of the proof shows that we may take

$$s_0 = O\left(\frac{\ell}{\log_q\left(\frac{1}{1-\delta/(1-1/q)}\right)}\right)\,.$$

In particular, noting that

$$\log_q\left(\frac{1}{1-\delta/(1-1/q)}\right) = \frac{1}{\ln(q)}\sum_{i=1}^{\infty}\frac{1}{i}\left(\frac{\delta}{1-1/q}\right)^i\,,$$

this part of the proof requires us to take

$$s_0 \ge C_0\cdot\frac{\ell\log(q)}{\delta}$$

for some constant $C_0 > 0$. There is one other place in the proof of Theorem 4.2.3 that requires s_0 to be sufficiently large; we comment on this in Remark 4.5.3.

4.5 Distance

In this section we prove Theorem 4.3.5, which shows that an LDPC code over any alphabet approaches the Gilbert-Varshamov bound with high probability. We restate the theorem below.

Theorem 4.3.5 (Random LDPC codes achieve the GV bound)**.** *For any $\delta \in (0, 1 - 1/q)$, $\varepsilon > 0$, and prime power q there exists $s_0 \geq 1$ such that the following holds for any $s \geq s_0$. A random s-LDPC code of rate $R \geq 1 - h_q(\delta) - \varepsilon$ over $\mathbb{F}_q$ has relative distance at least δ with high probability.*

4.5.1 Proof of Theorem 4.3.5, given a lemma

In this section we give an outline of the proof of Theorem 4.3.5 and prove the theorem based on Lemma 4.5.2 that we state below and prove in subsequent subsections.

Our goal is to show that a random s-LDPC code $\boldsymbol{\mathcal{C}}$ has good distance, or equivalently that there are no low-weight codewords in $\boldsymbol{\mathcal{C}}$ with high probability. To that end, we introduce the following notation.

Definition 4.5.1. For $\lambda \in (0, 1)$ such that λn is an integer, let $P_\lambda = \mathbb{P}(u \in \boldsymbol{\mathcal{C}})$, where $u \in \mathbb{F}_q^n$ has weight λ. Note that this probability is the same for every u of weight λ, so P_λ is well-defined.

Our main challenge is to find sufficiently tight upper bounds on these terms P_λ for $0 < \lambda \leq \delta$. The proof proceeds by giving a bound on P_λ in terms of a certain function $\varphi : (0, \frac{q-1}{q}] \to \mathbb{R}_{\leq 0}$. We will define φ below in Section 4.5.2, but for now we introduce its important properties in the following lemma. (The proof of this lemma appears in Sections 4.5.2 and 4.5.3.)

Lemma 4.5.2. *There is a function $\varphi : \left(0, \frac{q-1}{q}\right] \to \mathbb{R}_{\leq 0}$ which has the following properties.*

1. *For every $\lambda \in \left(0, 1 - \frac{1}{q}\right]$,*
$$\log_q P_\lambda \leq \varphi(\lambda)(1 - R)n \, .$$
2. *The function φ satisfies*
$$\varphi(\lambda) \leq \log_q \left(1 + (q-1)\left(1 - \frac{q}{q-1}\lambda\right)^s\right) - 1$$
for all $\lambda \in (0, \frac{q-1}{q}]$.
3. *The function $\frac{\varphi(\lambda)}{h_q(\lambda)}$ is strictly increasing in the range $0 < \lambda \leq \frac{q-1}{q}$.*

Before we prove Lemma 4.5.2, we show how it implies Theorem 4.3.5.

Proof of Theorem 4.3.5. Our goal is to show that if $\boldsymbol{\mathcal{C}}$ is a random s-LDPC code as in the statement of Theorem 4.3.5, then with high probability there are no codewords in $\mathcal{C}$ of relative weight less than δ. In the following, we assume without loss of generality that

δn is an integer. Now

$$\begin{aligned}
\mathbb{P}\left(\boldsymbol{\mathcal{C}} \text{ has relative distance less than } \delta\right) &\leq \sum_{i=1}^{\delta n} P_{\frac{i}{n}} \left|\left\{u \in \mathbb{F}_q^n : \mathrm{wt}(u) = \tfrac{i}{n}\right\}\right| && (4.4)\\
&\leq \sum_{i=1}^{\delta n} P_{\frac{i}{n}} q^{n h_q\left(\frac{i}{n}\right)} \\
&\leq \sum_{i=1}^{\delta n} q^{\left(\varphi\left(\frac{i}{n}\right)(1-R)+h_q\left(\frac{i}{n}\right)\right)n} && (4.5)\\
&= \sum_{i=1}^{\delta n} q^{n h_q\left(\frac{i}{n}\right)\left(\frac{(1-R)\varphi\left(\frac{i}{n}\right)}{h_q\left(\frac{i}{n}\right)}+1\right)} && (4.6)\\
&\leq \sum_{i=1}^{\delta n} q^{n h_q\left(\frac{i}{n}\right)\left(\frac{(1-R)\varphi(\delta)}{h_q(\delta)}+1\right)}. && (4.7)
\end{aligned}$$

Above, (4.4) follows from the union bound, (4.5) from Item 1 of Lemma 4.5.2, and (4.7) from Item 3 of Lemma 4.5.2. By Item 2 of Lemma 4.5.2,

$$\frac{(1-R)\varphi(\delta)}{h_q(\delta)} + 1 = \frac{(1-R)\cdot\left(\log_q\left(1+(q-1)\left(1-\frac{q}{q-1}\delta\right)^s\right)-1\right)}{h_q(\delta)} + 1\,.$$

Recall our hypothesis that the rate of the code satisfies $R \leq 1 - h_q(\delta) - \varepsilon$, and so $1 - R \geq h_q(\delta) + \varepsilon$. Noting that $\log_q\left(1+(q-1)\left(1-\frac{q}{q-1}\delta\right)^s\right) - 1 \leq 0$, we may thus bound the right hand side from above by

$$\begin{aligned}
&\frac{(h_q(\delta)+\varepsilon)\cdot\left(\log_q\left(1+(q-1)\left(1-\frac{q}{q-1}\delta\right)^s\right)-1\right)}{h_q(\delta)} + 1\\
&= \left(1+\frac{\varepsilon}{h_q(\delta)}\right)\cdot\left(\log_q\left(1+(q-1)\left(1-\frac{q}{q-1}\delta\right)^s\right)-1\right)+1\\
&= \left(1+\frac{\varepsilon}{h_q(\delta)}\right)\cdot\log_q\left(1+(q-1)\left(1-\frac{q}{q-1}\delta\right)^s\right) - \frac{\varepsilon}{h_q(\delta)}\\
&\leq \left(1+\frac{\varepsilon}{h_q(\delta)}\right)\frac{(q-1)}{\ln(q)}\left(1-\frac{q\delta}{q-1}\right)^s - \frac{\varepsilon}{h_q(\delta)}.
\end{aligned}$$

Thus, as long as s is sufficiently large in terms of δ, ε and q, we conclude that

$$\frac{(1-R)\varphi(\delta)}{h_q(\delta)} + 1 \leq -\frac{\varepsilon}{2h_q(\delta)} \leq -\frac{\varepsilon}{2}.$$

Hence, the right-hand side of Eq. (4.7) is upper bounded by

$$\sum_{i=1}^{\delta n} q^{-\frac{n h_q\left(\frac{i}{n}\right)\varepsilon}{2}}\,.$$

This sum is dominated by its first term, so it is at most $O(n^{-\Omega(1)})$. □

Remark 4.5.3 (The choice of s)**.** An inspection of the proof above shows that it suffices to take $s \gtrsim \ln(q/\varepsilon)/\delta$. Thus, this part of the proof requires that $s_0 \gtrsim \ln(q/\varepsilon)/\delta$.

Remark 4.5.4 (Polynomially small failure probability)**.** In the proof, we see that the failure probability, while $o(1)$, is only polynomially small in n. In fact, this is tight: it is not hard to see that an s-random LDPC code $\boldsymbol{\mathcal{C}}$ (for $s = O(1)$) contains a codeword of weight 2 with inverse polynomial probability. Specifically, consider the vector $v = (1, 1, 0, \ldots, 0)$. For each $j \in [t]$, we have $\mathbb{P}(\boldsymbol{H}_j v = 0) \geq \frac{s-1}{(q-1)n}$: first, with probability at least $(s-1)/n$ the vertices 1 and 2 are adjacent to the same right vertex in W_j, say i; then, with probability $\frac{1}{q-1}$, $\boldsymbol{\alpha}_{1,i} = -\boldsymbol{\alpha}_{2,i}$. Thus, with probability $O(n^{-t})$[8] we have $\boldsymbol{H}_j v = 0$ for all $j \in [t]$, i.e., $v \in \boldsymbol{\mathcal{C}}$.

4.5.2 The Function φ and Proof of Items 1 and 2 of Lemma 4.5.2

Let $\lambda \in \left(0, \frac{q-1}{q}\right]$ such that λn is an integer, and let $u \in \mathbb{F}_q^n$ have weight λn. Let $\boldsymbol{H}_1, \ldots, \boldsymbol{H}_t$ be the layers of the parity-check matrix $\boldsymbol{H}$ of $\boldsymbol{\mathcal{C}}$, as in Fig. 4.2.

Note that the matrices $\boldsymbol{H}_1, \ldots, \boldsymbol{H}_t$ are identically and independently distributed. In particular, the events $\mathbb{P}(\boldsymbol{H}_j u = 0)$ are independent. Hence,

$$P_\lambda = \mathbb{P}(u \in \boldsymbol{\mathcal{C}}) = \mathbb{P}(\boldsymbol{H} u = 0) = \mathbb{P}(\boldsymbol{H}_1 u = 0)^t . \tag{4.8}$$

Recall that $\boldsymbol{H}_1$ is sampled from the distribution $F\boldsymbol{\Pi}\boldsymbol{D}$, where $\boldsymbol{\Pi}$ is a random permutation and $\boldsymbol{D}$ is a random full-rank diagonal matrix (cf. Figure 4.2). Note that $\boldsymbol{\Pi}\boldsymbol{D}u$ is uniform over the set of weight λ vectors in $\mathbb{F}_q^n$. Hence, if $\boldsymbol{u}$ is uniform over the set of weight λ vectors,

$$P_\lambda = \mathbb{P}(F\boldsymbol{u} = 0)^t .$$

We turn to bound this expression. Let $\beta \in \left(0, \frac{q-1}{q}\right]$. Denote by $\mu_q(\beta)$ the distribution on $\mathbb{F}_q$ which is 0 with probability $1-\beta$ and uniform on $\mathbb{F}_q^*$ with probability β. When β is clear from context, we shorthand $\mu_q = \mu_q(\beta)$. Let $\mu_q^n \sim \mathbb{F}_q^n$ denote the distribution obtained by sampling each coordinate independently according to μ_q, and let $\boldsymbol{v} \sim \mu_q^n$. Observe that the distribution of $\boldsymbol{v}$, conditioned on $\mathrm{wt}(\boldsymbol{v}) = \lambda$, is identical to the distribution of $\boldsymbol{u}$. Hence, by Bayes' rule,

$$\begin{aligned}
\mathbb{P}(F\boldsymbol{u} = 0) &= \mathbb{P}(F\boldsymbol{v} = 0 \mid \mathrm{wt}(\boldsymbol{v}) = \lambda) && (4.9)\\
&= \mathbb{P}(\mathrm{wt}(\boldsymbol{v}) = \lambda \mid F\boldsymbol{v} = 0) \cdot \frac{\mathbb{P}(F\boldsymbol{v} = 0)}{\mathbb{P}(\mathrm{wt}(\boldsymbol{v}) = \lambda)} \\
&\leq \frac{\mathbb{P}(F\boldsymbol{v} = 0)}{\mathbb{P}(\mathrm{wt}(\boldsymbol{v}) = \lambda)} . && (4.10)
\end{aligned}$$

[8] Recall that we think of q as a constant.

We proceed to bound the right-hand side of (4.9). For the denominator, note that

$$\mathbb{P}\left(\mathrm{wt}(\boldsymbol{v}) = \lambda\right) = \binom{n}{\lambda n}\beta^{\lambda n}(1-\beta)^{(1-\lambda)n} \geq q^{-D_{\mathrm{KL}_q}(\lambda\|\beta)n} \tag{4.11}$$

where above $D_{\mathrm{KL}_q}(x \parallel y)$ denotes the q-ary KL Divergence,

$$D_{\mathrm{KL}_q}(x \parallel y) = -x\log_q\frac{y}{x} - (1-x)\log_q\frac{1-y}{1-x} \text{ for } x \in [0,1] \text{ and } y \in (0,1).$$

We next focus on the numerator. The following notation will be useful:

Definition 4.5.5. For $k \in \mathbb{N}$, let

$$\mathbb{V}_q^k = \left\{ w \in \mathbb{F}_q^k : \sum_{i=1}^{k} w_i = 0 \right\}.$$

Let $f_1, \ldots, f_{\frac{n}{s}}$ denote the rows of the matrix F. Note that the vectors $f_1, \ldots f_{\frac{n}{s}}$ have disjoint supports, so the inner products $\langle f_i, \boldsymbol{v}\rangle$ are independently and identically distributed. Hence, $\mathbb{P}\left(F\boldsymbol{v} = 0\right) = \mathbb{P}\left(\langle f_1, \boldsymbol{v}\rangle = 0\right)^{\frac{n}{s}}$. Observe that the distribution of $\boldsymbol{v}$ is symmetric to multiplication of each entry by a nonzero element of $\mathbb{F}_q$. Consequently, if $\boldsymbol{w} \sim \mu_q^s$,

$$\mathbb{P}\left(F\boldsymbol{v} = 0\right) = \mathbb{P}\left(\langle f_1, \boldsymbol{v}\rangle = 0\right)^{\frac{n}{s}} = \mathbb{P}\left(\sum_{i=1}^{s} \boldsymbol{v}_i = 0\right)^{\frac{n}{s}} = \mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}_q^s\right)^{\frac{n}{s}}. \tag{4.12}$$

The following lemma gives a closed form for this last expression.

Lemma 4.5.6.

$$\mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}_q^s\right) = \frac{1 + (q-1)\left(1 - \frac{q\beta}{q-1}\right)^s}{q}.$$

Proof. We proceed by induction. The base case ($s = 0$) is immediate. Now suppose that the statement holds for $s-1$ and let $\pi : \mathbb{F}_q^s \to \mathbb{F}_q^{s-1}$ denote the projection onto the first $s-1$ coordinates. Then

$$\begin{aligned}
\mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}_q^s\right) &= \mathbb{P}\left(\pi(\boldsymbol{w}) \in \mathbb{V}_q^{s-1}\right) \cdot \mathbb{P}\left(\boldsymbol{w}_s = 0\right) \\
&\quad + \mathbb{P}\left(\pi(\boldsymbol{w}) \notin \mathbb{V}_q^{s-1}\right) \cdot \mathbb{P}\left(\boldsymbol{w}_s = -\sum_{i=1}^{s-1} \boldsymbol{w}_i \mid \pi(\boldsymbol{w}) \notin \mathbb{V}_q^{s-1}\right) \\
&= \frac{1 + (q-1)\left(1 - \frac{q\beta}{q-1}\right)^{s-1}}{q} \cdot (1-\beta) \\
&\quad + \left(1 - \frac{1 + (q-1)\left(1 - \frac{q\beta}{q-1}\right)^{s-1}}{q}\right) \cdot \frac{\beta}{q-1} \\
&= \frac{1}{q} + \left(1 - \frac{q\beta}{q-1}\right)^s \left(\frac{q-1}{q}\right),
\end{aligned}$$

which establishes the inductive hypothesis for s. $\square$

Motivated by the computations above, we can define the following useful shorthands:

Definition 4.5.7. For $\lambda, \beta \in (0, \frac{q-1}{q}]$, define

$$Z(\beta) = \mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}_q^s\right) = \frac{1 + (q-1)\left(1 - \frac{q\beta}{q-1}\right)^s}{q}\,, \tag{4.13}$$
$$\psi(\lambda, \beta) = sD_{\mathrm{KL}q}\left(\lambda \parallel \beta\right) + \log_q Z(\beta)\,.$$

From Equations (4.8), (4.9), (4.11) and (4.12), we conclude that

$$\begin{aligned}\log_q P_\lambda = t \log_q \mathbb{P}\left(F\boldsymbol{u} = 0\right) &\leq tn\left(D_{\mathrm{KL}q}\left(\lambda \parallel \beta\right) + \frac{\log_q\left(1 + (q-1)\left(1 - \frac{q\beta}{q-1}\right)^s\right) - 1}{s}\right) \\ &= (1-R)n\left(sD_{\mathrm{KL}q}\left(\lambda \parallel \beta\right) + \log_q\left(1 + (q-1)\left(1 - \frac{q\beta}{q-1}\right)^s\right) - 1\right) \\ &= (1-R)n\psi(\lambda, \beta) \end{aligned} \tag{4.14}$$

for every $\beta \in \left(0, \frac{q-1}{q}\right]$. Above, we have used the choice $t = (1-R)s$.

This motivates the following definition:

Definition 4.5.8. Let Z and ψ be as in Definition 4.5.7. Define:

$$\varphi(\lambda) = \inf_{\beta \in (0, \frac{q-1}{q}]} \psi(\lambda, \beta)\,.$$

Definition 4.5.8, along with (4.14), implies that $\log_q P_\lambda \leq \varphi(\lambda)$, which establishes Item 2 of Lemma 4.5.2. Next we establish Item 1 of Lemma 4.5.2. This follows from Definition 4.5.8, since

$$\varphi(\lambda) \leq \psi(\lambda, \lambda) = \log_q\left(1 + (q-1)\left(1 - \frac{q\lambda}{q-1}\right)^s\right) - 1\,,$$

using the fact that $D_{\mathrm{KL}q}\left(\lambda \parallel \lambda\right) = 0$.

This almost completes the proof of Lemma 4.5.2, except for Item 3, which we establish in the next section using calculus.

4.5.3 Proof of Item 3 of Lemma 4.5.2

In this section we prove Item 3, which will establish Lemma 4.5.2 and hence Theorem 4.3.5.

Remark 4.5.9 (Difference between [Gal62] and this proof)**.** This is the part of the proof where our techniques diverge from Gallager's. The part of [Gal62] which corresponds to our Item 3 consists of an intricate analytic argument which does not seem (to us) to generalize to larger alphabets. Thus, our proof has to rely on a different, more general, argument, which we now provide.

Before proving Item 3 of Lemma 4.5.2, we need to better understand the relation between a given $\lambda \in (0, \frac{q-1}{q}]$, and the β which minimizes the expression $\psi(\lambda, \beta)$.

Lemma 4.5.10. *Let $\lambda \in (0, \frac{q-1}{q}]$. Then, $\psi(\lambda, \beta)$ is minimized by a unique $\beta \in (0, \frac{q-1}{q}]$. This β is the only solution for*

$$\mathop{\mathbb{E}}_{\boldsymbol{w} \sim \mu_q(\beta)} \left[\mathrm{wt}(\boldsymbol{w}) \mid \boldsymbol{w} \in \mathbb{V}_q^s \right] = \lambda \, .$$

Proof. We compute the derivative. Recall that $\ln$ denotes logarithm to the base e.

$$\begin{aligned}
\frac{d \ln Z(\beta)}{d\beta} &= \frac{1}{\mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}_q^s\right)} \cdot \frac{d\left(\mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}_q^s\right)\right)}{d\beta} \\
&= \frac{1}{\mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}_q^s\right)} \cdot \sum_{w \in \mathbb{V}_q^s} \frac{d\left(\frac{\beta}{q-1}\right)^{s \cdot \mathrm{wt}(w)} (1-\beta)^{s \cdot (1 - \mathrm{wt}(w))}}{d\beta} \\
&= \frac{\sum_{w \in \mathbb{V}_q^s} \left(\left(\frac{\beta}{q-1}\right)^{s \cdot \mathrm{wt}\, w} (1-\beta)^{s \cdot (1 - \mathrm{wt}\, w)} \cdot s \cdot \left(\frac{\mathrm{wt}(w)}{\beta} - \frac{1 - \mathrm{wt}(w)}{1-\beta} \right) \right)}{\mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}_q^s\right)} \\
&= s \cdot \left(\frac{\mathbb{E}\left[\mathrm{wt}(\boldsymbol{w}) \mid \boldsymbol{w} \in \mathbb{V}_q^s\right]}{\beta} - \frac{1 - \mathbb{E}\left[\mathrm{wt}(\boldsymbol{w}) \mid \boldsymbol{w} \in \mathbb{V}_q^s\right]}{1-\beta} \right) \, . \qquad (4.15)
\end{aligned}$$

Also, it is not hard to see that

$$\frac{\partial D_{\mathrm{KL}q}\left(\lambda \parallel \beta\right)}{\partial \beta} = \frac{1}{\ln(q)} \cdot \left(\frac{1-\lambda}{1-\beta} - \frac{\lambda}{\beta} \right) \, .$$

Consequently,

$$\begin{aligned}
\frac{\partial \psi(\lambda, \beta)}{\partial \beta} &= s \frac{\partial D_{\mathrm{KL}q}\left(\lambda \parallel \beta\right)}{\partial \beta} + \frac{d \log_q Z(\beta)}{d\beta} \\
&= \log_q e \cdot \left(\frac{s(1-\lambda)}{1-\beta} - \frac{s\lambda}{\beta} + \frac{d \log_e Z(\beta)}{d\beta} \right) \\
&= s \cdot \log_q e \cdot \left(\mathbb{E}\left[\mathrm{wt}(\boldsymbol{w}) \mid \boldsymbol{w} \in \mathbb{V}_q^s\right] - \lambda \right) \left(\frac{1}{1-\beta} + \frac{1}{\beta} \right) \, .
\end{aligned}$$

We conclude that $\frac{\partial \psi(\lambda, \beta)}{\partial \beta}$ has the same sign as $\mathbb{E}\left[\mathrm{wt}(\boldsymbol{w}) \mid \boldsymbol{w} \in \mathbb{V}_q^s\right] - \lambda s$. The lemma now follows from the following claim:

Claim 4.5.11. *As β increases in the range $(0, \frac{q-1}{q}]$ the function $\mathbb{E}\left[\mathrm{wt}(\boldsymbol{w}) \mid \boldsymbol{w} \in \mathbb{V}_q^s\right]$ strictly increases from 0 to $\frac{q-1}{q}$.*

Proof of Claim 4.5.11. Due to (4.13) and (4.15),

$$\begin{aligned}
\mathbb{E}\left[\mathrm{wt}(\boldsymbol{w}) \mid w \in \mathbb{V}_q^s\right] &= \left(\frac{d \ln Z(\beta)}{s \cdot d\beta} + \frac{1}{1-\beta}\right)\beta(1-\beta) \\
&= \left(\frac{\frac{dZ(\beta)}{d\beta}}{s \cdot Z(\beta)} + \frac{1}{1-\beta}\right)\beta(1-\beta) \\
&= \left(\frac{-q\left(1-\frac{q\beta}{q-1}\right)^{s-1}}{1+(q-1)\left(1-\frac{q\beta}{q-1}\right)^{s}} + \frac{1}{1-\beta}\right)\beta(1-\beta) \\
&= \beta \cdot \frac{1-\left(1-\frac{q\beta}{q-1}\right)^{s-1} \cdot (1+q\beta)}{1+(q-1)\left(1-\frac{q\beta}{q-1}\right)^{s}}\,, \qquad (4.16)
\end{aligned}$$

(4.17)

and the claim readily follows. □

The proof of Lemma 4.5.10 is thus concluded. □

Lemma 4.5.10 and Claim 4.5.11 justify the following definition:

Definition 4.5.12. For $\lambda \in (0, \frac{q-1}{q}]$, denote by $\beta(\lambda)$ the unique $\beta \in (0, \frac{q-1}{q}]$ which minimizes $\psi(\lambda, \beta)$. The inverse of this function is denoted $\lambda(\beta)$.

By Lemma 4.5.10 and Equation (4.16),

$$\lambda(\beta) = \beta \frac{1-\left(1-\frac{q\beta}{q-1}\right)^{s-1}}{1+(q-1)\left(1-\frac{q\beta}{q-1}\right)^{s}}\,. \qquad (4.18)$$

Remark 4.5.13. Unfortunately, there are good reasons to suspect that the function $\beta(\lambda)$ has no closed-form expression (see, e.g., the discussion about backward mapping in [WJ+08, Sec. 3.4.2]), so we prefer to work with its inverse.

It is convenient to extend the definition of these functions to the closed interval $\left[0, \frac{q-1}{q}\right]$ by taking limits, namely, $\lambda(0) = \beta(0) = 0$, and

$$\begin{aligned}
\varphi(0) &= \lim_{\lambda\to 0}\varphi(\lambda) = \lim_{\lambda\to 0}\psi(\lambda,\beta(\lambda)) \lim_{\beta\to 0}\psi(\lambda(\beta),\beta) = \lim_{\beta\to 0} D_{\mathrm{KL}q}\left(\lambda(\beta) \parallel \beta\right) + \log_q Z(\beta) \\
&= \lim_{\beta\to 0} D_{\mathrm{KL}q}\left(\lambda(\beta) \parallel \beta\right) = \lim_{\beta\to 0} -\lambda(\beta)\log_q \beta = 0\,.
\end{aligned}$$

We are now able to prove Item 3 of Lemma 4.5.2. Namely, we show that $\frac{\varphi(\lambda)}{h_q(\lambda)}$ is strictly increasing in the range $0 < \lambda \le \frac{q-1}{q}$.

Proof of Lemma 4.5.2, Item 3. Let $\alpha(\lambda) = \frac{\varphi(\lambda)}{h_q(\lambda)}$. The desired result follows immediately from the four following claims:

Claim 4.5.14. $\alpha(\frac{q-1}{q}) = -1$.

Claim 4.5.15. $\alpha(\lambda) < -1$ *for some* $\lambda \in (0, \frac{q-1}{q})$.

Claim 4.5.16. *There exists* $\varepsilon > 0$ *such that* $\alpha(\lambda) > -\frac{s}{2}$ *for all* $\lambda \in (0, \varepsilon)$.

Claim 4.5.17. *For each* $y \in (-\frac{s}{2}, -1]$*, the equation* $\alpha(\lambda) = y$ *has at most one solution* $\lambda \in (0, \frac{q-1}{q}]$.

Indeed, Claims 4.5.14 and 4.5.17 show that $\alpha(\lambda) \neq -1$ for $\lambda < \frac{q-1}{q}$. Since α is continuous, it is either upper bounded or lower bounded by -1 in the whole range $(0, \frac{q-1}{q}]$. Claim 4.5.15 implies the former is the case. By Claim 4.5.17, if $-\frac{s}{2} < \alpha(\lambda_0) < -1$ for some $\lambda_0 \in (0, \frac{q-1}{q})$, then α must be strictly increasing in the range $[\lambda_0, \frac{q-1}{q}]$. The lemma now follows from Claim 4.5.16. We proceed to prove these claims.

Proof of Claim 4.5.14. Note that $\alpha(\frac{q-1}{q}) = \varphi(\frac{q-1}{q})$. Due to Item 2,

$$\varphi\left(\frac{q-1}{q}\right) \leq -1 \,.$$

In the reverse direction,

$$\begin{aligned}\varphi(\lambda) = \min_{\beta} \psi(\lambda, \beta) &= \min_{\beta} \left(s \cdot D_{\mathrm{KL}q}\left(\lambda \parallel \beta\right) + \log_q Z(\beta)\right) \\ &\geq \min_{\beta} \left(s \cdot D_{\mathrm{KL}q}\left(\lambda \parallel \beta\right)\right) - 1 \geq -1\end{aligned}$$

for all λ. The first inequality above holds since $Z(\beta) \geq \frac{1}{q}$, due to (4.13). □

Proof of Claim 4.5.15. By Item 1,

$$\alpha(\lambda) \leq \frac{\log_q\left(1 + (q-1)\left(1 - \frac{q}{q-1}\lambda\right)^s\right) - 1}{h_q(\lambda)} \,. \tag{4.19}$$

Let $\lambda = \frac{q-1}{q} - \varepsilon$. As ε tends from above to 0, the numerator of (4.19)'s right-hand side is $-1 + \Theta(\varepsilon^s)$, while the denominator is $1 - \Theta(\varepsilon^2)$. Thus, for ε small enough, (4.19) yields $\alpha(\lambda) < -1$. □

Proof of Claim 4.5.16. Let

$$\bar{Z}(\beta) = \mathbb{P}\left(\boldsymbol{w} \in \mathbb{V}^s \wedge \mathrm{wt}(\boldsymbol{w}) \leq \frac{2}{s}\right) = (1-\beta)^s + \binom{s}{2}(1-\beta)^{s-2}\beta^2$$

and

$$\bar{\psi}(\beta,\lambda) = sD_{\mathrm{KL}_q}(\lambda \parallel \beta) + \log_q \bar{Z}(\beta) .$$

Clearly, $\bar{\psi}(\beta,\lambda)$ is a lower bound on $\psi(\beta,\lambda)$, so

$$\varphi(\lambda) \geq \min_{\beta\in(0,\frac{q-1}{q}]} \bar{\psi}(\lambda,\beta) .$$

Note that

$$\frac{\partial\bar{\psi}(\lambda,\beta)}{\partial\beta} = \frac{s}{\beta(1-\beta)}\left(\frac{2(s-1)}{\left(\frac{1-\beta}{\beta}\right)^2 + \binom{s}{2}} - \lambda\right) ,$$

Hence, for $\lambda < \frac{2}{s}$, the minimum of $\bar{\psi}(\lambda,\beta)$ is attained at $\beta_0 = \frac{y}{1+y}$, where

$$y = \left(\frac{\lambda}{2(s-1) - \binom{s}{2}\lambda}\right)^{\frac{1}{2}} .$$

Therefore,

$$\begin{aligned}\alpha(\lambda) = \frac{\varphi(\lambda)}{h_q(\lambda)} &\geq \frac{\bar{\psi}(\lambda,\beta_0)}{h_q(\lambda)} \\ &= \frac{s}{2}\left(-1 + \frac{\lambda\left(\log_q\left(2(s-1) - \binom{s}{2}\lambda\right) - \log_q(1-\lambda s)\right) + (1-\lambda)\log_q(1-\lambda)}{h_q(\lambda)}\right) .\end{aligned}$$

For λ small enough, the right-hand side is clearly larger than $-\frac{s}{2}$. □

Proof of Claim 4.5.17. Denote $\beta^* = \beta(\lambda)$. Let $y \in (-\frac{s}{2}, -1]$, and define the function $\varphi_y(\lambda) = \varphi(\lambda) - yh_q(\lambda)$. We seek to show that $\varphi_y(\lambda)$ has at most one root in the range $(0, \frac{q-1}{q}]$. This is a consequence of the following three statements, proven below:

1. $\frac{d\varphi_y(\lambda)}{d\lambda}$ has at most one extremal point in the open interval $(0, \frac{q-1}{q})$.
2. $\frac{d\varphi_y(\lambda)}{d\lambda}(\frac{q-1}{q}) = 0$.
3. $\varphi_y(0) = 0$.

Indeed, Item 1 implies that $\frac{d\varphi_y(\lambda)}{d\lambda}$ has at most two roots in the interval $(0, \frac{q-1}{q}]$. Item 2 says that one of these roots is at $\frac{q-1}{q}$, so $\frac{d\varphi_y(\lambda)}{d\lambda}$ has at most one root in $(0, \frac{q-1}{q})$. Consequently $\varphi_y(\lambda)$ has at most one extremal point and two roots in $[0, \frac{q-1}{q}]$. Due to Item 3, one of these roots is 0, so there can only be one root in $(0, \frac{q-1}{q}]$. We turn to prove these statements.

Item 3 is trivial. For Item 2, note that in the derivative

$$\frac{d\varphi(\lambda)}{d\lambda} = \frac{\partial\psi(\lambda,\beta)}{\partial\beta}_{\beta=\beta^*} \cdot \frac{d\beta^*}{d\lambda} + \frac{\partial\psi(\lambda,\beta)}{\partial\lambda}_{\beta=\beta^*} ,$$

the first term vanishes since ψ has a minimum at (λ, β^*). Hence,

$$\frac{d\varphi(\lambda)}{d\lambda} = \frac{\partial\psi(\lambda,\beta)}{\partial\lambda}_{\beta=\beta^*} = s\frac{\partial D_{\mathrm{KL}q}(\lambda \parallel \beta)}{\partial\lambda}_{\beta=\beta^*} = s\log_q \frac{\lambda(1-\beta^*)}{(1-\lambda)\beta^*}.$$

In particular, $\beta(\frac{q-1}{q}) = \frac{q-1}{q}$, so

$$\frac{d\varphi_y(\lambda)}{d\lambda}_{\lambda=\frac{q-1}{q}} = \frac{d\varphi(\lambda)}{d\lambda}_{\lambda=\frac{q-1}{q}} - y\frac{dh_q(\lambda)}{d\lambda}_{\lambda=\frac{q-1}{q}} = 0\,,$$

since, in the last equality, the two terms vanish.

We turn to Item 1. Define the new variable $x = 1 - \frac{q\beta^*}{q-1}$. Note the following useful relations, the second of which follows from Equation (4.18):

$$\beta^* = \frac{q-1}{q}(1-x) \tag{4.20}$$

and

$$\frac{\lambda}{1-\lambda} = \frac{\beta^*}{1-\beta^*}\cdot\frac{1-x^{s-1}}{1+(q-1)x^{s-1}}\,. \tag{4.21}$$

By (4.20) and (4.21),

$$\begin{aligned}
\frac{d\varphi_y(\lambda)}{d\lambda} &= s\frac{\partial D_{\mathrm{KL}q}(\lambda \parallel \beta)}{\partial\lambda}_{|\beta=\beta^*} - y\frac{dh_q(\lambda)}{d\lambda} \\
&= s\log_q \frac{\lambda(1-\beta^*)}{(1-\lambda)\beta^*} + y\log_q \frac{\lambda}{1-\lambda} \\
&= s\log_q \frac{1-\beta^*}{\beta^*} + (s+y)\log_q \frac{\lambda}{1-\lambda} \\
&= -y\log_q \frac{1+(q-1)x}{(q-1)(1-x)} + (s+y)\log_q \frac{1-x^{s-1}}{1+(q-1)x^{s-1}}\,.
\end{aligned}$$

Now,

$$\frac{d^2\varphi_y(\lambda)}{dxd\lambda}\cdot\ln q = \frac{-yq}{(1+(q-1)x)(1-x)} - \frac{(s+y)(s-1)qx^{s-2}}{(1-x^{s-1})\,(1+(q-1)x^{s-1})}\,.$$

This second derivative vanishes when

$$\frac{-(s+y)}{y} = \frac{(1-x^{s-1})\,(1+(q-1)x^{s-1})}{(s-1)(1+(q-1)x)(1-x)x^{s-2}}\,.$$

Equivalently,

$$\frac{-(s+y)}{y} = \frac{1}{s-1}\sum_{i=0}^{s-2}\frac{x^{-i}+(q-1)x^{i+1}}{1+(q-1)x}. \tag{4.22}$$

By examining each term of this sum separately, it is straightforward to verify that the right-hand side of (4.22) is a convex function of x, which tends to ∞ (resp. 1) as $x \to 0$ (resp. $x \to 1$). Since $y > -\frac{s}{2}$, the left-hand side of (4.22) is larger than 1, so there is a unique $x \in (0,1)$ which solves (4.22). Item 1 follows. □

This establishes Item 3 of Lemma 4.5.2. □

4.6 Open Problems

In this work, we answered Question 4.1.1 with an emphatic "yes". There are LDPC codes that achieve list-decoding capacity, and moreover there are many of them, and moreover these codes are also likely to satisfy any local property which is likely to be satisfied by a random linear code. However, we feel that our results are just the tip of the iceberg. They raise several interesting questions:

1. **Derandomization?** Our results hold for a random ensemble of LDPC codes. It is natural to ask whether (or to what extent) this construction can be derandomized. In particular, it does not seem as though the underlying graph being an expander would be sufficient.
2. **Algorithms?** Our results are combinatorial, but one of our main motivations is algorithmic. At the moment we do not know of any truly linear-time list-decoding algorithms for any capacity-achieving list-decodable codes. Since essentially all known linear-time algorithms in coding theory arise from graph-based codes, such codes are a natural candidate for linear-time list-decoding. Now that we know that random LDPC codes achieve list-decoding capacity combinatorially, can we list-decode them efficiently? As a natural starting point, Hemenway and Wootters [HW15] have shown how to list-recover Tanner codes *from erasures* in linear-time. Furthemore, recent work by Ron-Zewi, Wootters and Zémor [RZWZ20] shows how to list-decode binary Tanner codes from erasures. Perhaps a modification of their techniques, together with our combinatorial proof of the LDPC codes' list-decodability, can lead to an analyzable list-decoding algorithm.

Chapter 5

On the List-Decodability of Random Linear Codes over the Rank Metric

As alluded to in Chapter 1, while coding theorists have typically used the Hamming metric to define the distance between words, there are other metrics that one could consider. Motivated by applications in network coding [KK08; SKK08], space time coding [LGB03; LK05], magnetic recording [Rot91], and cryptography [GPT91; Loi10; Loi17], researchers have turned their attention to the *rank metric*. Introduced by Delsarte [Del78], in a rank metric code, codewords are matrices over a finite field and the distance between codewords is the rank of their difference.

In this chapter, we will be concerned with the list-decodability of rank metric codes. Specifically, for $n \leq m \in \mathbb{N}$ an $\mathbb{F}_q$-linear rank-metric code over $\mathbb{F}_q^{m\times n}$ of rate $R = (1-\rho)(1-b\rho) - \varepsilon$ (where $b := \frac{n}{m}$) is shown to be (with high probability) list-decodable up to fractional radius $\rho \in (0,1)$ with lists of size at most $\frac{C_{\rho,q,b}}{\varepsilon}$, where $C_{\rho,q,b}$ is a constant depending only on ρ, q and b. This matches the bound for random rank metric codes (up to constant factors). The proof adapts the approach of Guruswami, Håstad and Kopparty [GHK11], who established a similar result for the Hamming metric case, to the rank metric setting.

5.1 Primer on Rank Metric Codes

For any matrices $X, Y \in \mathbb{F}_q^{m\times n}$ with $m \leq n$, the *(normalized) rank distance* between X and Y is

$$d_R(X,Y) = \frac{1}{n}\operatorname{rank}(X-Y)\ .$$

Observe that this indeed defines a metric (the triangle inequality is a consequence of the sub-additivity of rank). A *rank metric code* is then a subset $\mathcal{C} \subseteq \mathbb{F}_q^{m\times n}$, and if $\mathcal{C}$ happens to be a subspace then it is called a linear code (observe that $\mathbb{F}_q^{m\times n}$ naturally has the structure of an mn-dimensional vector space over $\mathbb{F}_q$).

The notions of *rate* and *distance* naturally apply to rank metric codes: the rate is $R(\mathcal{C}) = \frac{\log_q |\mathcal{C}|}{nm}$ and its distance is $\delta(\mathcal{C}) = \min\{d_R(X,Y) : X, Y \in \mathcal{C}, X \neq Y\}$. If $\mathcal{C}$ happens to be linear, we can simplify these expressions to $R(\mathcal{C}) = \frac{\dim(\mathcal{C})}{mn}$ and $\delta(\mathcal{C}) = \min\{\mathrm{rank}(X)/n : X \in \mathcal{C} \setminus \{0\}\}$.

Recall the Singleton bound (Theorem 2.4.1). An analogous result holds for the rank metric:

Theorem 5.1.1 (Rank Metric Singleton Bound [Gab85])**.** *If $\mathcal{C} \subseteq \mathbb{F}_q^{m \times n}$ is a rank metric code with minimum distance δ, then*

$$R(\mathcal{C}) \leq 1 - \delta + \frac{1}{n} .$$

Just as the Singleton bound is achievable by an explicit family of codes over the Hamming metric (namely, Reed-Solomon codes; cf. Example 2.5.1), there is an explicit family of rank-metric codes achieving the tradeoff in Theorem 5.1.1 which are called Gabidulin codes. We defer a formal definition of Gabidulin codes to Chapter 8 (specifically, Example 8.1.6 in Section 8.1.2); for now, suffice it to say that they are the analog of RS codes for the rank metric.

5.1.1 List-Decodable Rank Metric Codes

Next, we discuss the list-decodability of rank metric codes. First, as in any metric space, we have the concept of a metric ball:

Definition 5.1.2 (Rank Metric Ball)**.** For $\rho \in (0,1)$ and $Z \in \mathbb{F}_q^{m \times n}$, the *rank metric ball* of radius ρ centered at Z is

$$B_R(Z, \rho) = \{X \in \mathbb{F}_q^{m \times n} : d_R(X, Z) \leq \rho\} .$$

We can then define what it means for a rank metric code to be list-decodable. In this chapter, we are only concerned with the combinatorial property of list-decodability (and not the algorithmic task of computing the list from a received word).

Definition 5.1.3 (List-Decodable Rank Metric Code)**.** Let $\rho \in (0,1)$ and $L \geq 1$. A rank metric code $\mathcal{C} \subseteq \mathbb{F}_q^{m \times n}$ is (ρ, L)*-list-decodable* if for all $Z \in \mathbb{F}_q^{m \times n}$,

$$|B_R(Z, \rho) \cap \mathcal{C}| \leq L .$$

Recall that the size of a Hamming ball was captured quite well by the q-ary entropy function $h_q(\rho)$ from Definition 2.4.2: Proposition 2.4.5 provides the estimate $|B(z, \rho)| \approx q^{nh_q(\rho)}$. We would like to obtain a similar estimate for rank metric balls. Note that

$$|B_R(Z, \rho)| = \sum_{r=0}^{\lfloor \rho n \rfloor} N_q(r, m, n) ,$$

where

$$N_q(r, m, n) = \prod_{j=0}^{r-1} \frac{(q^n - q^j)(q^m - q^j)}{q^r - q^j}$$

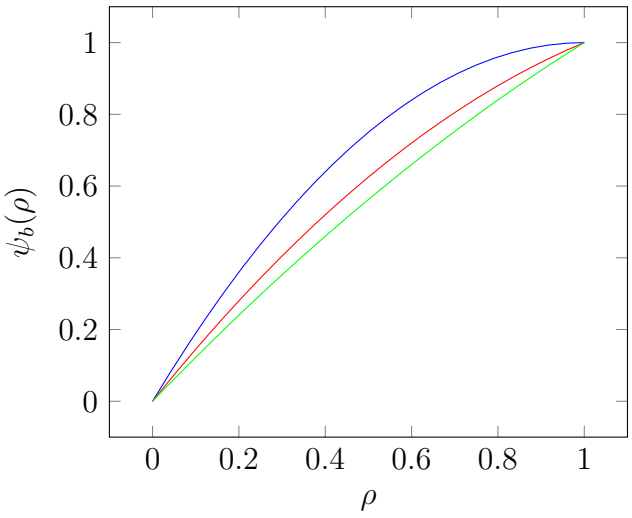

Figure 5.1: Graph of $\psi_b(\rho)$ for various values of balancedness b. In blue, $b = 1$; in red, $b = 0.5$; in green, $b = 0.25$.

counts the number of rank r matrices in $\mathbb{F}_q^{m\times n}$.

The following proposition from [GY08] provides a useful estimate of $|B_R(Z,\rho)|$.

Proposition 5.1.4 (Rank-Metric Ball Estimate)**.** *Define* $\psi_b : [0,1] \to [0,1]$ *by*

$$\psi_b(\rho) = \rho + \rho b - \rho^2 b\,. \tag{5.1}$$

Then

$$q^{mn\psi_b(\rho)} \le |B_R(Z,\rho)| \le K_q^{-1} q^{mn\psi_b(\rho)}\,, \tag{5.2}$$

where $K_q = \prod_{j=1}^{\infty}(1-q^{-j})$.

Remark 5.1.5. Observe that the quantity $K_q \in (0,1)$ and increases with q. Moreover, $K_2 \approx 0.2887$, so being a bit lax we have $K_q^{-1} < 4$. Thus, Proposition 5.1.4 actually shows $|B_R(Z,\rho)| = \Theta\left(q^{mn\psi_b(\rho)}\right)$, which is a tighter estimate than that guaranteed by Proposition 2.4.5.

Remark 5.1.6. For Hamming balls, the estimate was only effective when $\rho < 1 - 1/q$; however, note that $\psi_b(\rho) < 1$ whenever $\rho < 1$. See Figure 5.1.

Remark 5.1.7. Interestingly, the function $\psi_b(\rho)$ does not depend on the underlying field $\mathbb{F}_q$, but only on the ratio $b = \frac{n}{m}$. Moreover, recall that as $q \to \infty$, $h_q(\rho)$ approaches ρ. Similarly, as $b \to 0$, $\psi_b(\rho)$ approaches ρ. For this reason, we think of the "large q regime" in the Hamming metric as being morally similar to "small b regime" in the rank metric.

From this, we can deduce a list-decoding capacity theorem for rank metric codes. As in the proof of the list-decoding capacity theorem for the Hamming metric (Theorem 2.4.7), the first bullet-point is proved by considering the performance of a uniformly random rank metric code of the prescribed rate.

Theorem 5.1.8 (List-Decoding Capacity Theorem for Rank Metric Codes [Din14]). *Let $n \leq m \in \mathbb{N}$ and put $b = \frac{n}{m}$. Fix $\rho \in (0,1)$ and $\varepsilon > 0$.*

- *There exists a code $\mathcal{C} \subseteq \mathbb{F}_q^{m \times n}$ of rate $1 - \psi_b(\rho) - \varepsilon$ which is $(\rho, O(1/\varepsilon))$-list-decodable.*
- *For any code $\mathcal{C} \subseteq \mathbb{F}_q^{m \times n}$ of rate $1 - \psi_b(\rho) + \varepsilon$, there exists a $Z \in \mathbb{F}_q^{m \times n}$ for which $|B(Z, \rho) \cap \mathcal{C}| \geq q^{\varepsilon nm}$.*

In this chapter, we wish to determine whether we can match the performance of a uniformly random rank metric code with a random *linear* rank metric code. Our main result (see Section 5.3) indicates that this is the case for a wide regime of parameters.

Before stating our results, we survey prior work concerning the list-decodability of rank metric codes.

5.2 Prior Work

List decoding Gabidulin codes. Gabidulin codes are the rank metric code that have been the most extensively studied and we briefly collect certain facts concerning their list-decoding. An algorithm for Gabidulin codes akin to the Welch-Berlekamp algorithm has been provided by Loidreau [Loi06]; however, it only guarantees unique decoding up to half the distance of the code. Kuijper and Trautmann [KT14] have provided a list-decoding algorithm for Gabidulin codes, but it is not guaranteed to run in polynomial time, nor is it guaranteed to output a list of polynomial size.

On the negative side, Wachter-Zeh [WZ12] has shown that Gabidulin codes of rate R cannot be list-decoded beyond the Johnson radius $1 - \sqrt{R}$. (Recall that it remains an open problem to determine if Reed-Solomon codes can be list-decoded beyond the Johnson bound. However, by the Guruswami-Sudan algorithm [GS99], it is known how to decode up to the Johnson bound.) More recently, Raviv and Wachter-Zeh [RWZ16] (see also the correction in [RWZ17]) have shown that certain Gabidulin codes cannot be list-decoded beyond half the minimum distance.

Nonetheless, certain variants of Gabidulin codes can be list-decoded beyond half the minimum distance. Guruswami, Wang and Xing [GK16] (see also [GX13; GW14]) provided an explicit construction of a subcode of the Gabidulin code of rate R that can be list-decoded up to radius $1 - R - \varepsilon$, matching the Singleton bound for rank metric codes (Theorem 5.1.1). Their construction and analysis actually inspire the techniques we use in Chapter 8, so we expound upon them then; see Section 8.1. Mahdifar and Vardy [MV12] have also provided an algorithm for list-decoding a "folded" variant of Gabidulin codes up to the Singleton bound, although the list size is of exponential size.

List-decoding random rank metric codes. The study of the list-decodability of random rank metric codes was initiated by Ding [Din14]. First she considers the performance of a uniformly random rank metric code, and essentially proves Theorem 5.1.8. Furthermore, she studies when it is possible to list decode up to the Singleton bound

$1-R-\varepsilon$, and demonstrates that $b \lesssim \varepsilon$ is necessary and sufficient. (This can be compared to Proposition 2.4.6, reinforcing the motto that "small b" is akin to "large q".)

Furthermore, Ding shows that random linear rank metric codes of rate $1-\psi_b(\rho)-\varepsilon$ are $(\rho, \exp(O(1/\varepsilon)))$-list-decodable with high probability. Her argument can be viewed as a natural adaptation of the Zyablov-Pinsker argument (see Section 3.1), explaining the exponential dependence on the gap to capacity.

5.3 Our Results

Our main result shows that random linear rank metric codes have list sizes that grow linearly with the reciprocal of the gap to capacity.

Theorem 5.3.1 (Main Theorem)**.** *Let $\rho \in (0,1)$ and $n \leq m$. There exists a constant $C = C_{\rho,q,b} > 0$ such that a random $\mathbb{F}_q$-linear rank metric code $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^{m\times n}$ of rate $R = 1-\psi_b(\rho)-\varepsilon$ is $(\rho, C/\varepsilon)$-list-decodable with high probability.*

In the above theorem, the constant C blows up if $q \to \infty$, $\rho \to 1$ or $b \to 0$. This is reminiscent of the drawbacks that we discussed regarding the Guruswami-Håstad-Kopparty argument [GHK11] (recall that, informally, small b corresponds to large alphabet). This is no coincidence: our argument is an adaptation of their approach to the rank metric world. We now provide an informal overview of our proof strategy.

5.4 Overview of Approach

As indicated in Section 5.1, uniformly random rank metric codes $\boldsymbol{\mathcal{C}}$ of rate $1-\psi_b(\rho)-\varepsilon$ are with high probability $(\rho, O(1/\varepsilon))$-list-decodable. The proof follows from the fact that, for any center Z and a list $\{X_1, \ldots, X_L\} \subseteq B(Z,\rho)$, the events "$X_i \in \boldsymbol{\mathcal{C}}$" are independent. Hence, the probability that $\{X_1, \ldots, X_L\} \subseteq \boldsymbol{\mathcal{C}}$ is small enough to allow us to take a union bound over all possible lists. Unfortunately, for a random linear rank metric code $\boldsymbol{\mathcal{C}}$, the events "$X_i \in \boldsymbol{\mathcal{C}}$" are *not* independent; indeed, the events are not even 3-wise independent (as if X_i and X_j are in the code, then so is $X_i + X_j$). Since a list $\{X_1, \ldots, X_L\}$ is guaranteed to have a linearly independent subset of size $\lceil \log_q L \rceil$, one can use the argument for uniformly random codes to conclude that random linear rank metric codes are $(\rho, O(\exp(1/\varepsilon)))$-list-decodable – indeed, this is more-or-less the approach followed by Ding [Din14]. Thus, in order to prove that lists of size $O(1/\varepsilon)$ are sufficient, we will need to argue that, given a list contained in a small rank metric ball which does not contain a large linearly independent set, very few elements of their span will (with high probability) also lie in the rank metric ball.

Such an argument is given by Guruswami, Håstad and Kopparty [GHK11]. The technical core of their argument is to show that it is exponentially unlikely that ℓ vectors selected uniformly at random from the Hamming ball $B(0,\rho)$ have $\omega(\ell)$ elements of their linear span also lying in $B(0,\rho)$. That is, they show there exists a constant $C >$

0 (which depends on q and ρ) such that if $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell$ are sampled independently and uniformly at random from $B(0,\rho)$, the probability that $|\operatorname{span}\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell\} \cap B(0,\rho)| \geq C\ell$ is exponentially small in n. We prove an analogous result for matrices with the rank metric in Lemma 5.5.5.

In order to achieve this, the authors first show that, for any fixed vector $z \in \mathbb{F}_q^n$, if $\boldsymbol{x}_1, \boldsymbol{x}_2 \sim B(0,\rho)$ are sampled independently and uniformly, then it is exponentially unlikely that $\boldsymbol{x}_1 + \boldsymbol{x}_2 \in B(z,\rho)$. In order to bootstrap this to the case of selecting ℓ vectors from $B(0,\rho)$, the authors find a certain substructure in any set of $\mathbb{F}_q^\ell$ via a Ramsey-theoretic lemma which we introduce in Section 5.4.1.

We prove the appropriate generalization of this fact, concerning the sum of low-rank random matrices, in Lemma 5.5.4. This argument is a bit more involved than the analogous one in [GHK11] and represents the technical core of this chapter's contribution. Once we have proved this lemma, we are able to follow the framework of [GHK11] to conclude our main theorem (Theorem 5.3.1).

5.4.1 Increasing Sequences: A Ramsey-Theoretic Tool

As alluded to above, in order to analyze the probability that $C\ell$ elements from $\operatorname{span}\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_\ell\}$ lie in $B(0,\rho)$, we will look for combinatorial structure within any set of $C\ell$ linear combinations that we can exploit. Specifically, we will look for c-increasing sequences:

Definition 5.4.1 (c-Increasing Sequence)**.** Let $c \in \mathbb{N}$. A sequence of vectors $v_1, \ldots, v_d \in \mathbb{F}_q^\ell$ is a *c-increasing sequence* if for all $j \in [d]$,

$$\left| \operatorname{supp}(v_j) \setminus \bigcup_{i=1}^{j-1} \operatorname{supp}(v_i) \right| \geq c \,.$$

It is shown in [GHK11] that all sets in $\mathbb{F}_q^\ell$ have a translate containing a large c-increasing sequence. A crucial ingredient in their proof was a Ramsey-theoretic lemma proved by Sauer [Sau72] and Shelah [She72]. (More precisely, [GHK11] use a nonstandard q-ary version of the Sauer-Shelah lemma which they prove.) While we state the following result for general c, we remark that we will only ever require $c = 2$ in our proof.

Lemma 5.4.2 ([GHK11])**.** *For every prime power q, and all positive integers c, ℓ and $L \leq q^\ell$, the following holds. For every $S \subseteq \mathbb{F}_q^\ell$ with $|S| = L$, there is a $w \in \mathbb{F}_q^\ell$ such that $S + w$ has a c-increasing chain of length at least*

$$\frac{1}{c} \log_q \frac{L}{2} - \left(1 - \tfrac{1}{c}\right) \log_q((q-1)\ell) \;\; .$$

5.5 Proofs

The first statement we prove in this section gives an upper bound on the probability two random subspaces intersect significantly.

Claim 5.5.1. *Let $\boldsymbol{U}_1$ and $\boldsymbol{U}_2$ be independent and uniform subspaces of $\mathbb{F}_q^n$ of dimension $\rho_1 n$ and $\rho_2 n$, respectively. Assume $\rho_1 \leq \rho_2$. For any α satisfying $\max\{0, 1-\rho_1-\rho_2\} \leq \alpha \leq \rho_1$,*

$$\mathbb{P}\left(\dim(\boldsymbol{U}_1 + \boldsymbol{U}_2) \leq n(\rho_1+\rho_2-\alpha)\right) \leq 4^3 \exp_q\left(-n^2\alpha(1+\alpha-\rho_1-\rho_2)\right) .$$

The proof of Claim 5.5.1 makes use of the concept of the Grassmannian.

Definition 5.5.2 (Grassmannian)**.** For a vector space V over $\mathbb{F}_q$ and an integer $0 \leq k \leq \dim V$, denote by $G(k, V)$ the set of all subspaces $U \leq V$ of dimension k. If $n = \dim V$, we have

$$|G(k,V)| = \begin{bmatrix} n \\ k \end{bmatrix}_q = \prod_{j=0}^{k-1} \frac{q^n - q^j}{q^k - q^j} .$$

We record the following estimates for $\left[\begin{smallmatrix} n \\ k \end{smallmatrix}\right]_q$.

Lemma 5.5.3 ([GY08])**.** *We have*

$$K_q \cdot q^{k(n-k)} \leq \begin{bmatrix} n \\ k \end{bmatrix}_q \leq K_q^{-1} q^{k(n-k)} .$$

Proof of Claim 5.5.1. We will in fact bound the probability that $\dim(\boldsymbol{U}_1 \cap \boldsymbol{U}_2) \geq \alpha n$; since $\dim(\boldsymbol{U}_1 + \boldsymbol{U}_2) = \dim(\boldsymbol{U}_1) + \dim(\boldsymbol{U}_2) - \dim(\boldsymbol{U}_1 \cap \boldsymbol{U}_2) = \rho_1 n + \rho_2 n - \dim(\boldsymbol{U}_1 \cap \boldsymbol{U}_2)$, the claim will follow. Also, by increasing α if necessary it does no harm to assume αn is an integer. Finally, note that by conditioning on the realization of $\boldsymbol{U}_1$ it suffices to fix a dimension $\rho_1 n$ subspace U_1 and bound the probability that $\dim(U_1 \cap \boldsymbol{U}_2) \geq \alpha n$. To do this, we bound the probability that there exists a subspace $V \leq U_1$ of dimension αn for which $V \leq \boldsymbol{U}_2$. By the union bound,

$$\mathbb{P}\left(\exists V \in G(\alpha n, U_1) \text{ s.t. } V \leq \boldsymbol{U}_2\right) \leq \sum_{V \in G(\alpha n, U_1)} \mathbb{P}\left(V \leq \boldsymbol{U}_2\right) = \begin{bmatrix} \rho_1 n \\ \alpha n \end{bmatrix}_q \frac{\begin{bmatrix} n-\alpha n \\ \rho_2 n - \alpha n \end{bmatrix}_q}{\begin{bmatrix} n \\ \rho_2 n \end{bmatrix}_q} , \quad (5.3)$$

where the equality $\mathbb{P}\left(V \leq \boldsymbol{U}_2\right) = \frac{\left[\begin{smallmatrix} n-\alpha n \\ \rho_2 n - \alpha n \end{smallmatrix}\right]_q}{\binom{n}{\rho_2 n}}$ follows from the fact that the number of suspaces of $\mathbb{F}_q^n$ of dimension $\rho_2 n$ which contain a fixed subspace of dimension αn is precisely the number of subspaces of $\mathbb{F}_q^{n-\alpha n}$ of dimension $\rho_2 n - \alpha n$, i.e., $\left[\begin{smallmatrix} n-\alpha n \\ \rho_2 n - \alpha n \end{smallmatrix}\right]_q$. Using Lemma 5.5.3, (5.3) is at most

$$\begin{aligned} &K_q^{-3} \exp_q\left(\alpha n(\rho_1 n - \alpha n) + (\rho_2 n - \alpha n)(n - \rho_2 n) - \rho_2 n(n - \rho_2 n)\right) \\ &\qquad = K_q^{-3} \exp_q\left(-n^2\alpha(1+\alpha-\rho_1-\rho_2)\right) . \end{aligned}$$

Recalling $K_q^{-1} < 4$, the claim follows. □

With this claim in hand, we proceed to showing that if $\boldsymbol{X}_1, \boldsymbol{X}_2$ are uniformly and independently selected from $B_R(0,\rho)$, it is exponentially unlikely that $\boldsymbol{X}_1 + \boldsymbol{X}_2 \in B_R(Z,\rho)$, where Z is any fixed matrix.

Lemma 5.5.4. *Let $n \leq m$ be positive integers, $Z \in \mathbb{F}_q^{m \times n}$ a fixed matrix, and $\rho \in (0,1)$. Let $\boldsymbol{X}_1, \boldsymbol{X}_2 \sim B_R(0,\rho)$ be independent and uniform. Then, assuming n, m are sufficiently large compared to $1-\rho$,*

$$\mathbb{P}\left(\boldsymbol{X}_1 + \boldsymbol{X}_2 \in B_R(Z,\rho)\right) \leq q^{-\Omega_{\rho,b}(nm)} .$$

Informally, the proof proceeds as follows. First, we observe that it suffices to prove that it is exponentially unlikely that $\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z,\rho)$, where each $\boldsymbol{Y}_i$ is independently sampled by first choosing a subspace in $\mathbb{F}_q^n$ of dimension roughly ρn uniformly at random, then sampling m vectors from this subspace independently and uniformly at random, and setting them to be the rows of $\boldsymbol{Y}_i$. Claim 5.5.1 guarantees that the sum of the two random subspaces has large dimension except with probability $\exp(-\Omega(n^2))$. In this favorable case, $\boldsymbol{Y}_1 + \boldsymbol{Y}_2 - Z$ is obtained by sampling a reasonably large subspace of $\mathbb{F}_q^n$ and then sampling m vectors from affine shifts of this subspace, and a fairly simple analysis demonstrates that such a matrix is unlikely to have small rank.

The formal proof follows:

Proof. Let $\Phi = \mathbb{P}\left(\boldsymbol{X}_1 + \boldsymbol{X}_2 \in B_R(Z,\rho)\right)$ be the probability of interest. Let $r = \lfloor \rho n \rfloor$ and let $\eta = \eta(\rho, b) \in (0,1)$ be a parameter to be fixed later. Let $s_1, s_2 \leq r$ be integers such that, conditioned on $\operatorname{rank}(\boldsymbol{X}_1) = s_1$ and $\operatorname{rank}(\boldsymbol{X}_2) = s_2$, the probability Φ is maximized. That is, the pair (s_1, s_2) maximizes the expression

$$\mathbb{P}\left(\boldsymbol{X}_1, \boldsymbol{X}_2 \in B_R(Z,\rho) \middle| \operatorname{rank}(\boldsymbol{X}_i) = r_i, i = 1,2\right) .$$

Since there are at most n^2 choices for the pair (s_1, s_2) (as they must lie in the set $\{0, 1, \ldots, \lfloor \rho n \rfloor\}^2$), we have

$$\Phi \leq n^2 \cdot \mathbb{P}\left(\boldsymbol{X}_1 + \boldsymbol{X}_2 \in B_R(Z,\rho) \middle| \operatorname{rank}(\boldsymbol{X}_i) = s_i, i = 1,2\right) .$$

Next, note that if s_1 or s_2 is $\leq \eta r$, then since $|B_R(0,\eta\rho)|/|B_R(0,\rho)| \leq q^{-\Omega_\eta(nm)}$ (cf. Proposition 5.1.4), we conclude

$$\mathbb{P}\left(\operatorname{rank}(\boldsymbol{X}_1) \leq \eta r \vee \operatorname{rank}(\boldsymbol{X}_2) \leq \eta r\right) \leq q^{-\Omega_\eta(nm)} = q^{-\Omega_{\rho,b}(nm)} .$$

Thus, in this case, by the total probability rule,

$$\begin{aligned}
\Phi &= \sum_{(r_1,r_2)} \mathbb{P}\left(\boldsymbol{X}_1 + \boldsymbol{X}_2 \in B_R(Z,\rho) \wedge \operatorname{rank}(X_j) = r_i, i = 1,2\right) \\
&\leq n^2 \cdot \mathbb{P}\left(\operatorname{rank}(\boldsymbol{X}_i) = s_i, i = 1,2\right) \\
&\leq n^2 \cdot \mathbb{P}\left(\operatorname{rank}(\boldsymbol{X}_1) \leq \eta r \vee \operatorname{rank}(\boldsymbol{X}_2) \leq \eta r\right) \\
&\leq q^{-\Omega_{\rho,b}(nm)}.
\end{aligned}$$

Hence, we now assume $\eta r \leq s_i \leq r$ for $i = 1,2$. Let $\boldsymbol{Y}_i$ for $i = 1,2$ be independent random matrices sampled as follows:

(a) sample $\boldsymbol{U}_i$ uniformly at random among all dimension s_i subspaces of $\mathbb{F}_q^n$;

(b) sample m vectors uniformly and independently from $\boldsymbol{U}_i$ and set them as the rows of the matrix $\boldsymbol{Y}_i$.

For $i = 1, 2$, the matrix $\boldsymbol{Y}_i$ has rank s_i with probability at least

$$(1 - q^{-s_i})(1 - q^{-s_i+1}) \cdots (1 - q^{-2})(1 - q^{-1}) \geq \prod_{j=1}^{\infty}(1 - q^{-j}) \geq .288 > \frac{1}{4}$$

(this is actually the probability that the first s_i rows are linearly independent). Now, note that conditioned on obtaining rank s_i matrices, the random variables $(\boldsymbol{X}_1, \boldsymbol{X}_2)$ and $(\boldsymbol{Y}_1, \boldsymbol{Y}_2)$ are identically distributed: both are uniform over pairs of matrices (A_1, A_2) with $\mathrm{rank}(A_1) = s_1$ and $\mathrm{rank}(A_2) = s_2$. Let $\mathcal{E}$ denote the event that $\mathrm{rank}(\boldsymbol{X}_i) = s_i$ for $i = 1, 2$ and $\mathcal{F}$ the event $\mathrm{rank}(\boldsymbol{Y}_i) = s_i$ for $i = 1, 2$. Note that

$$\mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho)\right) \geq \mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Y, \rho) | \mathcal{F}\right) \cdot \mathbb{P}\left(\mathcal{F}\right) \ ,$$

so

$$\begin{aligned}\mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho) | \mathcal{F}\right) &\leq \frac{\mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho)\right)}{\mathbb{P}\left(\mathcal{F}\right)} \\ &\leq 4^2 \cdot \mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho)\right) \ .\end{aligned}$$

Recalling that

$$\Phi \leq n^2 \cdot \mathbb{P}\left(\boldsymbol{X}_1 + \boldsymbol{X}_2 \in B_R(Z, \rho) | \mathcal{E}\right) = n^2 \cdot \mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho) | \mathcal{F}\right) \ ,$$

we see that it suffices to prove

$$\mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho)\right) \leq q^{-\Omega_{\rho,b}(nm)} \ . \tag{5.4}$$

To prove Eq. (5.4), we use Claim 5.5.1. Let $\boldsymbol{U} = \boldsymbol{U}_1 + \boldsymbol{U}_2$. Since $0 < \rho_1, \rho_2 \leq \rho \in (0, 1)$, we can choose $\alpha \in (\max\{0, 1 - \rho_1 - \rho_2\}, \rho_1)$ (here, we assume wlog $\rho_1 \leq \rho_2$) such that $1 + \alpha - \rho_1 - \rho_2 > 0$ and $\rho_1 + \rho_2 - \alpha > \rho$. Let $\mathcal{G}$ denote the favorable event that $\dim(\boldsymbol{U}) > n(\rho_1 + \rho_2 - \alpha)$. Claim 5.5.1 states

$$\mathbb{P}\left(\neg\mathcal{G}\right) \leq 4^3 \exp_q(-n^2\alpha(1 + \alpha - \rho_1 - \rho_2)) = \exp_q(-\Omega_{\rho,b}(nm)) \ .$$

Utilizing Bayes' Rule,

$$\mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho)\right) \leq \mathbb{P}\left(\neg\mathcal{G}\right) + \mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho) | \mathcal{G}\right) \ ,$$

so we are reduced to bounding the second term. Note $\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z, \rho) \iff \boldsymbol{Y}_1 + \boldsymbol{Y}_2 - Z \in B_R(0, \rho)$. Let $z_1, \ldots, z_m \in \mathbb{F}_q^n$ denote the rows of Z. Let $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m$ denote the rows of $\boldsymbol{Y}_1 + \boldsymbol{Y}_2 - Z$, and observe that each $\boldsymbol{u}_i$ is uniform over the affine space $\boldsymbol{U} + z_i$ which, conditioned on $\mathcal{G}$, has cardinality at least $q^{(\rho+\gamma)n}$, where $\gamma = \rho_1 + \rho_2 - \alpha - \rho > 0$.

For $\boldsymbol{Y}_1 + \boldsymbol{Y}_2 - Z$ to have rank at most ρn, there must exist a set $S \subseteq [m]$ of cardinality ρn such that for all $i \in [m] \setminus S$, $\boldsymbol{u}_i \in \mathrm{span}\{\boldsymbol{u}_j : j \in S\} =: \boldsymbol{V}_S$. The probability of this event, conditioned on $\mathcal{G}$, is at most

$$\left(\frac{q^{\rho n}}{q^{(\rho+\gamma)n}}\right)^{m-\rho n} = \left(q^{-\gamma n}\right)^{\Omega_{\rho,b}(n)} = q^{-\Omega_{\rho,b}(nm)} \ .$$

Taking a union bound over all $\binom{m}{\rho m} \leq 2^{\Omega_{\rho,b}(n)}$ choices for S, we conclude

$$\mathbb{P}\left(\boldsymbol{Y}_1 + \boldsymbol{Y}_2 \in B_R(Z,\rho) | \mathcal{G}\right) \leq q^{-\Omega_{\rho,b}(nm)} ,$$

as desired.

□

We now show that if ℓ matrices from $B_R(0,\rho)$ are chosen at random, then it is unlikely that $\omega(\ell)$ of their linear combinations lie in $B_R(0,\rho)$. The proof combines Lemmas 5.4.2 and 5.5.4.

Lemma 5.5.5. *For every $\rho \in (0,1)$, there is a constant $K = K_{\rho,q,b} > 1$ such that for all integers $n \leq m$ and $\ell = o(\sqrt{nm})$, if $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_\ell$ are selected independently and uniformly at random from $B_R(0,\rho)$, then*

$$\mathbb{P}\left(|\operatorname{span}\{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_\ell\} \cap B_R(0,\rho)| \geq K \cdot \ell\right) \leq q^{-(4-o(1))nm} .$$

Proof. Let $L = K \cdot \ell$ (for some $K = K_{\rho,q,b}$ to be selected later) and let $c = 2$. Let $\delta = \delta_{\rho,b}$ be the constant in the $\Omega_{\rho,b}(\cdot)$ from Lemma 5.5.4. Let

$$\begin{aligned} d &= \left\lfloor \frac{1}{c}\log_q \frac{L}{2} - \left(1 - \frac{1}{c}\right)\log_q((q-1)\ell) \right\rfloor = \left\lfloor \frac{1}{2}\log_q \frac{L}{2} - \frac{1}{2}\log_q((q-1)\ell) \right\rfloor \\ &\geq \frac{1}{2}\log_q \frac{L}{2(q-1)\ell} - 1 = \frac{1}{2}\log_q \frac{C}{2(q-1)q^2}. \end{aligned}$$

Finally, for a vector $u \in \mathbb{F}_q^\ell$, let $\boldsymbol{X}(u) = \sum_i u_i \boldsymbol{X}_i$.

Towards proving the lemma, we prove the following claim:

Claim 5.5.6. *For any $S \subseteq \mathbb{F}_q^\ell$ with $|S| = L + 1$,*

$$\mathbb{P}\left(\forall v \in S, \boldsymbol{X}(v) \in B_R(0,\rho)\right) < q^{nm} q^{-\delta d nm} . \tag{5.5}$$

Proof of Claim 5.5.6. Let $w \in \mathbb{F}_q^\ell$ and $v_1, \ldots, v_d \in S$ be as given by Lemma 5.4.2. That is, $v_1 + w, v_2 + w, \ldots, v_d + w$ is a 2-increasing sequence. Then

$$\begin{aligned} \mathbb{P}\left(\forall v \in S, \boldsymbol{X}(v) \in B_R(0,\rho)\right) &\leq \mathbb{P}\left(\forall j \in [d], \boldsymbol{X}(v_j) \in B_R(0,\rho)\right) \\ &= \mathbb{P}\left(\forall j \in [d], \boldsymbol{X}(v_j) + \boldsymbol{X}(w) \in B_R(\boldsymbol{X}(w),\rho)\right) \\ &= \mathbb{P}\left(\forall j \in [d], \boldsymbol{X}(v_j + w) \in B_R(\boldsymbol{X}(w),\rho)\right) . \end{aligned}$$

Fix $Y \in \mathbb{F}_q^{m \times n}$. Then

$$\begin{aligned} &\mathbb{P}\left(\forall j \in [d], \boldsymbol{X}(v_j + w) \in B_R(Y,\rho)\right) \\ &= \prod_{j=1}^{d} \mathbb{P}\left(\boldsymbol{X}(v_j + w) \in B_R(Y,\rho) | \boldsymbol{X}(v_i + w) \in B_R(Y,\rho)\ \forall 1 \leq i \leq j-1\right) \\ &\leq \prod_{j=1}^{d} \max_{\substack{Z_k \in B_R(0,\rho): \\ k \in \bigcup_{i=1}^{j-1} \operatorname{supp}(v_i + w)}} \mathbb{P}\left(\boldsymbol{X}(v_j + w) \in B_R(Y,\rho) | \boldsymbol{X}_k = Z_k\ \forall k \in \bigcup_{i=1}^{j-1} \operatorname{supp}(v_i + w)\right) \\ &\leq \left(q^{-\delta nm}\right)^d . \end{aligned}$$

The last inequality follows from Lemma 5.5.4 as follows: let i_1, i_2 be distinct elements of $\operatorname{supp}(v_j + w) \setminus \bigcup_{i=1}^{j-1} \operatorname{supp}(v_i + w)$ (which exist thanks to the 2-increasing property). Then apply Lemma 5.5.4 with $(v_j)_{i_1} \boldsymbol{X}_{i_1}$ and $(v_j)_{i_2} \boldsymbol{X}_{i_2}$ (which are distributed uniformly over $B_R(0, \rho)$), and $Z = Y - \sum_{k \in [\ell] \setminus \{i_1, i_2\}} (v_j + w)_k Z_k$ (which is a fixed matrix).

By taking a union bound over all q^{nm} choices of $Y \in \mathbb{F}_q^{m \times n}$, the claim follows. □

We now bound the probability that more than L elements of $\operatorname{span}\{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_\ell\}$ lie in $B_R(0, \rho)$. This occurs iff there exists a subset $S \subseteq \mathbb{F}_q^\ell$ of size $L + 1$ such that $\forall v \in S$, $\boldsymbol{X}(v) \in B_R(0, \rho)$. By taking a union bound over the probability in (5.5), this occurs with probability at most $q^{\ell(L+1)} q^{nm} q^{-\delta dnm}$. Assuming $C = C_{\rho,q}$ is large enough so that $d \geq \frac{5}{\delta}$, this probability is at most

$$q^{o(nm)+nm-5nm} = q^{-(4-o(1))nm} \,. \qquad \square$$

We are now prepared to prove Theorem 5.3.1, which we restate for convenience.

Theorem 5.3.1 (Main Theorem)**.** *Let $\rho \in (0, 1)$ and $n \leq m$. There exists a constant $C = C_{\rho,q,b} > 0$ such that a random $\mathbb{F}_q$-linear rank metric code $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^{m \times n}$ of rate $R = 1 - \psi_b(\rho) - \varepsilon$ is $(\rho, C/\varepsilon)$-list-decodable with high probability.*

Proof. Let $C = 2K$, where K is the constant from Lemma 5.5.5, let $L = \lceil \frac{C}{\varepsilon} \rceil$, and let n, m be larger than L and sufficiently large so that the $o(1)$ term of Lemma 5.5.5 is at most 1.

For $\boldsymbol{Z} \in \mathbb{F}_q^{m \times n}$ selected uniformly at random, we will study the quantity

$$\Phi := \mathbb{P}\left(|B_R(\boldsymbol{Z}, \rho) \cap \boldsymbol{\mathcal{C}}| \geq L\right) \,.$$

By taking a union bound over all $Z \in \mathbb{F}_q^{m \times n}$, note that proving $\Phi \leq q^{-nm} \cdot q^{-nm}$ will suffice to conclude the theorem.

As a first step, we observe that we can move $\boldsymbol{Z}$ to the origin without significantly changing the probability Φ. Indeed, if $\boldsymbol{\mathcal{C}} = \operatorname{span}\{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_{Rnm}\}$ for $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_{Rnm} \in \mathbb{F}_q^{m \times n}$ sampled independently and uniformly and $\boldsymbol{\mathcal{C}}^* = \boldsymbol{\mathcal{C}} + \boldsymbol{Z}$, we have

$$\begin{aligned}
\Phi &= \mathbb{P}\left(|B_R(\boldsymbol{Z}, \rho) \cap \boldsymbol{\mathcal{C}}| \geq L\right) \\
&= \mathbb{P}\left(|B_R(0, \rho) \cap (\boldsymbol{\mathcal{C}} + \boldsymbol{Z})| \geq L\right) \\
&\leq \mathbb{P}\left(|B_R(0, \rho) \cap \boldsymbol{\mathcal{C}}^*| \geq L\right) \,.
\end{aligned}$$

Thus, it suffices to bound the probability $|B_R(0, \rho) \cap \boldsymbol{\mathcal{C}}^*| \geq L$, where we now have that $\boldsymbol{\mathcal{C}}^*$ is a random linear code of dimension $Rnm + 1$.

Now, for each integer ℓ satisfying $\log_q L \leq \ell \leq L$, let $\mathcal{F}_\ell$ denote the set of all tuples $(A_1, \ldots, A_\ell) \in B_R(0, \rho)^\ell$ such that $A_1, \ldots, A_\ell$ are linearly independent and $|\operatorname{span}\{A_1, \ldots, A_\ell\} \cap B_R(0, \rho)| \geq L$. Let

$$\mathcal{F} = \bigcup_{\log_q L \leq \ell \leq L} \mathcal{F}_\ell \,.$$

Denote $\mathcal{A} = (A_1, \ldots, A_\ell)$ and, as a slight abuse of notation, we write $\boldsymbol{\mathcal{C}}^* \supseteq \mathcal{A}$ to mean that $A_i \in \boldsymbol{\mathcal{C}}^*$ for all $i \in [\ell]$.

Towards bounding $\mathbb{P}\left(|B_R(0,\rho) \cap \boldsymbol{\mathcal{C}}^*| \geq L\right)$, notice that if $|B_R(0,\rho) \cap \boldsymbol{\mathcal{C}}^*| \geq L$, then there must exist some $\mathcal{A} \in \mathcal{F}$ for which $\boldsymbol{\mathcal{C}}^* \supseteq \mathcal{A}$. Indeed, we may choose any maximal linearly independent subset of $B_R(0,\rho) \cap \boldsymbol{\mathcal{C}}^*$ if this set has size at most L, or any linearly independent subset of $B_R(0,\rho) \cap \boldsymbol{\mathcal{C}}^*$ of size L otherwise.

Thus, by a union bound,

$$\Phi \leq \sum_{\mathcal{A} \in \mathcal{F}} \mathbb{P}\left(\boldsymbol{\mathcal{C}}^* \supseteq \mathcal{A}\right) = \sum_{\ell=\lceil \log_q L \rceil}^{L} \sum_{\mathcal{A} \in \mathcal{F}_\ell} \mathbb{P}\left(\boldsymbol{\mathcal{C}}^* \supseteq \mathcal{A}\right) \ .$$

Note that for $\mathcal{A} = (A_1, \ldots, A_\ell) \in \mathcal{F}$, by linear independence we have

$$\mathbb{P}\left(\boldsymbol{\mathcal{C}}^* \supseteq \mathcal{A}\right) = \left(\frac{q^{Rnm+1}}{q^{nm}}\right)^{\ell} \ .$$

Thus, we find

$$\Phi \leq \sum_{\ell=\lceil \log_q L \rceil}^{L} |\mathcal{F}_\ell| \cdot \left(\frac{q^{Rnm+1}}{q^{nm}}\right)^{\ell} \ .$$

We now bound $|\mathcal{F}_\ell|$ depending on the value of ℓ.

- *Case 1:* $\ell < \frac{3}{\varepsilon}$.

  In this case, note that $\frac{|\mathcal{F}_\ell|}{|B_R(0,\rho)|^\ell}$ is a lower bound on the probability that ℓ matrices $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_\ell$ chosen independently and uniformly at random from $B_R(0,\rho)$ are such that

  $$|\operatorname{span}\{\boldsymbol{X}_1, \ldots, \boldsymbol{X}_\ell\} \cap B_R(0,\rho)| \geq L \ .$$

  Lemma 5.5.5 tells us that this probability is at most q^{-3nm}. Thus,

  $$|\mathcal{F}_\ell| \leq |B_R(0,\rho)|^\ell q^{-3nm} \leq \left(4q^{mn\psi_b(\rho)}\right)^{\ell} \cdot q^{-3nm} \ .$$

- *Case 2:* $\ell \geq \frac{3}{\varepsilon}$.

  In this case, we have the (simple) bound of

  $$|\mathcal{F}_\ell| \leq |B_R(0,\rho)|^\ell \leq \left(4q^{mn\psi_b(\rho)}\right)^{\ell} \ .$$

Combining these inequalities, we obtain the following bound:

$$\begin{aligned}
\Phi &\leq \sum_{\ell=\lceil \log_q L \rceil}^{\lceil \frac{3}{\varepsilon} \rceil - 1} |\mathcal{F}_\ell| \cdot \left(\frac{q^{Rnm+1}}{q^{nm}} \right)^\ell + \sum_{\ell=\lceil \frac{3}{\varepsilon} \rceil}^{L} |\mathcal{F}_\ell| \cdot \left(\frac{q^{Rnm+1}}{q^{nm}} \right)^\ell \\
&\leq \sum_{\ell=\lceil \log_q L \rceil}^{\lceil \frac{3}{\varepsilon} \rceil - 1} \left(4q^{mn\psi_b(\rho)} \right)^\ell \cdot q^{-3nm} \cdot \left(\frac{q^{Rnm}}{q^{nm}} \right)^\ell \cdot q^\ell + \sum_{\ell=\lceil \frac{3}{\varepsilon} \rceil}^{L} \left(4q^{mn\psi_b(\rho)} \right)^\ell \cdot \left(\frac{q^{Rnm}}{q^{nm}} \right)^\ell \cdot q^\ell \\
&\leq q^{-3nm} \sum_{\ell=\lceil \log_q L \rceil}^{\lceil \frac{3}{\varepsilon} \rceil - 1} 4^\ell \cdot q^\ell \cdot q^{(-\varepsilon nm)\ell} + \sum_{\ell=\lceil \frac{3}{\varepsilon} \rceil}^{L} 4^\ell \cdot q^\ell \cdot q^{(-\varepsilon nm)\ell} \\
&\leq (4q)^L \left(q^{-3nm} \cdot \frac{3}{\varepsilon} + L \cdot q^{-\varepsilon nm \cdot \frac{3}{\varepsilon}} \right) \\
&< q^{-nm} \cdot q^{-nm}
\end{aligned}$$

assuming n, m are large enough compared to ε. □

5.6 Open Problems

Many open directions remain to be pursued; we mention a couple of problems that we find particularly interesting. First of all, we are unable to give good control of the list size when $\rho \to 1$ or when $b \to 0$. As alluded to earlier, this is inherent in the [GHK11] analysis of the list-decodability of random linear codes in the Hamming metric case, and unfortunately our analysis inherits this limitation. Recall that in the case of the Hamming metric, different techniques were developed in order to understand the high noise regime: Gaussian processes and chaining [CGV13; Woo13; RW14] and structure vs. pseudorandomness [RW18]. A natural hope would be to port these ideas over to the rank metric; unfortunately, there does not appear to be an obvious way to make this work. Lastly, we comment that the recent Li-Wootters [LW18] does indeed apply to rank metric codes and therefore obtains the optimal result when $q = 2$, but generalizing this approach to larger q appears nontrivial.

Lastly, we note that it is common to view a rank metric code $\mathcal{C}$ as a subset of $\mathbb{F}_{q^m}^n$, and then insist that such a code be $\mathbb{F}_{q^m}$-linear. This is done by fixing a basis for $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$ and then identifying a vector $x \in \mathbb{F}_{q^m}^n$ with the matrix $X \in \mathbb{F}_q^{m \times n}$, where the ith column of X is x_i written in the coordinates defined by the basis. Thus, it is natural to ask if a random $\mathbb{F}_{q^m}$-linear subspace $\mathcal{C} \subset \mathbb{F}_{q^m}^n$ is rank metric list-decodable. By adjusting the constant C in the proof of Lemma 5.5.5, one can see that the proof still goes through. Unfortunately, C will have to grow polynomially in q^m (rather than just q), so the resulting list sizes will be on the order of $q^{O(m)}/\varepsilon$. Thus, we are unable to conclude that random $\mathbb{F}_{q^m}$-linear codes are rank metric list-decodable with polynomial list sizes, let alone prove

the optimal $O(1/\varepsilon)$ list size.[1] Indeed, the situation is even more dire: we are currently unaware of a proof that *any* $\mathbb{F}_{q^m}$-linear rank-metric codes are list-decodable beyond half the minimum distance (the codes constructed by Guruswami, Wang and Xing [GWX16] are just $\mathbb{F}_q$-linear, not $\mathbb{F}_{q^m}$-linear). Thus, existentially proving that some $\mathbb{F}_{q^m}$-linear rank metric code is list-decodable or concluding that no such code exists would represent an important step forward in our understanding of the list-decodability of rank metric codes.

[1] A bound of size $q^{O(m)/\varepsilon}$ follows from Zyablov-Pinsker style considerations, so this is perhaps a mild improvement over what was known previously, but certainly not a very impressive result.

Chapter 6

Average-Radius List-Decodability of Binary Random Linear Codes

In this chapter, we strengthen an argument of Li and Wootters (which is itself a strengthening of an argument of Guruswami, Håstad, Sudan and Zuckerman) to show that random linear codes over $\mathbb{F}_2$ of rate $1 - h_2(\rho) - \varepsilon$ are (ρ, L)-*average-radius* list-decodable, where $L = O(1/\varepsilon)$. In fact, just as Li and Wootters did for absolute-radius list-decoding, we nail down the constant in the big-O notation to obtain $L = \lfloor h_2(\rho)/\varepsilon + 1 \rfloor$.

Furthermore, just as is the case for the argument of Li and Wootters, we observe that the same techniques apply equally well over the rank metric. In this way, we deduce that random linear rank metric codes over $\mathbb{F}_2$ of rate $1 - \psi_b(\rho) - \varepsilon$ are (ρ, L)-*average-radius* list-decodable, where $L = \lfloor \psi_b(\rho)/\varepsilon + 1 \rfloor$.

6.1 Overview of Approach

In this chapter we prove the following theorem. Recall that we abbreviate $h(\rho) = h_2(\rho)$.

Theorem 6.1.1. *Let $n \in \mathbb{N}$. Let $\rho \in (0, \frac{1}{2})$ and $R = 1 - h(\rho) - \varepsilon$, where $0 < \varepsilon < 1 - h(\rho)$. Let $L = \left\lfloor \frac{h(\rho)}{\varepsilon} + 1 \right\rfloor$. Then, a random linear code $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_2^n$ of rate R is (ρ, L)-average-radius list-decodable with probability $1 - 2^{-\Omega_{\rho,\varepsilon}(n)}$.*

Our argument closely follows that of [LW18] which itself builds on the argument of [Gur+02]. The argument imagines building the random linear code one dimension at a time and uses a potential function to show that, so long as we don't add too many dimensions, no ball intersects the code too much. We now provide an informal overview of our approach, specifically comparing and contrasting it with the arguments of Guruswami, Håstad, Sudan and Zuckerman [Gur+02]; and Li and Wootters [LW18].

Let $R = 1 - h(\rho) - \varepsilon$ and put $k := Rn$ (which we assume is an integer). Note that sampling a random linear code of rate R is the same as sampling $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \in \mathbb{F}_2^n$ independently and uniformly at random and outputting $\mathrm{span}\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\}$. Consider the

"intermediate" codes $\boldsymbol{\mathcal{C}}_i = \operatorname{span}\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_i\}$; [LW18] (following [Gur+02]) define a potential function $S_{\boldsymbol{\mathcal{C}}_i}$ and show that it remains small. [Gur+02] demonstrated that this holds in expectation; [LW18] improved their argument to show that it holds with high probability. It is easy to show that, so long as $S_{\mathcal{C}}$ is $O(1)$, the code $\mathcal{C}$ is suitably list-decodable.

We now describe this potential function in more detail. First, for a code $\mathcal{C}$ and a vector $x \in \mathbb{F}_2^n$, define

$$L_{\mathcal{C}}(x) := |B(x, \rho) \cap \mathcal{C}| \, .$$

Note that (ρ, L)-list-decodability is equivalent to $L_{\mathcal{C}}(x) \leq L$ for all x. In [Gur+02], the authors define

$$S_{\mathcal{C}} := \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} 2^{\varepsilon n L_{\mathcal{C}}(x)}$$

and observe that, for any $b_1, \ldots, b_i \in \mathbb{F}_2^n$,

$$\mathop{\mathbb{E}}_{\boldsymbol{b}_{i+1} \sim \mathbb{F}_2^n} \left[S_{\mathcal{C}_i + \{0, \boldsymbol{b}_{i+1}\}} \right] = S_{\mathcal{C}_i}^2 \, ,$$

where $\mathcal{C}_i = \operatorname{span}\{b_1, \ldots, b_i\}$. That is, the potential function squares in expectation, so the probabilistic method guarantees that we can choose some b_{i+1} for which $S_{\mathcal{C}_{i+1}} \leq S_{\mathcal{C}_i}$. Thus, for some choice of $b_1, \ldots, b_k$, one has $S_{\mathcal{C}_k} \leq (S_{\{0\}})^{2^k}$.

In [LW18], the definition of $S_{\mathcal{C}}$ is slightly modified:

$$S_{\mathcal{C}} := \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} 2^{\frac{\varepsilon n L_{\mathcal{C}}(x)}{1+\varepsilon}} \, .$$

This little bit of extra room allows to show that, in fact, with high probability over the choice of $\boldsymbol{b}_{i+1}$, $S_{\mathcal{C}_i + \{0, \boldsymbol{b}_{i+1}\}} \leq S_{\mathcal{C}_i}^2$. By a union bound, it follows that with high probability, $S_{\boldsymbol{\mathcal{C}}_k} \leq (S_{\{0\}})^{2^k}$.

In either case, to conclude the proof, one observes the bound[1] $S_{\{0\}} \leq 1 + 2^{-n(1-h(\rho)-\varepsilon)}$ and then uses

$$S_{\boldsymbol{\mathcal{C}}_k} \leq (S_{\{0\}})^{2^k} \leq (2^{-n(1-h(\rho)-\varepsilon)})^{2^k} \leq \exp\left(2^{k-n(1-h(\rho)-\varepsilon)}\right) \leq O(1)$$

for k chosen as above.

6.1.1 Alterations for Average-Radius List-Decoding

While this argument analyzes the absolute-radius list-decodability of random linear codes very effectively, it is not immediately clear how to generalize the argument to study average-radius list-decodability. We now introduce the additional ideas we need

[1] Actually, for the Li-Wootters potential function, one has $S_{\{0\}} \leq 1 + 2^{-n(1-h(\rho)-\frac{\varepsilon}{1+\varepsilon})}$, but this difference is unimportant.

to derive Theorem 6.1.1. We fix a threshold parameter $\lambda \in (0, \frac{1}{2})$ for which $h(\lambda) < 1 - R = h(\rho) + \varepsilon$ and put

$$\eta := 1 - R - h(\lambda) .$$

The value of λ (and hence η) will be fixed later.

We define the function $M_{R,\lambda} : [0,1] \to \mathbb{R}$ by

$$M_{R,\lambda}(\gamma) := \begin{cases} 1 - R - h(\gamma) & \text{if } \gamma < \lambda \\ 0 & \text{if } \gamma \geq \lambda \end{cases} .$$

Remark 6.1.2. One can think of this quantity as a sort of "normalized entropy change" up to the threshold λ. Recalling $1 - R = h(\rho) + \varepsilon$, if $\gamma < \lambda$, then

$$M_{R,\lambda}(\gamma) \approx \frac{1}{n}(h(\rho) - h(\lambda)) \approx \log\left(\frac{|B^n(0,\rho)|}{|B^n(0,\gamma)|}\right) .$$

Hence, $M_{R,\lambda}(\gamma)$ is something like a normalized "surprise" an observer would experience if they are expecting a random vector of weight $\leq \rho$ and see a vector of weight $\leq \gamma$.

For a linear code $\mathcal{C} \leq \mathbb{F}_2^n$ and $x \in \mathbb{F}_2^n$ we define

$$L_{\mathcal{C},R,\lambda}(x) := \sum_{y \in \mathcal{C}} M_{R,\lambda}(d(x,y)) .$$

This is intuitively the "smoothed-out" list size of x, where nearby codewords are weighted more heavily than far away codewords, and the weighting is given by the "entropy change" implied by the distance from x to y.

Next, we define

$$A_{\mathcal{C},R,\lambda}(x) := 2^{\frac{nL_{\mathcal{C},R,\lambda}(x)}{1+\eta}}$$

and

$$S_{\mathcal{C},R,\lambda} := \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} A_{\mathcal{C},R,\lambda}(x) .$$

The quantity $S_{\mathcal{C},R,\lambda}$ is the potential function we will analyze.

6.2 The Proof

In this section we prove Theorem 6.1.1. As R and λ (and hence $\eta = 1 - R - h(\lambda)$) will be fixed throughout,[2] we will suppress their dependence and simply write $M(x)$, $L_{\mathcal{C}}(x)$, $A_{\mathcal{C}}(x)$ and $S_{\mathcal{C}}$.

First, we observe that the following analog of [LW18, Lemma 3.2] holds. The proof is a simple adaptation of theirs (which in turn follows [Gur+02]).

[2] Although the precise value of λ will be determined later.

Lemma 6.2.1. *For all $\mathcal{C} \leq \mathbb{F}_2^n$ and $b \in \mathbb{F}_2^n$,*

$$L_{\mathcal{C}+\{0,b\}}(x) \leq L_{\mathcal{C}}(x) + L_{\mathcal{C}}(x+b)\ , \tag{6.1}$$
$$A_{\mathcal{C}+\{0,b\}}(x) \leq A_{\mathcal{C}}(x) \cdot A_{\mathcal{C}}(x+b)\ . \tag{6.2}$$

Moreover, equality holds if and only if $b \notin \mathcal{C}$.

Proof. To derive (6.1):

$$\begin{aligned} L_{\mathcal{C}+\{0,b\}}(x) &= \sum_{y \in \mathcal{C}+\{0,b\}} M(d(x,y)) \\ &\leq \sum_{y \in \mathcal{C}} M(d(x,y)) + \sum_{y \in \mathcal{C}+b} M(d(x,y)) \\ &= \sum_{y \in \mathcal{C}} M(d(x,y)) + \sum_{y \in \mathcal{C}} M(d(x,y-b)) \\ &= \sum_{y \in \mathcal{C}} M(d(x,y)) + \sum_{y \in \mathcal{C}} M(d(x+b,y)) \\ &= L_{\mathcal{C}}(x) + L_{\mathcal{C}}(x+b)\ , \end{aligned}$$

and equality holds in the second line iff $\mathcal{C} \cap (\mathcal{C}+b) = \emptyset$, which holds iff $b \notin \mathcal{C}$. (6.2) follows immediately. □

Next, we bound $S_{\{0\}}$. We have

$$\begin{aligned} S_{\{0\}} &\leq 1 + 2^{-n} \sum_{\substack{x \in \mathbb{F}_2^n \\ \mathrm{wt}(x) \leq \lambda}} 2^{\frac{n \cdot (1 - R - h(\mathrm{wt}(x)))}{1+\eta}} \\ &\leq 1 + \sum_{i=0}^{\lfloor \lambda n \rfloor} 2^{-n\left(1 - h(i/n) - \frac{h(\lambda)+\eta-h(i/n)}{1+\eta}\right)}\ . \end{aligned}$$

As this sum is dominated by its last term, we deduce

$$S_{\{0\}} \leq 1 + (\lambda n) 2^{-n\left(1-h(\lambda)-\frac{\eta}{1+\eta}\right)}\ . \tag{6.3}$$

From here, following the argument of Li and Wootters we can combine Lemma 6.2.1 and Eq. (6.3) to deduce the following.

Lemma 6.2.2. *Let $\rho \in (0, \frac{1}{2})$ and $R = 1 - h(\rho) - \varepsilon$ for $0 < \varepsilon < 1 - h(\rho)$. Let $\boldsymbol{\mathcal{C}}_{Rn} \leq \mathbb{F}_2^n$ be a random linear code of rate R. Then with probability $1 - \exp(-\Omega_\eta(n))$, $S_{\boldsymbol{\mathcal{C}}_{Rn}} \leq 2$.*

The proof of this lemma is completely analogous to that of [LW18, Lemma 3.3]. The core of the proof is encapsulated by the following claim which crucially uses pairwise independence and the field size of 2; it is completely analogous to [LW18, Lemma 3.4]. Following them, we define

$$B_{\mathcal{C}}(x) := A_{\mathcal{C}}(x) - 1 \text{ and } T_{\mathcal{C}} := S_{\mathcal{C}} - 1\ .$$

Claim 6.2.3. *Suppose that $\mathcal{C} \leq \mathbb{F}_2^n$ is a fixed code satisfying $T_{\mathcal{C}} < 1$. Then*

$$\mathop{\mathbb{P}}_{\boldsymbol{b} \sim \mathbb{F}_2^n} \left(S_{\mathcal{C} + \{0, \boldsymbol{b}\}} > 1 + 2T_{\mathcal{C}} + T_{\mathcal{C}}^{1.5} \right) < T_{\mathcal{C}}^{0.5} \, .$$

To conclude the lemma, one only needs to be careful about the growth rate of $S_{\mathcal{C}}$. In particular, the proof crucially uses that η is positive. We again choose vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{Rn}$ independently and uniformly at random. If $\boldsymbol{\mathcal{C}}_i = \operatorname{span}\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_i\}$, we need that $S_{\boldsymbol{\mathcal{C}}_i} \leq 1 + 2^{-\Omega(n)}$ in expectation for all i for the error bounds to succeed. As we expect the $o(1)$ term to roughly double, we need $2^{Rn} \cdot T_{\{0\}} \approx 2^{-n\left(\eta - \frac{\eta}{1+\eta}\right)} \leq 2^{-\Omega_\eta(n)}$.

Thus, in order to conclude Theorem 6.1.1, we are simply required to demonstrate that $S_{\mathcal{C}} \leq 2$ implies that $\mathcal{C}$ is (ρ, L)-average-radius list-decodable: this is the crux of our contribution. The main lemma we require is the following.

Lemma 6.2.4. *Let $\mathcal{C} \leq \mathbb{F}_2^n$ be a linear code of rate R such that $S_{\mathcal{C}} \leq 2$. Then, for all $x \in \mathbb{F}_2^n$ and $D \subseteq \mathcal{C} \cap B(x, \lambda)$, it holds that*

$$\sum_{y \in D} h(d(x,y)) \geq (|D| - 1 - \eta)(1 - R) - \frac{1+\eta}{n} \, .$$

Proof. First, observe that

$$\begin{aligned} L_{\mathcal{C}}(x) &\geq \sum_{y \in D} \left((1 - R) - h(d(x,y)) \right) = |D|(1-R) - \sum_{y \in D} h(\delta(x,y)) \, , \\ \text{so } \log A_{\mathcal{C}}(x) &\geq n \frac{|D|(1-R) - \sum_{y \in D} h(\delta(x,y))}{1+\eta} \, . \end{aligned} \tag{6.4}$$

Next, as $d(x,y) = d(x+z, y+z)$ for any $z \in \mathbb{F}_2^n$, observe that for any $x \in \mathbb{F}_2^n$ and $c \in \mathcal{C}$, $L_{\mathcal{C}}(x) = L_{\mathcal{C}}(x+c)$ and hence $A_{\mathcal{C}}(x) = A_{\mathcal{C}}(x+c)$. Thus, $\max_{x \in \mathbb{F}_2^n} A_{\mathcal{C}}(x)$ is attained at at least $|\mathcal{C}|$ different values of x, so

$$S_{\mathcal{C}} = \frac{1}{2^n} \sum_{x \in \mathbb{F}_2^n} A_{\mathcal{C}}(x) \geq \frac{|\mathcal{C}|}{2^n} \cdot \max_{x \in \mathbb{F}_2^n} A_{\mathcal{C}}(x) = 2^{-(1-R)n} \cdot \max_{x \in \mathbb{F}_2^n} A_{\mathcal{C}}(x) \, .$$

Combining this with (6.4) we conclude that for any $x \in \mathbb{F}_2^n$,

$$\begin{aligned} 1 \geq \log_2 S_{\mathcal{C}} &\geq -(1-R)n + \log_2 \left(A_{\mathcal{C}}(x) \right) \\ &\geq n \cdot \left(-(1-R) + \frac{|D|(1-R) - \sum_{y \in D} h(d(x,y))}{1+\eta} \right) \\ &= n \cdot \frac{(|D| - 1 - \eta)(1-R) - \sum_{y \in D} h(d(x,y))}{1+\eta} \, . \end{aligned}$$

Rearranging yields the lemma. □

We may now conclude Theorem 6.1.1.

Proof of Theorem 6.1.1. Since $L > \frac{h(\rho)}{\varepsilon} = \frac{1-R}{\varepsilon} - 1$, there exists $\eta > 0$ small enough so that for all sufficiently large n

$$L > \frac{1 - R + \eta + \frac{1+\eta}{n}}{\varepsilon - \eta} - 1 \,. \tag{6.5}$$

Thus, we define λ so that η (which we defined as $\eta = 1 - R - h(\lambda)$) satisfies (6.5). Let $\boldsymbol{\mathcal{C}}$ be a random linear code of rate R. Due to Lemma 6.2.2, the conclusion of Lemma 6.2.4 holds with probability $1 - 2^{-\Omega_{R,\rho}(n)}$ for $\boldsymbol{\mathcal{C}}$. It remains to show that, assuming n is sufficiently large, any code $\mathcal{C}$ satisfying the conclusion of Lemma 6.2.4 is (ρ, L)-average-radius list-decodable.

Let $x \in \mathbb{F}_2^n$ and $\Lambda \subseteq \mathcal{C}$ such that $|\Lambda| = L + 1$; our goal is to show that

$$\frac{1}{L+1} \sum_{y \in \Lambda} d(x, y) > \rho \,. \tag{6.6}$$

Let

$$D = \{y \in \Lambda : d(x, y) \leq \lambda\}$$

and define

$$h^*(\alpha) = \begin{cases} h(\alpha) & \text{if } \alpha \leq \frac{1}{2} \\ 1 & \text{if } x > \frac{1}{2} \end{cases} \,.$$

Now,

$$\begin{aligned}
\sum_{y \in \Lambda} h^*(d(x, y)) &\geq \sum_{y \in D} h(\delta(x, y)) + (L - |D|)h(\lambda) && (6.7) \\
&\geq (|D| - 1 - \eta)(1 - R) + (L - |D|)(1 - R - \eta) - \frac{1+\eta}{n} && (6.8) \\
&= (1 - R) \cdot (L - 1) - \eta \cdot (1 - R) - \eta \cdot (L - |D|) - \frac{1+\eta}{n} \\
&\geq (1 - R) \cdot (L - 1) - \eta \cdot (L + 1) - \frac{1+\eta}{n} \\
&= (1 - R)L - (1 - R) - \eta \cdot (L + 1) - \frac{1+\eta}{n} \\
&= Lh(p) - (1 - R) - (L + 1)\eta + L\varepsilon - \frac{1+\eta}{n} && (6.9) \\
&= Lh(p) - (1 - R + \eta) + (L - 1)(\varepsilon - \eta) - \frac{1+\eta}{n} \\
&> Lh(p) \,. && (6.10)
\end{aligned}$$

Here, the Inequality (6.7) holds because $h^*(\alpha) > h(\lambda)$ for all $\alpha > \lambda$; Inequality (6.8) is the conclusion of Lemma 6.2.4; Equality (6.9) follows from the fact that $R = 1 - h(p) - \varepsilon$; and Inequality (6.10) follows from (6.5). Thus, we deduce

$$\frac{1}{L+1} \sum_{y \in \Lambda} h^*(d(x, y)) > h(\rho) \,. \tag{6.11}$$

Since h^* is concave,

$$h^*\left(\frac{1}{L+1}\sum_{y\in\Lambda} h^*(d(x,y))\right) \geq \frac{1}{L+1}\sum_{y\in\Lambda} h^*(d(x,y))\ ,$$

and so (6.6) follows from (6.11), the monotonicity of h^* and the fact that $h^*(\rho) = h(\rho)$. $\square$

6.3 Rank Metric

Pleasingly, just as the argument in [LW18] generalizes easily to the case of rank metric codes, the same holds for the argument given above. We just describe the changes one needs to make the definitions given in Section 6.1 and leave to the reader the straightforward verification that the proof of Section 6.2 holds *mutatis mutandis*.

The theorem we obtain is as follows:

Theorem 6.3.1. *Let $n \leq m \in \mathbb{N}$ and put $b = \frac{n}{m}$. Let $\rho \in (0,1)$ and $R = 1 - \psi_b(\rho) - \varepsilon$, where $0 < \varepsilon < 1 - \psi_b(\rho)$. Let $L = \left\lfloor \frac{\psi_b(\rho)}{\varepsilon} + 1 \right\rfloor$. Then, a random linear code $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_2^{m\times n}$ of rate R is (ρ, L)-average-radius list-decodable with probability $1 - 2^{-\Omega_{\varepsilon,\rho}(n)}$.*

To remove notational clutter, in the remainder of this section $b \in (0,1)$ will be fixed and we let $\psi(\rho) := \psi_b(\rho)$.

Similar to what was done before, we will fix a threshold parameter $\lambda \in (0,1)$ for which $\psi(\lambda) < 1 - R = \psi(\rho) + \varepsilon$, and put $\eta = 1 - R - \psi(\lambda)$. We then define the function $M_{R,\lambda} : [0,1] \to \mathbb{R}$ by

$$M_{R,\lambda}(\gamma) := \begin{cases} 1 - R - \psi(\gamma) & \text{if } \gamma < \lambda \\ 0 & \text{if } \gamma \geq \lambda \end{cases}\ .$$

Again, this is some sort of "normalized entropy change"; cf. Remark 6.1.2.

Next, for a linear code $\mathcal{C} \leq \mathbb{F}_2^{m\times n}$ and $X \in \mathbb{F}_2^{m\times n}$ we define

$$L_{\mathcal{C},R,\lambda}(X) := \sum_{y\in\mathcal{C}} M_{R,\lambda}(d(x,y))\ .$$

Next, define

$$A_{\mathcal{C},R,\lambda}(X) := 2^{\frac{nmL_{\mathcal{C},R,\lambda}(X)}{1+\eta}}$$

and

$$S_{\mathcal{C},R,\lambda} := \frac{1}{2^{nm}} \sum_{X\in\mathbb{F}_2^{m\times n}} A_{\mathcal{C},R,\lambda}(X)\ .$$

It is not difficult now to reuse the arguments of Section 6.2 with these definitions to derive Theorem 6.3.1.

Chapter 7

Tensor Codes: List-Decodable Codes with Efficient Algorithms

We continue the study of list-decoding and recovery properties of high-rate tensor codes, initiated by Hemenway, Ron-Zewi, and Wootters [HRZW17]. In that work it was shown that the tensor product of an efficient (poly-time) high-rate globally list-recoverable code is *approximately* locally list-recoverable, as well as globally list-recoverable in *probabilistic* near-linear time. This was used in turn to give the first capacity-achieving list-decodable codes with (a) local list-decoding algorithms, and (b) with *probabilistic* near-linear time global list-decoding algorithms. This also yielded constant-rate codes approaching the Gilbert-Varshamov bound with *probabilistic* near-linear time global unique-decoding algorithms.

In the current work we obtain the following results:

1. The tensor product of an efficient (poly-time) high-rate globally list-recoverable code is globally list-recoverable in *deterministic* near-linear time. This yields in turn the first capacity-achieving list-decodable codes with *deterministic* near-linear time global list-decoding algorithms. It also gives constant-rate codes approaching the Gilbert-Varshamov bound with *deterministic* near-linear time global unique-decoding algorithms.
2. If the base code is additionally locally correctable, then the tensor product is (genuinely) locally list-recoverable. This yields in turn (non-explicit) constant-rate codes approaching the Gilbert-Varshamov bound that are *locally correctable* with query complexity and running time $n^{o(1)}$. This improves over prior work by Gopi et. al. [Gop+18] that only gave query complexity n^{ε} with rate that is exponentially small in $1/\varepsilon$.
3. A nearly-tight combinatorial lower bound on output list size for list-recovering high-rate tensor codes. This bound implies in turn a nearly-tight lower bound of $n^{\Omega(1/\log\log n)}$ on the product of query complexity and output list size for locally list-recovering high-rate tensor codes.

7.1 Introduction

Over the years, many techniques have been devised for constructing new codes from old codes, where the new codes inherit desirable properties from the base codes. In this thesis, we aim to broaden our understanding of the effectiveness of these techniques. Prior work of mine has focused on the popular *tensoring* operation, which we present next.

Tensor codes. Given two linear codes $\mathcal{C} \leq \mathbb{F}_q^n$ and $\mathcal{C}' \leq \mathbb{F}_q^{n'}$, one can form their *tensor product* $\mathcal{C} \otimes \mathcal{C}' \leq \mathbb{F}_q^{n \otimes n'}$, which can be abstractly defined as the subspace spanned by the tensors $c \otimes c'$ for $c \in \mathcal{C}$ and $c' \in \mathcal{C}'$, or more concretely as the space of matrices whose columns lie in $\mathcal{C}$ and rows lie in $\mathcal{C}'$. One can easily show $\delta(\mathcal{C} \otimes \mathcal{C}') = \delta(\mathcal{C})\delta(\mathcal{C}')$ and $R(\mathcal{C} \otimes \mathcal{C}') = R(\mathcal{C})R(\mathcal{C}')$, and Gopalan, Guruswami and Raghavendra [GGR11] also gave a formula for its list-decoding radius. More recently, by studying $\mathcal{C}^{\otimes t}$,[1] Hemenway, Ron-Zewi and Wootters [HRZW17] showed how to obtain codes with near-linear time X-decoding, where X is a qualifier that takes values in an impressively large set. We defer a precise discussion of their results to Section 7.1.2,

Remark 7.1.1. While we typically like to reserve n to refer to the block length of a code, when we study tensor codes $\mathcal{C}^{\otimes t}$ it is unclear if n should refer to the block length of the base code $\mathcal{C}$ of the block length of the resulting code, which is n^t. In this chapter, we have made the choice that n should be the block length of $\mathcal{C}$ and N will be the block length of $\mathcal{C}^{\otimes t}$.

In this chapter, along with the familiar concepts of list-decoding and list-recovery we will study various notions of "local" decoding.[2] Indeed, a predominant reason for the interest in the tensoring operation is that the codes thus obtained tend to have interesting locality properties. Before proceeding to a discussion of prior work and our results, we provide a gentle introduction to these concepts. (The formal definitions are provided in Section 7.2.)

7.1.1 The Cast

Local decoding/correction. In *local decoding*, the goal is to uniquely decode in sublinear time. Since outputting the entire codeword already takes linear time, we need to relax our requirements. For a given $w \in \Sigma^n$ and a message coordinate $i \in [k]$, we are asked to recover the ith coordinate of the message underlying the unique codeword closest to w. As we want to run in sublinear time (and in particular query a sublinear number of the coordinates of w), we allow the algorithm to be randomized and have a small probability of error. *Local correction* is similar to local decoding, except now we are given a codeword coordinate $i \in [n]$ and expected to output the ith coordinate of the

[1] Here, for an integer $t \geq 1$, we define inductively $\mathcal{C}^{\otimes 1} = \mathcal{C}$ and $\mathcal{C}^{\otimes t} = \mathcal{C}^{\otimes (t-1)} \otimes \mathcal{C}$ for $t \geq 2$.

[2] Indeed, $X =$local is an allowed setting in [HRZW17].

closest codeword. Finally, in *approximate* local decoding (resp., local correction) one is only required to recover correctly most of the message (resp., codeword) coordinates.

Local list-decoding/recovery Local list-decoding combines the notions of local decoding and list-decoding. We are given some $w \in \Sigma^n$, and the goal is that for any nearby codeword, one can in sublinear time recover the i-th symbol of the message corresponding to the codeword for any $i \in [k]$. More precisely, the local list-decoding algorithm first does some preprocessing and then produces as output a collection of algorithms $\{A_1, \ldots, A_L\}$. For any nearby codeword c, with high probability one of these algorithms corresponds to it. These algorithms then behave like local decoding algorithms: on input $i \in [k]$, if the algorithm corresponded to a codeword c, then by making queries to only a sublinear number of coordinates the algorithm with high probability outputs the correct value of the ith symbol of the message underlying c.

The above definition of local list-decoding can be extended to local list-recovery in a straightforward way: now the algorithms A_j correspond to all codewords that agree with most of the input lists. As above, we can also define a local correction version of local list-decoding (or local list-recovery) where the algorithms A_j are required to recover codeword symbols as opposed to message symbols. Finally, we can also define approximate local list-decoding (or local list-recovery) where the algorithms A_j are only required to recover correctly most of the message coordinates (or codeword coordinates in the local correction version).

Remark 7.1.2. In this chapter, if we wish to emphasize that we are referring to a decoding problem in the typical sense (i.e., not in the local senses discussed above), we may add the qualifier *global*. E.g., global list-decoding is the standard notion of list-decoding encountered previously.

7.1.2 The Context

The starting point for this work is the recent result of [HRZW17] on high-rate list-recoverable tensor codes and its corollaries. The main technical result of [HRZW17] was that the tensor product of an efficient (poly-time) high-rate globally list-recoverable code is approximately locally list-recoverable (in either the local decoding or local correction sense). They then observed that the "approximately" modifier can be eliminated by pre-encoding the tensor product with a locally decodable code. This gave the first construction of codes with rate arbitrarily close to 1 that are locally list-recoverable from an $\Omega(1)$ fraction of errors, but only in the local decoding version. Finally, using the expander-based distance amplification method of [AEL95; AL96] (specialized to the setting of local list-recovery [GI02; Gop+18]), this gave the first capacity-achieving locally list-recoverable (and in particular, list-decodable) codes with sublinear (and in fact $N^{\tilde{O}(1/\log\log N)}$) query complexity and running time (once more, in the local decoding version).

The above result also yielded further consequences for global decoding. Specifically,

[HRZW17] observed that the approximate local list-recovery algorithm for tensor codes naturally gives a probabilistic near-linear time global list-recovery algorithm. Once more, using the expander-based distance amplification method of [AEL95; AL96; GI02], this gave the first capacity-achieving list-recoverable (and in particular, list-decodable) codes with probabilistic near-linear time global list-recovery algorithms. Finally, via the random concatenation method of [Tho83; GI04], this yielded in turn a (randomized) construction of constant-rate binary codes approaching the Gilbert-Varshamov bound with a probabilistic near-linear time algorithm for global unique decoding up to half the minimum distance.

One could potentially hope (following [Gop+18] which implemented a local version of [Tho83; GI04]) for an analogous result that would give constant-rate codes approaching the GV bound that are locally correctable (or locally decodable) with query complexity and running time $N^{o(1)}$. However, what prevented [HRZW17] from obtaining such a result was the fact that their capacity-achieving locally list-recoverable codes only worked in the local decoding version (i.e., they were only able to recover message coordinates).

7.1.3 Our Results

We revisit the techniques of [HRZW17] and show the following.

Deterministic near-linear time global list-recovery. The tensor product of an efficient (poly-time) high-rate globally list-recoverable code is globally list-recoverable in *deterministic* near-linear time. Plugging this into the machinery of [AEL95; AL96; GI02], we get the first capacity-achieving list-recoverable (and in particular, list-decodable) codes with *deterministic* near-linear time global list-recovery algorithms. Plugging this into the machinery of [Tho83; GI04], yields in turn constant-rate binary codes (with a randomized construction) approaching the GV bound with deterministic near-linear time global unique-decoding algorithms.

Our deterministic global list-recovery algorithm is obtained by derandomizing the random choices of the [HRZW17] algorithm using appropriate samplers.

Local list-recovery. An instantiation of the base code to produce tensor product codes which are themselves genuinely locally list-recoverable (i.e., not just approximately locally list-recoverable) in the local correction version. Once more, plugging this into the machinery of [AEL95; AL96; GI02], we get capacity-achieving locally list-recoverable codes, but now in the local correction version. This now plugs in turn into the machinery of [Tho83; GI04] to give constant-rate binary codes (with a randomized construction) approaching the GV bound that are locally decodable with query complexity and running time $N^{o(1)}$. This improves over prior work [Gop+18] that only gave query complexity N^{ε} with rate that is exponentially small in $1/\varepsilon$.

We obtain our result by taking the base code to be the intersection of an efficient (poly-time) high-rate globally list-recoverable code and a high-rate locally correctable code. Assuming both codes are linear, we have that the intersection is a high-rate code that is both. The result of [HRZW17] already guarantees that this tensor product is approximately locally list-recoverable (in the local correction version), and we use the fact that the tensor product of a locally correctable codes is also locally correctable [Vid15] to remove the "approximately'âĂŹ modifier.

Limitations on list-recoverability. We establish a combinatorial lower bound showing limitations on the list-recoverability of high-rate tensor codes. Specifically, we show that when the rate of the base code is high, every t-wise tensor product of this code has output list size doubly-exponential in t. This means that taking t to be more than $\log \log N$ leads to superpolynomial output list size, precluding the possibility of efficient list-recovery.

Instantiating this appropriately, one implication of this result is that there is a base code such that for every tensor power with block length N, the product of the query complexity and output list size for local list-recovery is at least $N^{\Omega(1/\log \log N)}$. We note that in contrast, it could be that for every base code, there is a tensor power with block length N for which local correction can be done with query complexity $O(1)$.

A key observation that we use is that a high-rate code has many codewords with pairwise disjoint supports. We combine this along with other linear-algebraic arguments to design a list-recovery instance for the tensor product of a high-rate code which has many codewords that are consistent with it.

Finally, we note that the recent work [Kop+18] has shown that high-rate *multiplicity codes* are also genuinely locally list-recoverable in the local correction version with $N^{o(1)}$ query complexity. However, these codes do not suffice for our GV bound application, as this application requires the codes to also be locally testable, and we do not currently know a local testing procedure for multiplicity codes. Moreover, we do not know how to derandomize the local list-recovery procedure for multiplicity codes.

Below we give precise statements of our results. For formal definitions of the various notions of decoding in the following theorem statements, see Section 7.2.

7.1.4 Deterministic Near-Linear Time Global List-Recovery

Our first main result shows that the tensor product of an efficient (poly-time) high-rate globally list-recoverable code is globally list-recoverable in *deterministic* near-linear time. In the theorem statement, one should think of all parameters δ, ρ, L, t, and consequently also s, as constants (or more generally, as slowly increasing/decreasing functions of n). In that case, the theorem says that if $\mathcal{C} \leq \mathbb{F}_q^n$ is (ρ, ℓ, L)-globally list-recoverable deterministically in time $T = \mathrm{poly}(n)$, then the t-iterated tensor product $\mathcal{C}^{\otimes t}$ of length $N = n^t$ is $(\Omega(\rho), \ell, L^{O(1)})$-globally list-recoverable deterministically in time $O(n^t \cdot T) =$

$n^{t+O(1)} = N^{1+O(1/t)}$.

Theorem 7.1.3 (Deterministic Near-Linear Time List-Recovery of High-Rate Tensor Codes)**.** *The following holds for any $\delta, \rho > 0$, and $s = \mathrm{poly}(1/\delta, 1/\rho)$. Suppose that $\mathcal{C} \leq \mathbb{F}_q^n$ is a linear code of relative distance δ that is (ρ, ℓ, L)-globally list-recoverable deterministically in time T. Then $\mathcal{C}^{\otimes t} \leq \mathbb{F}_q^{n^t}$ is $(\rho \cdot s^{-t^2}, \ell, L^{s^{t^3} \cdot L^t})$-globally list-recoverable deterministically in time $n^t \cdot T \cdot L^{s^{t^3} \cdot L^t}$.*

Applying the expander-based distance amplification method of [AEL95; AL96; GI02] on the codes given by the above theorem, we obtain the first capacity-achieving list-recoverable (and in particular, list-decodable) codes with *deterministic* near-linear time global list-recovery algorithms.

Corollary 7.1.4 (Deterministic Near-Linear Time Capacity-Achieving List-Recoverable Codes)**.** *For any constants $R \in [0, 1]$, $\varepsilon > 0$, and $\ell \geq 1$ there exists an infinite family of codes $\{\mathcal{C}_N\}_N$, where $\mathcal{C}_N$ has block length N, alphabet size $N^{o(1)}$, rate R, and is $(1 - R - \varepsilon, \ell, N^{o(1)})$-globally list recoverable deterministically in time $N^{1+o(1)}$.*

Remark 7.1.5. The precise $o(1)$ quantities are a bit difficult to determine. However, an inspection of the proof shows that the list size grows slower than $\log^{(c)}(n)$ for any constant c. (Recall $\log^{(c)} n$ is the c-th iterated logarithm, i.e., $\underbrace{\log \circ \cdots \circ \log}_{c \text{ times}} n$.) Furthermore the running time is roughly $N^{1+O(1/\log\log\log N)}$.

Applying the random concatenation method of [Tho83; GI04], the above corollary yields in turn constant-rate codes approaching the Gilbert-Varshamov bound with *deterministic* near-linear time global unique decoding algorithms.

Corollary 7.1.6 (Deterministic Near-Linear Time Unique-Decoding up to the GV Bound)**.** *For any constants $R \in [0, 0.02]$ and $\varepsilon > 0$ there exists an infinite family of binary linear codes $\{\mathcal{C}_N\}_N$, where $\mathcal{C}_N$ has block length N and rate R, and is globally uniquely-decodable deterministically from $\frac{h_2^{-1}(1-R)-\varepsilon}{2}$-fraction of errors in time $N^{1+o(1)}$.*

Remark 7.1.7. Again, inspecting the proof shows the running time is roughly $N^{1+O(1/\log\log\log N)}$.

7.1.5 Local List-Recovery

Our second main result shows that if the base code is *both* globally list-recoverable and locally correctable, then the tensor product is (genuinely) locally list-recoverable (in the local correction version).

Theorem 7.1.8 (Local List-Recovery of High-Rate Tensor Codes)**.** *The following holds for any $\delta, \rho > 0$, and $s = \mathrm{poly}(1/\delta, 1/\rho)$. Suppose that $\mathcal{C} \leq \mathbb{F}_q^n$ is a linear code of relative distance δ that is (ρ, ℓ, L)-globally list-recoverable, and locally correctable from $(\delta/2)$-fraction of errors with query complexity Q, and $t \geq 3$. Then $\mathcal{C}^{\otimes t} \leq \mathbb{F}_q^{n^t}$ is $(\rho \cdot s^{-t^3}, \ell, L^{s^{t^3} \cdot \log^t L})$-locally list-recoverable with query complexity $n^{O(1)} \cdot Q^{O(t)} \cdot L^{s^{t^3} \cdot \log^t L}$.*

Once more, applying the expander-based distance amplification method of [AEL95; AL96; GI02; Gop+18], as well as the random concatenation method of [Tho83; GI04;

Gop+18], the above theorem yields constant-rate codes approaching the Gilbert-Varshamov bound that are *locally correctable* with query complexity $N^{o(1)}$.

Corollary 7.1.9 (Local Correction up to the GV Bound)**.** *For any constants $R \in [0, 0.02]$ and $\varepsilon > 0$ there exists an infinite family of binary linear codes $\{\mathcal{C}_N\}_N$, where $\mathcal{C}_N$ has block length N and rate R, and is locally correctable from $\frac{h_2^{-1}(1-R)-\varepsilon}{2}$-fraction of errors with query complexity $N^{o(1)}$.*

Remark 7.1.10. An inspection of the proof reveals that the query complexity is roughly $N^{O(1/\log\log N)}$.

7.1.6 Combinatorial Lower Bound on Output List Size

Our final main result shows a nearly-tight combinatorial lower bound on output list size for list-recovering high-rate tensor codes.

Theorem 7.1.11 (Output List Size for List-Recovering High-Rate Tensor Codes)**.** *Let $\varepsilon > 0$. Suppose that $\mathcal{C} \leq \mathbb{F}_q^n$ is a linear code of rate $1 - \varepsilon$, and that $\mathcal{C}^{\otimes t} \leq \mathbb{F}_q^{n^t}$ is $(0, \ell, L)$-list-recoverable. Then $L \geq \ell^{1/\varepsilon^t}$.*

The above bound can be instantiated concretely as follows.

Corollary 7.1.12. *For any $\delta > 0$ and $\ell > 1$ there exists $L > 1$ such that the following holds for any sufficiently large n. There exists a linear code $\mathcal{C} \leq \mathbb{F}_q^n$ of relative distance δ that is $(\Omega(\delta), \ell, L)$-list-recoverable, but $\mathcal{C}^{\otimes t} \leq \mathbb{F}_q^{n^t}$ is only $(0, \ell, L')$-list-recoverable for $L' \geq \exp((2\delta)^{-(t-3/2)} \cdot \sqrt{\log L})$.*

Finally, we also obtain a nearly-tight lower bound of $N^{\Omega(1/\log\log N)}$ on the product of query complexity and output list size for locally list-recovering high-rate tensor codes.

Corollary 7.1.13. *For any $\delta > 0$ and sufficiently large n there exists a linear code $\mathcal{C} \leq \mathbb{F}_q^n$ of relative distance δ such that the following holds. Suppose that $\mathcal{C}^{\otimes t} \leq \mathbb{F}_q^N$ is $(\frac{1}{N}, 2, L)$-locally list-recoverable with query complexity Q. Then $Q \cdot L \geq N^{\Omega_\delta(1/\log\log N)}$.*

7.2 Preliminaries

Many of the coding theoretic concepts we will need for this chapter were introduced in Chapter 2. The notable exceptions are the various notions of local decoding we investigate. We also provide important facts concerning tensor codes in Section 7.2.2.

Remark 7.2.1. In this chapter, many of our results are completely agnostic to the base field; indeed, some results apply even if the field is infinite. For this reason, we just denote the field by $\mathbb{F}$. As usual, an arbitrary alphabet that need not be a field is denoted by Σ.

7.2.1 Local Codes

Intuitively, a code $\mathcal{C}$ is said to be *locally testable* [FS95; RS96; GS06] if, given a string $w \in \Sigma^n$, it is possible to determine whether w is a codeword of $\mathcal{C}$, or rather far from $\mathcal{C}$, by reading only a small part of w. For our purposes, we will also require an additional *tolerance* property of determining whether w is sufficiently close to the code.

Definition 7.2.2 (Tolerant Locally Testable Code (Tolerant LTC))**.** We say that a code $\mathcal{C} \subseteq \Sigma^n$ is (Q, ρ, σ)-*tolerantly locally testable* if there exists a randomized algorithm A that satisfies the following requirements:

- **Input:** A gets oracle access to a string $w \in \Sigma^n$.
- **Query complexity:** A makes at most Q queries to the oracle w.
- **Completeness:** If $d(w, \mathcal{C}) \leq \rho$, then A accepts with probability at least $\frac{2}{3}$.
- **Soundness:** If $d(w, \mathcal{C}) \geq \sigma$, then A rejects with probability at least $\frac{2}{3}$.

Remark 7.2.3. The definition requires $0 \leq \rho < \sigma \leq 1$. The above success probability of $\frac{2}{3}$ can be amplified using sequential repetition, at the cost of increasing the query complexity. Specifically, amplifying the success probability to $1 - \exp(-t)$ requires increasing the query complexity by a multiplicative factor of $O(t)$.

Next, we introduce locally correctable codes. Intuitively, a code is said to be *locally correctable* [Bab+91; STV01; KT00] if, given a codeword $c \in \mathcal{C}$ that has been corrupted by some errors, it is possible to decode any coordinate of c by reading only a small part of the corrupted version of c.

Definition 7.2.4 (Locally Correctable Code (LCC))**.** We say that a code $\mathcal{C} \subseteq \Sigma^n$ is (Q, ρ)-*locally correctable* if there exists a randomized algorithm A that satisfies the following requirements:

- **Input:** A takes as input a coordinate $i \in [n]$, and also gets oracle access to a string $w \in \Sigma^n$ that is ρ-close to a codeword $c \in \mathcal{C}$.
- **Query complexity:** A makes at most Q queries to the oracle w.
- **Output:** A outputs c_i with probability at least $\frac{2}{3}$.

Remark 7.2.5. The definition requires $\rho < \delta(\mathcal{C})/2$, as otherwise it is not guaranteed that the codeword closest to w is unique. The above success probability of $\frac{2}{3}$ can be amplified using sequential repetition, at the cost of increasing the query complexity. Specifically, amplifying the success probability to $1 - \exp(-t)$ requires increasing the query complexity by a multiplicative factor of $O(t)$.

The following definition from [GL89; STV01; Gop+18] generalizes the notion of locally correctable codes to the setting of list-decoding/recovery. In this setting, the local list-recovery algorithm is required to output in an implicit sense all codewords that are consistent with most of the input lists.

Definition 7.2.6 (Locally List Recoverable Code)**.** We say that a code $\mathcal{C} \subseteq \Sigma^n$ is (Q, ρ, η, ℓ, L)-*locally list-recoverable* if there exists a randomized algorithm A that satisfies the following requirements:

- **Input:** A gets oracle access to a string $S \in \binom{\Sigma}{\leq \ell}^n$.
- **Query complexity:** A makes at most Q queries to the oracle S.
- **Output:** A outputs L randomized algorithms $A_1, \ldots, A_L$, where each A_j takes as input a coordinate $i \in [n]$, makes at most Q queries to the oracle S, and outputs a symbol in Σ.
- **Completeness:** For any codeword $c \in \mathcal{C}$ which satisfies $d(c, S) \leq \rho$, with probability at least $1 - \eta$ over the randomness of A, the following event happens: there exists some $j \in [L]$ such that for all $i \in [n]$,

$$\mathbb{P}\left(A_j(i) = c_i\right) \geq \frac{2}{3}\,, \tag{7.1}$$

  where the probability is over the internal randomness of A_j.
- **Soundness:** With probability at least $1-\eta$ over the randomness of A, the following event happens: for every $j \in [L]$, there exists some $c \in \mathcal{C}$ such that for all $i \in [n]$,

$$\mathbb{P}\left(A_j(i) = c_i\right) \geq \frac{2}{3}\,,$$

  where the probability is over the internal randomness of A_j.

We say that A has *preprocessing time* T_{pre} if A outputs the description of the algorithms $A_1, \ldots, A_L$ in time at most T_{pre}, and has *running time* T if each A_j has running time at most T. Finally, we say that the code $\mathcal{C}$ is (Q, ρ, η, L)*-locally list-decodable* if it is $(Q, \rho, \eta, 1, L)$-locally list-recoverable.

Remark 7.2.7. The above definition of locally list-recoverable code defers from that given in [HRZW17, Definition 4.5] in two ways. First, our definition requires that the local algorithms $A_1, \ldots, A_L$ in the output list of A locally decode codeword coordinates as opposed to just message coordinates. Second, following [Gop+18], we require an additional soundness property that guarantees that with high probability, each local algorithm in the output list locally decodes a true codeword. These two requirements will be crucial for our GV bound local correction application (Corollary 7.1.9).

7.2.2 Tensor Codes

In this chapter we study the list-recovery properties of high-rate tensor product codes, defined as follows.

Definition 7.2.8 (Tensor product codes)**.** Let $\mathcal{C}_1 \leq \mathbb{F}^{n_1}$, $\mathcal{C}_2 \leq \mathbb{F}^{n_2}$ be linear codes. Their *tensor product code* $\mathcal{C}_1 \otimes \mathcal{C}_2 \leq \mathbb{F}^{n_1 \times n_2}$ consists of all matrices $M \in \mathbb{F}^{n_1 \times n_2}$ such that all the columns of M are codewords of $\mathcal{C}_1$ and all the rows are codewords of $\mathcal{C}_2$.

What follows are some well-known facts about the tensor product operation, including its effect on the classical parameters of a code.

Fact 7.2.9. *Suppose that $\mathcal{C}_1 \leq \mathbb{F}^{n_1}$, $\mathcal{C}_2 \leq \mathbb{F}^{n_2}$ are linear codes of rates R_1, R_2 and relative distances δ_1, δ_2 respectively. Then the tensor product code $\mathcal{C}_1 \otimes \mathcal{C}_2 \leq \mathbb{F}^{n_1 \times n_2}$ is a linear code of rate $R_1 \cdot R_2$ and relative distance $\delta_1 \cdot \delta_2$.*

Moreover, if $\mathcal{C}_1$, $\mathcal{C}_2$ are encodable in times T_1, T_2, respectively, then $\mathcal{C}_1 \otimes \mathcal{C}_2$ is encodable in time $n_1 T_2 + n_2 T_1$, and if $\mathcal{C}_1$, $\mathcal{C}_2$ are decodable from ρ_1, ρ_2-fraction of errors in times T_1, T_2, respectively, then $\mathcal{C}_1 \otimes \mathcal{C}_2$ is decodable from $(\rho_1 \cdot \rho_2)$-fraction of errors in time $n_1 T_2 + n_2 T_1$.

For a linear code $\mathcal{C}$, define inductively $\mathcal{C}^{\otimes 1} := C$ and $\mathcal{C}^{\otimes t} := \mathcal{C} \otimes \mathcal{C}^{\otimes (t-1)}$. By induction on t we have the following.

Corollary 7.2.10. *Suppose that $\mathcal{C} \leq \mathbb{F}^n$ is a linear code of rate R and relative distance δ. Then the tensor product code $\mathcal{C}^{\otimes t} \leq \mathbb{F}^{n^t}$ is a linear code of rate R^t and relative distance δ^t.*

Moreover, if $\mathcal{C}$ is encodable in time T then $\mathcal{C}^{\otimes t}$ is encodable in time $t \cdot n^{t-1} \cdot T$, and if $\mathcal{C}^{\otimes t}$ is decodable from ρ-fraction of errors in time T then $\mathcal{C}^{\otimes t}$ is decodable from ρ^t-fraction of errors in time $t \cdot n^{t-1} \cdot T$.

For a pair of matrices $G_1 \in \mathbb{F}^{n_1 \times k_1}$ and $G_2 \in \mathbb{F}^{n_2 \times k_2}$, their *tensor product* $G_1 \otimes G_2$ is the $(n_1 \cdot n_2) \times (k_1 \cdot k_2)$-matrix over $\mathbb{F}$ with entries

$$(G_1 \otimes G_2)_{(i_1,i_2),(j_1,j_2)} = (G_1)_{i_1,j_1} \cdot (G_2)_{i_2,j_2}$$

for every $i_1 \in [n_1]$, $i_2 \in [n_2]$, $j_1 \in [k_1]$, and $j_2 \in [k_2]$.

Fact 7.2.11. *Suppose that G_1, G_2 are generating matrices of linear codes $\mathcal{C}_1 \leq \mathbb{F}^{n_1}$, $\mathcal{C}_2 \leq \mathbb{F}^{n_2}$, respectively. Then the tensor product $G_1 \otimes G_2$ is a generating matrix of $\mathcal{C}_1 \otimes \mathcal{C}_2$.*

7.3 Deterministic Near-Linear Time Global List-Recovery

In this section we prove Theorem 7.1.3, restated bellow, which shows that the tensor product of an efficient (poly-time) high-rate globally list-recoverable code is globally list-recoverable in *deterministic* near-linear time.

Theorem 7.1.3 (Deterministic Near-Linear Time List-Recovery of High-Rate Tensor Codes)**.** *The following holds for any $\delta, \rho > 0$, and $s = \mathrm{poly}(1/\delta, 1/\rho)$. Suppose that $\mathcal{C} \leq \mathbb{F}_q^n$ is a linear code of relative distance δ that is (ρ, ℓ, L)-globally list-recoverable deterministically in time T. Then $\mathcal{C}^{\otimes t} \leq \mathbb{F}_q^{n^t}$ is $(\rho \cdot s^{-t^2}, \ell, L^{s^{t^3} \cdot L^t})$-globally list-recoverable deterministically in time $n^t \cdot T \cdot L^{s^{t^3} \cdot L^t}$.*

Theorem 7.1.3 follows by applying the lemma below iteratively.

Lemma 7.3.1. *The following holds for any $\delta, \rho, \delta_{\mathsf{dec}}, \delta'_{\mathsf{dec}} > 0$, and $\bar{s} = \mathrm{poly}(1/\delta, 1/\rho, 1/\delta_{\mathsf{dec}}, 1/\delta'_{\mathsf{dec}})$.*

Suppose that $\mathcal{C} \leq \mathbb{F}^n$ is a linear code of relative distance δ that is (ρ, ℓ, L)-globally list-recoverable deterministically in time T, and $\mathcal{C}' \leq \mathbb{F}^{n'}$ is a linear code that is (ρ', ℓ, L')-globally list-recoverable deterministically in time T'. Suppose furthermore that $\mathcal{C}, \mathcal{C}'$ are uniquely decodable deterministically from $\delta_{\mathsf{dec}}, \delta'_{\mathsf{dec}}$-fraction of errors in times $T_{\mathsf{dec}}, T'_{\mathsf{dec}}$, respectively.

Then $\mathcal{C} \otimes \mathcal{C}' \leq \mathbb{F}^{n \times n'}$ is $(\rho'/\bar{s}, \ell, (L')^{\bar{s} \cdot L/(\rho')^2})$-globally list-recoverable deterministically in time

$$(L')^{\bar{s} \cdot L/(\rho')^2} \cdot n \cdot (n' \cdot (T + T_{\mathsf{dec}}) + n \cdot T'_{\mathsf{dec}} + T') \ .$$

Before we prove the above lemma, we first show how it implies Theorem 7.1.3.

Proof of Theorem 7.1.3. We start with the code $\mathcal{C}$, and iteratively tensor with a new copy of $\mathcal{C}$ $t-1$ times. Specifically, we initially set $\mathcal{C}' := \mathcal{C}$, and at each step we apply Lemma 7.3.1 with the code $\mathcal{C}'$ being the code constructed so far, and the code $\mathcal{C}$ being a new copy of $\mathcal{C}$.

On each iteration, we can set in Lemma 7.3.1 $\delta_{\mathsf{dec}} := \min\{\rho, \delta/2\}$ and $T_{\mathsf{dec}} := T$ since the code $\mathcal{C}$ can be uniquely decoded from δ_{dec}-fraction of errors by running the list-recovery algorithm for $\mathcal{C}$ on the received word, and returning the codeword from the output list that is closest to the received word. Moreover, by Corollary 7.2.10, on the i-th iteration we can set $\delta'_{\mathsf{dec}} := \delta^t_{\mathsf{dec}}$ and $T'_{\mathsf{dec}} := i \cdot n^{i-1} \cdot T$. We conclude that on each iteration we can apply Lemma 7.3.1 with $\bar{s} := s^t$ for $s = \mathrm{poly}(1/\delta, 1/\rho)$.

In the above setting of parameters, we have that the list-recovery radius of $\mathcal{C}^{\otimes t}$ is at least $\tilde{\rho} := \rho/\bar{s}^t = \rho/s^{t^2}$, and that the output list size is at most $\tilde{L} := L^{\bar{s}^t \cdot L^t/\tilde{\rho}^{2t}} \leq L^{s^{O(t^3)} \cdot L^t}$.

Finally, on the i-th iteration the running time is increased by an additive factor of $n' \cdot (T + T_{\mathsf{dec}}) + n \cdot T'_{\mathsf{dec}} = O(i \cdot n^i \cdot T)$, and then by a multiplicative factor of at most $\tilde{L} \cdot n$, yielding a total running time of at most

$$\sum_{i=1}^{t-1} O(i \cdot n^i \cdot T) \cdot \left(\tilde{L} \cdot n\right)^{t-i} \cdot T \leq L^{s^{O(t^3)} \cdot L^t} \cdot n^t \cdot T\,.$$

So the desired conclusion holds by slightly enlarging the size of the polynomial s. □

We now proceed to the proof of Lemma 7.3.1. Our plan is to derandomize the approximate local list-recovery algorithm for the high-rate tensor codes of [HRZW17]. Recall that an approximate local list-recovery algorithm (local correction version) is a randomized algorithm A that outputs a collection of (without loss of generality, deterministic) local algorithms A_j satisfying the following: for any codeword c that is consistent with most of the input lists, with high probability (over the randomness of A) one of the local algorithms A_j locally corrects most of the coordinates of c.

As observed in [HRZW17], an approximate local list-recovery algorithm naturally gives a *probabilistic* near-linear time *global* list-recovery algorithm as follows. First run the algorithm A to obtain the collection of local algorithms A_j. Then for each A_j, output a codeword that is obtained by applying A_j on each codeword coordinate, and then uniquely decoding the resulting word to the closest codeword. The guarantee now is that any codeword that is consistent with most of the input lists will be output with high probability.

To derandomize the probabilistic global algorithm described above, we note that the preprocessing algorithm A in [HRZW17] produces the collection of local algorithms A_j by choosing a random subset of rows in the tensor product,[3] that is chosen uniformly at random amongst all subsets of the appropriate size. We then observe that this subset

[3] In [HRZW17], the role of columns and rows is swapped.

can be alternatively chosen using a randomness-efficient *sampler* without significantly hampering the performance. Finally, since the sampler uses a small amount of randomness (logarithmic in the blocklength of $\mathcal{C}$), we can afford to iterate over all seeds and return the union of all output lists. This gives a *deterministic* near-linear time global list-recovery algorithm that outputs all codewords that are consistent with most of the input lists.

7.3.1 Samplers

We start by defining the appropriate samplers we use.

Definition 7.3.2 ((Averaging) Sampler)**.** An (n, η, γ)*-sampler* with *randomness* r and *sample size* m is a randomized algorithm that tosses r random coins and outputs a subset $I \subseteq [n]$ of size m such that the following holds. For any function $f : [n] \to [0, 1]$, with probability at least $1 - \eta$ over the choice of I,

$$\left| \mathop{\mathbb{E}}_{\boldsymbol{i} \in I} [f(\boldsymbol{i})] - \mathop{\mathbb{E}}_{\boldsymbol{i} \in [n]} [f(\boldsymbol{i})] \right| \leq \gamma \, .$$

We shall use the following construction from Goldreich [Gol11].

Theorem 7.3.3 ([Gol11, Corollary 5.6])**.** *For any $\eta, \gamma > 0$ and integer n, the thare exists an (n, η, γ)-sampler with randomness $\log(n/\gamma)$, sample size $O\left(1/(\eta\gamma^2)\right)$, and running time $\mathrm{poly}(\log n, 1/\eta, 1/\gamma)$.*

In what follows, let Γ denote the (n, η, γ)-sampler promised by the above theorem, where we set $\eta := \frac{0.1}{L} \cdot \frac{\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{3}$ and $\gamma := \rho' \cdot \frac{\delta \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{24}$. Let $r := \log(n/\gamma) \leq \log(n \cdot \bar{s}/\rho')$ and $m := O(1/(\eta\gamma^2)) \leq \bar{L} \cdot \bar{s}/(\rho')^2$ denote the randomness and sample size of Γ, respectively (assuming that $\bar{s}$ is a sufficiently large polynomial).

7.3.2 Randomness-Efficient Algorithm

We first describe a randomness-efficient global list-recovery algorithm $\tilde{A}$ for $\mathcal{C} \otimes \mathcal{C}'$ that is obtained by replacing the choice of a uniform random subset of rows made in [HRZW17] with a sample from Γ. We will later observe that the randomness can be eliminated by iterating over all seeds of Γ and returning the union of all output lists.

The algorithm $\tilde{A}$ behaves as follows. First, it uses Γ to sample a subset of m rows $I = \{i_1, \ldots, i_m\} \subseteq [n]$. Then for $k = 1, \ldots, m$, it runs the list-recovery algorithm A' for $\mathcal{C}'$ on the i_k-th row $S|_{\{i_k\} \times [n']}$; let $\mathcal{L}'_{i_1}, \mathcal{L}'_{i_2}, \ldots, \mathcal{L}'_{i_m} \subseteq \mathcal{C}'$ denote the lists output by A' on each of the rows in I. Finally, for any choice of codewords $c'_1 \in \mathcal{L}'_{i_1}, c'_2 \in \mathcal{L}'_{i_2}, \ldots, c'_m \in \mathcal{L}'_{i_m}$, the algorithm $\tilde{A}$ outputs a codeword $\tilde{c} \in \mathcal{C} \otimes \mathcal{C}'$ that is obtained as follows.

For each column $j \in [n']$, the algorithm $\tilde{A}$ runs the list-recovery algorithm A for $\mathcal{C}$ on the j-th column $S|_{[n] \times \{j\}}$; let $\mathcal{L}_1, \mathcal{L}_2, \ldots \mathcal{L}_{n'} \subseteq \mathcal{C}$ denote the lists output by A on each of the n' columns. Then the algorithm $\tilde{A}$ chooses for each column $j \in [n']$ the codeword

$c_j \in \mathcal{L}_j$ whose restriction to I is closest to $((c'_1)_j, (c'_2)_j \ldots, (c'_m)_j)$ (i.e., the restriction of $c'_1, c'_2, \ldots, c'_m$ to the j-th column). Finally, the algorithm $\tilde{A}$ sets the value of each column $j \in [n']$ to c_j, and uniquely decodes the resulting word $\tilde{c}_0$ to the nearest codeword $\tilde{c} \in \mathcal{C} \otimes \mathcal{C}'$, assuming there is one at distance at most $\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}$. If $d(\tilde{c}, S) \leq \rho'/\bar{s}$, then $\tilde{A}$ includes $\tilde{c}$ in the output list $\tilde{\mathcal{L}}$. The formal description is given in Algorithm 1.

Algorithm 1 The randomness-efficient global list-recovery algorithm $\tilde{A}$ for $\mathcal{C} \otimes \mathcal{C}'$.

function $\tilde{A}(S \in \binom{\mathbb{F}}{\leq \ell}^{n \times n'})$
  Sample $I = \{i_1, \ldots, i_m\} \subseteq [n]$ of size m using sampler Γ.
  for $k = 1, \ldots, m$ **do**
    Run the list-recovery algorithm A' for $\mathcal{C}'$ on the i_k-th row $S|_{\{i_k\} \times [n']}$, and let $\mathcal{L}'_{i_k} \subseteq \mathcal{C}'$ be the list of codewords output by A'.
  end for
  Initialize $\tilde{c}_0 \in \mathbb{F}^{n \times n'}$, $\tilde{\mathcal{L}} \leftarrow \emptyset$.
  for any choice of codewords $c'_1 \in \mathcal{L}'_{i_1}, c'_2 \in \mathcal{L}'_{i_2}, \ldots, c'_m \in \mathcal{L}'_{i_m}$ **do**
    for $j \in [n']$ **do**
      Run the list-recovery algorithm A for $\mathcal{C}$ on the j-th column $S|_{[n] \times \{j\}}$, and let $\mathcal{L}_j \subseteq \mathcal{C}$ be the list of codewords output by A.
      Choose a codeword $c_j \in \mathcal{L}_j$ for which $c_j|_I$ is closest to $((c'_1)_j, (c'_2)_j, \ldots, (c'_m)_j)$ (breaking ties arbitrarily).
      Set the j-th column of $\tilde{c}_0$ to c_j.
    end for
    Uniquely decode $\tilde{c}_0$ from $(\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}})$-fraction of errors, and let $\tilde{c} \in C \otimes C'$ be the resulting codeword (if it exists). If $d(\tilde{c}, S) \leq \rho'/\bar{s}$, add $\tilde{c}$ to $\tilde{\mathcal{L}}$.
  end for
end function

7.3.3 Output List Size, Randomness, and Running Time

The output list size is at most the number of choices of $c'_1 \in \mathcal{L}'_1, c'_2 \in \mathcal{L}'_2, \ldots, c'_m \in \mathcal{L}'_m$ which is $(L')^m \leq (L')^{L \cdot \bar{s}/(\rho')^2}$, and the randomness is $r \leq \log(n \cdot \bar{s}/\rho')$.

As to running time, the algorithm $\tilde{A}$ invokes the sampler Γ, followed by m invocations of the list-recovery algorithm A' for $\mathcal{C}'$, and $(L')^m \cdot n'$ invocations of the list-recovery algorithm A for $\mathcal{C}$. Finally, it invokes $(L')^m$ times the unique decoding algorithm for $\mathcal{C} \otimes \mathcal{C}'$ which can be implemented to run in time $n \cdot T'_{\mathsf{dec}} + n' \cdot T_{\mathsf{dec}}$ by Fact 7.2.9. Thus the total running time is at most

$$\begin{aligned} &\mathrm{poly}(\log n, m) + m \cdot T' + (L')^m \cdot n' \cdot T + (L')^m \cdot (n \cdot T'_{\mathsf{dec}} + n' \cdot T_{\mathsf{dec}}) \\ &\qquad \leq (L')^{\bar{s} \cdot L/(\rho')^2} \cdot (n' \cdot (T + T_{\mathsf{dec}}) + n \cdot T'_{\mathsf{dec}} + T') \ , \end{aligned}$$

where the inequality holds for a sufficiently large polynomial $\bar{s}$.

Correctness

Next we establish the following.

Claim 7.3.4. *Suppose that $\tilde{c} \in \mathcal{C} \otimes \mathcal{C}'$ has $d(\tilde{c}, S) \leq \rho'/\bar{s}$. Then with probability at least $2/3$, the codeword $\tilde{c}$ is included in $\tilde{\mathcal{L}}$.*

Note that the above claim in particular implies that there are at most $O((L')^m)$ codewords $\tilde{c} \in \mathcal{C} \otimes \mathcal{C}'$ with $d(\tilde{c}, S) \leq \rho'/\bar{s}$. To prove Claim 7.3.4, it is enough to show that with probability at least $2/3$ over the choice of $I = \{i_1, \ldots, i_m\}$, there exists a choice of $c'_1 \in \mathcal{L}'_{i_1}, c'_2 \in \mathcal{L}'_{i_2}, \ldots, c'_m \in \mathcal{L}'_{i_m}$ such that at the iteration corresponding to $c'_1, c'_2, \ldots, c'_m$, the word $\tilde{c}_0$ satisfies $d(\tilde{c}_0, \tilde{c}) \leq \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}$. Once we establish this, the unique-decoding algorithm for $\mathcal{C} \otimes \mathcal{C}'$ will successfully decode $\tilde{c}$ from $\tilde{c}_0$.

For a row $i \in [n]$, let $\hat{c}_i$ be the codeword in $\mathcal{L}'_i$ that is closest to the i-th row of $\tilde{c}$ (breaking ties arbitrarily), that is, the codeword $\hat{c}_i \in \mathcal{L}'_i$ for which $d(\hat{c}_i, \tilde{c}|_{\{i\} \times [n']})$ is minimal. We will show that with probability at least $2/3$ over the choice of $I = \{i_1, \ldots, i_m\}$, at the iteration corresponding to the choice of $\hat{c}_{i_1} \in \mathcal{L}'_{i_1}, \hat{c}_{i_2} \in \mathcal{L}'_{i_2}, \ldots, \hat{c}_{i_m} \in \mathcal{L}'_{i_m}$, the word $\tilde{c}_0$ will satisfy that $d(\tilde{c}_0, \tilde{c}) \leq \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}$.

Following [HRZW17], to establish the above, we show that with high probability over the choice of I, a large fraction of the columns $j \in [n']$ are "good", in the sense that $\tilde{c}_0$ and $\tilde{c}$ agree on all of these columns in the iteration corresponding to the choice of $\hat{c}_{i_1} \in \mathcal{L}'_{i_1}, \hat{c}_{i_2} \in \mathcal{L}'_{i_2}, \ldots, \hat{c}_{i_m} \in \mathcal{L}'_{i_m}$. In what follows, let $\tilde{c}_1, \tilde{c}_2 \ldots, \tilde{c}_{n'}$ denote the columns of $\tilde{c}$.

Definition 7.3.5 (Good Column)**.** Let $I = \{i_1, \ldots, i_m\} \subseteq [n]$ be a subset of m rows. We say that a column $j \in [n']$ is *good with respect to I* if it satisfies the following properties:

1. The codeword $\tilde{c}$ is consistent with all but a ρ-fraction of the input lists on column j, that is, $d(\tilde{c}_j, S|_{[n] \times \{j\}}) \leq \rho$.
2. Let $\mathcal{L}_j$ denote the list of all codewords in $\mathcal{C}$ that are consistent with all but a ρ-fraction of the input lists on column j. Then for any $c \in \mathcal{L}_j \setminus \{\tilde{c}_j\}$ it holds that $d(c|_I, \tilde{c}_j|_I) > \delta/2$.
3. $d\big(\tilde{c}_j|_I, ((\hat{c}_{i_1})_j, \ldots, (\hat{c}_{i_m})_j)\big) \leq \delta/4$.

Claim 7.3.6 below shows that at the iteration corresponding to the choice of $\hat{c}_{i_1} \in \mathcal{L}'_{i_1}, \hat{c}_{i_2} \in \mathcal{L}'_{i_2}, \ldots, \hat{c}_{i_m} \in \mathcal{L}'_{i_m}$, $\tilde{c}_0$ and $\tilde{c}$ agree on all of the good columns. Claim 7.3.7 complements this by showing that with probability at least $2/3$ over the choice of I, at least a $(1 - \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}})$-fraction of the columns are good with respect to I. The combination of these claims yields the desired conclusion.

Claim 7.3.6. *Let $I = \{i_1, \ldots, i_m\} \subseteq [n]$ be a subset of m rows, and suppose that a column $j \in [n']$ is good with respect to I. Then at the iteration corresponding to the choice of $\hat{c}_{i_1} \in \mathcal{L}'_{i_1}, \hat{c}_{i_2} \in \mathcal{L}'_{i_2}, \ldots, \hat{c}_{i_m} \in \mathcal{L}'_{i_m}$ it holds that $\tilde{c}_0|_{[n] \times \{j\}} = \tilde{c}_j$.*

Proof. By Item 1 in the definition of a good column, $\tilde{c}$ is consistent with all but a ρ-fraction of the input lists on column j, and so $\tilde{c}_j \in \mathcal{L}_j$. By Item 3,

$$d\big(\tilde{c}_j|_I, ((\hat{c}_{i_1})_j, \ldots, (\hat{c}_{i_m})_j)\big) \leq \delta/4.$$

On the other hand, by Item 2 for any other codeword $c \in \mathcal{L}_j$ we have that

$$d\big(c|_I, ((\hat{c}_{i_1})_j, \ldots, (\hat{c}_{i_m})_j)\big) \geq d\big(\tilde{c}_j|_I, c|_I\big) - d\big(\tilde{c}_j|_I, ((\hat{c}_{i_1})_j, \ldots, (\hat{c}_{i_m})_j)\big) > \delta/4.$$

Thus, $\tilde{c}_j$ is the codeword in $\mathcal{L}_j$ whose restriction to I is closest to $((\hat{c}_{i_1})_j, \ldots, (\hat{c}_{i_m})_j)$, and so the algorithm $\tilde{A}$ will set $c_j := \tilde{c}_j$ at the iteration corresponding to the choice of $\hat{c}_{i_1} \in \mathcal{L}'_{i_1}, \hat{c}_{i_2} \in \mathcal{L}'_{i_2}, \ldots, \hat{c}_{i_m} \in \mathcal{L}'_{i_m}$. Consequently, the j-th column of $\tilde{c}_0$ will be set to the j-th column of $\tilde{c}$. □

Claim 7.3.7. *With probability at least $2/3$ over the choice of I, at least a $(1 - \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}})$-fraction of the columns are good with respect to I.*

For the proof of the above claim we shall also use the notion of a "good row".

Definition 7.3.8 (Good Row)**.** A row $i \in [n]$ is *good* if the codeword $\tilde{c}$ is consistent with all but a ρ'-fraction of the input lists row i, that is, $d(\tilde{c}|_{\{i\}\times[n']}, S|_{\{i\}\times[n']}) \leq \rho'$.

We claim that with high probability over the choice of I, a large fraction of the rows in I are good.

Claim 7.3.9. *With probability at least 0.9 over the choice of I, at least a $\left(1 - \frac{\delta \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{12}\right)$-fraction of the rows in I are good.*

Proof of Claim 7.3.9. For $i \in [n]$, let $f(i) := d(\tilde{c}|_{\{i\}\times[n']}, S|_{\{i\}\times[n']})$, and note that by the sampling property of Γ, with probability at least 0.9 over the choice of I we have that

$$\mathop{\mathbb{E}}_{\boldsymbol{i}\in I}[f(\boldsymbol{i})] \leq \mathop{\mathbb{E}}_{\boldsymbol{i}\in[n]}[f(\boldsymbol{i})] + \gamma = d(\tilde{c}, S) + \gamma \leq \rho' \cdot \frac{\delta \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{12},$$

where the last inequality holds by assumption that $\gamma = \rho' \cdot \frac{\delta \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{24}$ and $d(\tilde{c}, S) \leq \rho' \cdot \frac{\delta \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{24}$ (which holds assuming that $\bar{s}$ is a sufficiently large polynomial).

An averaging argument yields that in this case, for at least a $(1 - \frac{\delta \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{12})$-fraction of the rows $i \in I$ it holds that $d(\tilde{c}|_{\{i\}\times[n']}, S|_{\{i\}\times[n']}) = f(i) \leq \rho'$. □

Finally, we provide the proof of Claim 7.3.7.

Proof of Claim 7.3.7. We will show that each of the three properties in the definition of a good column holds for at least a $(1 - \frac{\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{3})$-fraction of the columns with probability at least 0.9 over the choice of I. The claim will then follow by a union bound over the choice of I and the fraction of bad columns.

Item 1. Assuming that $d(\tilde{c}, S) \leq \frac{\rho \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{3}$ (which once more holds assuming that $\bar{s}$ is a sufficiently large polynomial), an averaging argument implies that for at least a $(1 - \frac{\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{3})$-fraction of the columns $j \in [n']$ it holds that $d\left(\tilde{c}_j, S|_{[n]\times\{j\}}\right) \leq \rho$.

Item 2. Fix $j \in [n']$ and $c \in \mathcal{L}_j \setminus \{\tilde{c}_j\}$, and note that $d(c, \tilde{c}_j) \geq \delta$ since $\mathcal{C}$ has distance δ. For $i \in [n]$, let

$$f(i) := \begin{cases} 1, & \text{if } c_i = (\tilde{c}_j)_i \\ 0, & \text{otherwise.} \end{cases},$$

and note that by the sampling property of Γ, with probability at least $1 - \frac{0.1}{L} \cdot \frac{\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{3}$ over the choice of I we have that

$$d(c|_I, \tilde{c}_j|_I) = \mathop{\mathbb{E}}_{\boldsymbol{i} \in I}[f(\boldsymbol{i})] \geq \mathop{\mathbb{E}}_{\boldsymbol{i} \in [n]}[f(\boldsymbol{i})] - \gamma = d(c, \tilde{c}_j) - \gamma > \delta/2,$$

where the last inequality follows by choice of $\gamma < \delta/2$. Hence, by a union bound, with probability at least $1 - 0.1 \cdot \frac{\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{3}$ over the choice of I, we have $d(c|_I, \tilde{c}_j|_I) > \delta/2$ for all $c \in \mathcal{L}_j \setminus \{\tilde{c}_j\}$.

Finally, by an averaging argument we conclude that with probability at least 0.9 over the choice of I, at least a $(1 - \frac{\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{3})$-fraction of the columns $j \in [n']$ satisfy Item 2.

Item 3. By Claim 7.3.9, with probability at least 0.9 over the choice of $I = \{i_1, \ldots, i_m\}$, at least a $(1 - \frac{\delta \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{12})$-fraction of the rows in I are good, where for a good row $i_k \in I$ we have that $\tilde{c}|_{\{i_k\} \times [n']} \in \mathcal{L}'_{i_k}$, and so $\hat{c}_{i_k} = \tilde{c}|_{\{i_k\} \times [n']}$. Assuming this is the case, we have that $\tilde{c}$ agrees with $(\hat{c}_{i_1}, \ldots, \hat{c}_{i_m})$ on at least a $\left(1 - \frac{\delta \cdot \delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{12}\right)$-fraction of the points in $I \times [n']$, and so by averaging for at least a $(1 - \frac{\delta_{\mathsf{dec}} \cdot \delta'_{\mathsf{dec}}}{3})$-fraction of the columns $j \in [n']$ it holds that $d\left(\tilde{c}_j|_I, ((\hat{c}_{i_1})_j, \ldots, (\hat{c}_{i_m})_j)\right) \leq \delta/4$. □

Deterministic Algorithm

Lastly, to obtain a *deterministic* global list-recovery algorithm, we simply iterate over the randomness of Γ, and output the union of all output lists. This increases the running time by a multiplicative factor of $2^r = n \cdot \bar{s}/\rho'$. Moreover, Claim 7.3.4 guarantees that any codeword that is consistent with all but $(\rho'/\bar{s})$-fraction of the input lists will be output in one of the invocations, and consequently will be included in the final output list (which is of size at most $(L')^{\bar{s} \cdot L/(\rho')^2}$ by the same claim).

7.3.4 Deterministic Near-Linear Time Capacity-Achieving List-Recoverable Codes

In this section we prove the following lemma which implies Corollary 7.1.4 from the introduction.

Lemma 7.3.10. *For any constants $R \in [0, 1]$, $\varepsilon > 0$, and $\ell \geq 1$ there exists an infinite family of codes $\{\mathcal{C}_N\}_N$ that satisfy the following.*

- *$\mathcal{C}_N$ is an $\mathbb{F}_2$-linear code of block length N and alphabet size $N^{o(1)}$.*

- *$\mathcal{C}_N$ has rate at least R and relative distance at least $1 - R - \varepsilon$.*
- *$\mathcal{C}_N$ is $(1 - R - \varepsilon, \ell, N^{o(1)})$-globally list-recoverable deterministically in time $N^{1+o(1)}$.*
- *$\mathcal{C}_N$ is encodable deterministically in time $N^{1+o(1)}$.*

To prove the above lemma, we first use Theorem 7.1.3 to obtain deterministic nearly-linear time *high-rate* list-recoverable codes, and then use the Alon-Edmonds-Luby (AEL) distance amplification method [AEL95; AL96] to turn these codes into deterministic nearly-linear time *capacity-achieving* list-recoverable codes. Specifically, we shall use the following version of the AEL method for list-recovery from [GI02] which roughly says the following. Given an efficient "outer" code $\mathcal{C}$ of rate approaching 1 that is list-recoverable from a tiny fraction of errors, and a small "inner" code $\mathcal{C}'$ that is a (possibly non-efficient) capacity-achieving list-recoverable code, they can be combined to get a new code $\mathcal{C}_{\text{AEL}}$ that on the one hand, inherits the tradeoff between rate and error correction that $\mathcal{C}'$ enjoys, and on the other hand, is almost as efficient as $\mathcal{C}$.

Lemma 7.3.11 (Distance Amplification for List-Recovery, [GI02, Lemma 6])**.** *There exists an absolute constant b_0 such that the following holds for any $\delta, \rho, \varepsilon > 0$ and $t \geq (\delta \cdot \rho \cdot \varepsilon)^{-b_0}$.*

Suppose that $\mathcal{C} \subseteq (\Sigma^{R \cdot t})^n$ is an outer code of rate $1-\varepsilon$ and relative distance δ that is (ρ, ℓ, L)-globally list-recoverable in time T, and $\mathcal{C}' \subseteq \Sigma^t$ is an inner code of rate R and relative distance $1 - R - \varepsilon$ that is $(1 - R - \varepsilon, \ell', \ell)$-globally list-recoverable in time T'.

Then there exists a code $\mathcal{C}_{AEL} \subseteq (\Sigma^t)^n$ of rate $R - \varepsilon$ and relative distance $1 - R - 2\varepsilon$ that is $(1 - R - 2\varepsilon, \ell', L)$-globally list-recoverable in time $T + n \cdot (T' + \mathrm{poly}(t, \log n))$.

Moreover,

- *If $\mathcal{C}, \mathcal{C}'$ have encoding times T, T', respectively, then $\mathcal{C}_{AEL}$ has encoding time $T + n \cdot (T' + \mathrm{poly}(t, \log n))$.*
- *If $\mathcal{C}, \mathcal{C}'$ are $\mathbb{F}$-linear then so is $\mathcal{C}_{AEL}$.*

Remark 7.3.12. Lemma 6 in [GI02] is stated for the special case of $\ell' = 1$, and for a more specific choice of list-recovery radius and running times. Also, it does not mention explicitly relative distance, encoding time, and linearity. However, these can be deduced from the proof of the lemma, combined with the expander graph construction described in [Kop+17, Lemma 2.12] (see also [Gop+18, Lemma 5.4] for a similar transformation for the setting of local list-recovery).

Next we prove Lemma 7.3.10, based on Theorem 7.1.3 and Lemma 7.3.11. We will require the following code constructed in [HRZW17].

Theorem 7.3.13 ([HRZW17, Theorem A.1])**.** *There exists an absolute constant b_0 so that the following holds. For any $\varepsilon > 0$, $\ell \geq 1$, $q \geq \ell^{b_0/\varepsilon}$ that is an even power of a prime,*[4] *and integer $n \geq q^{b_0 \ell/\varepsilon}$, there exists a linear code $\mathcal{C} \leq \mathbb{F}^n$ of rate $1 - \varepsilon$ and relative distance $\Omega(\varepsilon^2)$ that is $(\Omega(\varepsilon^2), \ell, L)$-list-recoverable for $L = q^{q^{(\ell/\varepsilon) \cdot \exp(\log^* n)}}$. Moreover, $\mathcal{C}$ can be encoded in time $\mathrm{poly}(n, \log q)$ and list-recovered in time $\mathrm{poly}(n, L)$.*

Furthermore, we will require the following results. The first follows from the GV bound (Theorem 2.4.4), the second follows from a simple adaptation of the Zyablov-

[4] That is, q is of the form p^{2t} for a prime p and for an integer t.

Pinsker argument for list-decoding random linear codes.

Corollary 7.3.14. *For any $R \in [0,1]$ and $\varepsilon > 0$, and prime power $q \geq 2^{h_2(1-R-\varepsilon)/\varepsilon}$, a random linear code $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ of rate R has relative distance at least $1 - R - \varepsilon$ with probability $1 - \exp(-\Omega(n))$.*

Corollary 7.3.15 ([HRZW17], Corollary 2.2)**.** *For any $R \in [0,1]$, $\varepsilon > 0$, and $\ell \geq 1$, and for sufficiently large prime power q, a random linear code $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ of rate R is $(1 - R - \varepsilon, \ell, q^{O(\ell/\varepsilon)})$-list-recoverable with probability $1 - \exp(-\Omega(n))$.*

Proof of Lemma 7.3.10. We shall first apply Theorem 7.1.3 on a suitable base code $\mathcal{C}$ to obtain a deterministic near-linear time high-rate list-recoverable code $\mathcal{C}'$, and then use the transformation given by Lemma 7.3.11 to obtain a deterministic near-linear time capacity-achieving list-recoverable code $\mathcal{C}''$.

Base code $\mathcal{C}$: The code $\mathcal{C}$ will be the efficient high-rate list-recoverable code given by Theorem 7.3.13, in an appropriate setting of parameters.

Specifically, in what follows, we let $\beta := (\log\log\log N)^{-o(1)}$ (where the $o(1)$ term in the exponent is an arbitrarily slowly decreasing function of N), and we choose the block length of C to be N^β, and the rate to be $1 - \varepsilon\beta/4$. As we will see in a moment, the rationale for these choices is that if we raise $\mathcal{C}$ to the tensor power of $1/\beta$, Theorem 7.1.3 will yield a code of block length N with running time $N^{1+O(\beta)} = N^{1+o(1)}$ and rate greater than $1 - \varepsilon$.

Theorem 7.3.13 then guarantees, for any constant $\ell' \geq 1$, the existence of a linear code $\mathcal{C}$ as above that has relative distance $(\log\log\log N)^{-o(1)}$, and furthermore is $((\log\log\log N)^{-o(1)}, \ell', \exp\exp((\log\log\log N)^{o(1)}))$-globally list-recoverable in time $N^{O(\beta)}$, provided that the alphabet size is a sufficiently large even power of a prime on the order of $\exp((\log\log\log N)^{o(1)})$.

High-rate list-recoverable code $\mathcal{C}'$: Let $\mathcal{C}'$ be the code obtained by raising $\mathcal{C}$ to a tensor power of $1/\beta = (\log\log\log N)^{o(1)}$. Then the code $\mathcal{C}'$ has block length N, alphabet size $\exp((\log\log\log N)^{o(1)})$, rate at least $1 - \varepsilon/4$, and relative distance $\exp(-(\log\log\log N)^{o(1)})$. Furthermore, by Theorem 7.1.3, it is $(\exp(-(\log\log\log N)^{o(1)}), \ell', N^{o(1)})$-globally list-recoverable deterministically in time $N^{1+O(\beta)} = N^{1+o(1)}$.

Capacity-achieving list-recoverable code $\mathcal{C}''$: Let $\mathcal{C}''$ be the code obtained by applying Lemma 7.3.11 with the outer code being the code $\mathcal{C}'$ constructed so far, and the inner code being a capacity-achieving list-recoverable code $\mathcal{D}''$ of rate $R + \varepsilon/4$ and relative distance at least $1 - R - \varepsilon/2$.

Corollaries 7.3.14 and 7.3.15 guarantee the existence of a code $\mathcal{D}''$ as above that is $(1 - R - \varepsilon/2, \ell, \ell')$-globally list-recoverable for some constant ℓ', provided that the alphabet size is a sufficiently large constant prime power, and the block length is sufficiently large.

To satisfy the conditions of Lemma 7.3.11, we further require that the block length of $\mathcal{D}''$ is sufficiently large $\exp((\log\log\log N)^{o(1)})$, and that the alphabet size of $\mathcal{C}'$ is $\exp\exp((\log\log\log N)^{o(1)})$—the size of $\mathcal{D}''$—which can be achieved by grouping together consecutive symbols of $\mathcal{C}'$.

Lemma 7.3.11 then implies that $\mathcal{C}''$ is a code of block length N, alphabet size $N^{o(1)}$, rate R, and relative distance $1-R-\varepsilon$, that is $(1-R-\varepsilon, \ell, N^{o(1)})$-globally list-recoverable deterministically in time $N^{1+o(1)}$ (using brute-force decoding of the inner code).

Finally, it can be verified that encoding time is as claimed, and that all codes in the process can be taken to be $\mathbb{F}_2$-linear, and all transformations preserve $\mathbb{F}_2$-linearity, so the final code can be guaranteed to be $\mathbb{F}_2$-linear as well. □

7.3.5 Deterministic Near-Linear Time Unique Decoding up to the GV Bound

In this section we prove the following lemma which implies Corollary 7.1.6 from the introduction.

Lemma 7.3.16. *For any constants $R \in [0, 0.02]$ and $\varepsilon > 0$ there exists an infinite family of binary linear codes $\{\mathcal{C}_N\}_N$, where $\mathcal{C}_N$ has block length N and rate R, and is globally uniquely decodable deterministically from $\frac{h_2^{-1}(1-R)-\varepsilon}{2}$-fraction of errors in time $N^{1+o(1)}$.*

Furthermore, there exists a randomized algorithm which, on input N, runs in time $N^{1+o(1)}$ and outputs with high probability a description of a code $\mathcal{C}_N$ with the properties above. Given the description, the code $\mathcal{C}_N$ can be encoded deterministically in time $N^{1+o(1)}$.

To prove the above lemma, we rely on the following lemma from [Tho83; HRZW17] which says that one can turn a code that approximately satisfies the Singleton bound into one that approximately satisfies the GV bound via random concatenation. In what follows let $\theta(x) := 1 - h_2(1-2^{x-1})$ for $x \in [0,1]$.

Claim 7.3.17 ([GR10, Lemma 2.2])**.** *$\theta(x) \le x$ for all $x \in [0,1]$.*

Lemma 7.3.18 (Random Concatenation, [HRW17a, Lemma 7.3])**.** *There exists an absolute constant b_0 such that the following holds for any $\varepsilon > 0$, $R' \in [0,1]$, $R \in \left[0, \frac{\theta(R')-\varepsilon/2}{R'}\right]$, and $t \ge \frac{b_0}{\varepsilon^2\cdot(1-R)}$.*

Suppose that $\mathcal{C} \le (\mathbb{F}_2^{R'\cdot t})^n$ is an $\mathbb{F}_2$-linear code of rate R and relative distance $1-R-\frac{\varepsilon^2}{b_0}$, and $\boldsymbol{\mathcal{C}}_{\mathrm{con}} \le \mathbb{F}_2^{tn}$ is a code obtained from $\mathcal{C}$ by applying a random linear code $\boldsymbol{\mathcal{C}}^{(i)} \le \mathbb{F}_2^t$ of rate R' on each coordinate $i \in [n]$ of $\mathcal{C}$ independently. Then $\boldsymbol{\mathcal{C}}_{con}$ has relative distance at least $h_2^{-1}(1-R\cdot R')-\varepsilon$ with probability $1-\exp(-\Omega(n))$.

We shall also use the following lemma that states the effect of concatenation on list-recovery properties.

Lemma 7.3.19 (Concatenation for List-Recovery, [HRW17a, Lemma 7.4])**.** *Suppose that $\mathcal{C} \subseteq (\Sigma^{\rho'\cdot t})^n$ is (ρ, ℓ, L)-globally list-recoverable in time T, and $\mathcal{C}_{\mathrm{con}} \subseteq \Sigma^{tn}$ is a code obtained from $\mathcal{C}$ by applying a code $\mathcal{C}^{(i)} \subseteq \Sigma^t$ of rate R' on each coordinate $i \in [n]$ of $\mathcal{C}$. Suppose*

furthermore that at least $(1-\varepsilon)$-fraction of the codes $\mathcal{C}^{(i)}$ are (ρ', ℓ', ℓ)-globally list-recoverable in time T'. Then $\mathcal{C}_{\mathrm{con}}$ is $((\rho-\varepsilon)\cdot\rho', \ell', L)$-globally list-recoverable in time $T + n \cdot T'$.

Next we prove Lemma 7.3.16, based on Lemma 7.3.10 and the above Lemmas 7.3.18 and 7.3.19.

Proof of Lemma 7.3.16. We apply random concatenation on the deterministic near-linear time capacity-achieving list-recoverable code $\mathcal{C}$ given by Lemma 7.3.10. By Lemma 7.3.18, the resulting code $\tilde{\mathcal{C}}$ will approach the Gilbert-Varshmaov bound with high probability, while by Lemma 7.3.19, the code $\tilde{\mathcal{C}}$ will also be near-linear time list-recoverable (and in particular, list-decodable) with high probability. Thus, whenever the list-decoding radius exceeds half the minimum distance (which is the case whenever the rate is smaller than 0.02), the code $\tilde{\mathcal{C}}$ can be uniquely decoded from half the minimum distance in near-linear time by first running the list-decoding algorithm, and then choosing the codeword from the output list that is closest to the received word. Details follow.

The code $\mathcal{C}$: Let b_0 be the absolute constant guaranteed by Lemma 7.3.18, and apply Lemma 7.3.10 with rate $R_0 := \frac{R}{\theta^{-1}(R+\varepsilon/2)}$ (noting that this is at most 1 since $\theta(x) \le x$ for all $x \in [0,1]$, and θ is monotonically increasing in $[0,1]$), proximity parameter $\varepsilon := \varepsilon^2/b_0$, and input list size $\ell_0 := 2^{1/\varepsilon}$. Lemma 7.3.10 then guarantees that, for an infinite number of N's, the existence of an $\mathbb{F}_2$-linear code $\mathcal{C}$ of block length N, alphabet size $N^{o(1)}$, rate R_0, and relative distance $1-R_0-\varepsilon_0$, that is $(1-R_0-\varepsilon_0, \ell_0, N^{o(1)})$-globally list-recoverable deterministically in time $N^{1+o(1)}$.

The code $\tilde{\mathcal{C}}$: Let $\tilde{\boldsymbol{\mathcal{C}}} \le \mathbb{F}_2^{tN}$ be a binary linear code obtained from $\mathcal{C}$ by applying a random linear code $\boldsymbol{\mathcal{C}}^{(i)} \le \mathbb{F}_2^t$ of rate $R' := \theta^{-1}(R+\varepsilon/2)$ on each coordinate $i \in [n]$ of $\mathcal{C}$ independently. Then the code $\tilde{\boldsymbol{\mathcal{C}}}$ has rate ρ, and by Lemma 7.3.18 it also has relative distance at least $h_2^{-1}(1-R)-\varepsilon$ with probability $1-\exp(-N)$. Moreover, by Corollary 7.3.15, each $\boldsymbol{\mathcal{C}}^{(i)}$ is $(h_2^{-1}(1-R'-\varepsilon), 2^{1/\varepsilon})$-list-decodable with probability $1-o(1)$, so with probability $1-\exp(-\Omega(N))$ this property holds for at least $(1-\varepsilon^2/b_0)$-fraction of the $\boldsymbol{\mathcal{C}}^{(i)}$'s. Lemma 7.3.19 implies in turn that the code $\tilde{\boldsymbol{\mathcal{C}}}$ is $(\tilde{\rho}, N^{o(1)})$-globally list-decodable in time $N^{1+o(1)}$ (using brute-force decoding of inner codes $\boldsymbol{\mathcal{C}}^{(i)}$) for

$$\tilde{\rho} = (1 - R_0 - 2\varepsilon^2/b_0) \cdot h_2^{-1}(1-R'-\varepsilon)\,. \tag{7.2}$$

We now fix a code $\tilde{\mathcal{C}}$ achieving these parameters.

Decoding. Next assume that the list-decoding radius $\tilde{\rho}$ exceeds the desired decoding radius, i.e.,

$$(1 - R_0 - 2\varepsilon^2/b_0) \cdot h_2^{-1}(1-R'-\varepsilon) \ge \frac{h_2^{-1}(1-R)-\varepsilon}{2}\,, \tag{7.3}$$

where $R_0 := \frac{R}{\theta^{-1}(R+\varepsilon/2)}$ and $R' := \theta^{-1}(R+\varepsilon/2)$. It was shown in [Rud07, Section 4.4] that this is indeed the case whenever $R \le 0.02$ and ε is a sufficiently small constant.

Assuming that (7.3) holds, one can globally uniquely decode $\tilde{\mathcal{C}}$ up to half the minimum distance in time $N^{1+o(1)}$ by list-decoding $\tilde{\mathcal{C}}$, and outputting the codeword in the output list that is closest to the received word. □

7.4 Local List-Recovery

7.4.1 Local List-Recovery of High-Rate Tensor Codes

In this section we prove the following lemma which implies Theorem 7.1.8 from the introduction.

Lemma 7.4.1. *The following holds for any $\delta, \rho, \eta > 0$ and $s = \mathrm{poly}(1/\delta, 1/\rho)$. Suppose that $\mathcal{C} \leq \mathbb{F}^n$ is a linear code of relative distance δ that is (ρ, ℓ, L)-globally list-recoverable, and $(Q, \delta/2)$-locally correctable, and $t \geq 3$. Then $\mathcal{C}^{\otimes t} \leq \mathbb{F}^{n^t}$ is $(\tilde{Q}, \rho \cdot s^{-t^3}, \eta, \ell, L^{s^{t^3} \cdot \log^t L} \cdot \log(1/\eta))$-locally list-recoverable for*

$$\tilde{Q} = n^3 \cdot (Q \log Q)^t \cdot L^{s^{t^3} \cdot \log^t L} \cdot \log^2(1/\eta) \, .$$

Moreover, if $\mathcal{C}$ is globally list-recoverable in time $\mathrm{poly}(n)$, locally correctable in time T, and globally decodable for $(\delta/2)$-fraction of errors in time $\mathrm{poly}(n)$, then the local list-recovery algorithm for $\mathcal{C}^{\otimes t}$ has preprocessing time $\mathrm{poly}(n) \cdot L^{s^{t^3} \cdot \log^t L} \cdot \log^2(1/\eta)$ and running time $\mathrm{poly}(n) \cdot (T \log T)^t \cdot (s^{t^3} \log^t L)$.

Lemma 7.4.1 relies on the following lemma from [HRZW17] which says that the tensor product of a high-rate globally list-recoverable code (which is not necessarily locally correctable) is *approximately* locally list-recoverable. Approximate local list-recovery is a relaxation of local list-recovery, where the local algorithms in the output list are not required to recover *all* the codeword coordinates, but only *most* of them. Formally, a *β-approximately $(Q, \alpha, \eta, \ell, L)$-locally list-recoverable code $\mathcal{C} \subseteq \Sigma^n$* satisfies all the requirements of Definition 7.2.6, except that the requirement (7.1) is replaced with the relaxed condition that

$$\mathop{\mathbb{P}}_{\boldsymbol{i} \in [n]} (A_j(\boldsymbol{i}) = c_i) \geq 1 - \beta, \tag{7.4}$$

where the probability is over the choice of uniform random $\boldsymbol{i} \in [n]$,[5] and the soundness requirement is eliminated.

Lemma 7.4.2 (Approximate Local List-Recovery of High-Rate Tensor Codes, [HRW17b, Lemma 4.1])**.** *The following holds for any $\delta, \rho, \beta, \eta > 0$ and $s = \mathrm{poly}(1/\delta, 1/\rho, 1/\beta)$. Suppose that $\mathcal{C} \leq \mathbb{F}^n$ is a linear code of relative distance δ that is (ρ, ℓ, L)-globally list-recoverable. Then $\mathcal{C}^{\otimes t} \leq \mathbb{F}^{n^t}$ is β-approximately $(n \cdot (s^{t^2} \log^t L), \rho \cdot s^{-t^2}, \eta, \ell, L^{s^{t^2} \cdot \log^t L} \cdot \log(1/\eta))$-locally list-recoverable.*

[5] A standard averaging argument shows that in the case of approximate local list recovery, each of the local algorithms $A_1, \ldots, A_L$ can be assumed to be deterministic.

Moreover, if $\mathcal{C}$ is globally list-recoverable in time $\mathrm{poly}(n)$, then the approximate local list-recovery algorithm for $\mathcal{C}^{\otimes t}$ has preprocessing time $\log(n) \cdot L^{s^{t^2} \cdot \log^t L} \cdot \log(1/\eta)$ and running time $\mathrm{poly}(n) \cdot (s^{t^2} \log^t L)$.

To turn the approximate local list-recovery algorithm given by the above lemma into a local list-recovery algorithm we shall use the fact that the tensor product of a locally correctable code is also locally correctable with slightly worse parameters. A similar observation was made in [Vid15, Proposition 3.15.], but for completeness we provide a full proof below in Section 7.4.1.

Lemma 7.4.3 (Local Correction of Tensor Codes)**.** *Suppose that $\mathcal{C} \leq \mathbb{F}^n$ is a linear code that is (Q, ρ)-locally correctable. Then $\mathcal{C}^{\otimes t} \leq \mathbb{F}^{n^t}$ is $((O(Q \log Q))^t, \rho^t)$-locally correctable.*

Moreover, if $\mathcal{C}$ is locally correctable in time T, then the local correction algorithm for $\mathcal{C}^{\otimes t}$ runs in time $(O(T \log T))^t$.

To guarantee the soundness property we shall also use the following lemma which says that high-rate tensor codes are tolerantly locally testable. We prove this lemma in Section 7.4.1, based on a robust local testing procedure for high-rate tensor codes given in [Vid15].

Lemma 7.4.4 (Tolerant Local Testing of High-Rate Tensor Codes)**.** *Suppose that $\mathcal{C} \leq \mathbb{F}^n$ is a linear code of relative distance δ, and $t \geq 3$. Then $\mathcal{C}^{\otimes t} \leq \mathbb{F}^{n^t}$ is $(n^2 \cdot \delta^{-O(t)}, \delta^{O(t)}, (\delta/2)^t)$-tolerantly locally testable.*

Moreover, if $\mathcal{C}$ is globally decodable from $(\delta/2)$-fraction of errors in time T, then the tolerant local testing algorithm for $\mathcal{C}^{\otimes t}$ runs in time $T \cdot n \cdot \delta^{-O(t)}$.

Finally, we show a general transformation that turns an approximately locally list-recoverable code that is also locally correctable and tolerantly locally testable into a (genuinely) locally list-recoverable code.

Lemma 7.4.5. *Suppose that $\mathcal{C} \subseteq \Sigma^n$ is a β-approximately (Q, ρ, η, ℓ, L)-locally list-recoverable code that is also $(Q_{\mathrm{corr}}, \gamma)$-locally correctable and $(Q_{\mathrm{test}}, \beta, \gamma)$-tolerantly locally testable. Then $\mathcal{C}$ is $(\tilde{Q}, \rho, 2\eta, \ell, L)$-locally list-recoverable for*

$$\tilde{Q} = \max\{Q \cdot Q_{\mathrm{test}} \cdot O(L \log(L/\eta)), Q \cdot Q_{\mathrm{corr}}\} .$$

Moreover, if the approximate local list-recovery algorithm has preprocessing time T_{pre} and running time T, and the local correction and tolerant local testing algorithms run in times $T_{\mathrm{test}}, T_{\mathrm{corr}}$, respectively, then the local list-recovery algorithm has preprocessing time $T_{\mathrm{pre}} + T \cdot T_{\mathrm{test}} \cdot O(L \log(L/\eta))$ and running time $T \cdot T_{\mathrm{corr}}$.

Proof. First note that by Remark 7.2.3, we may assume that the tolerant local testing algorithm A_{test} fails with probability at most η/L, at the cost of increasing the query complexity and running time by a multiplicative factor of $O(\log(L/\eta))$.

The local list recovery algorithm $\tilde{A}$ first runs the approximate local list recovery algorithm A; denote the (deterministic) local algorithms that are output by $A_1, A_2, \ldots, A_L$, and let $w_1, \ldots, w_L$ be the words they implicitly compute. Then for each $j = 1, \ldots, L$, the local list recovery algorithm $\tilde{A}$ runs the tolerant local testing algorithm A_{test} on A_j,

and outputs $A_{\mathrm{corr}}(A_j)$ if and only if the test passes, where A_{corr} is the local correction algorithm.

It can be verified that the query complexity, output list size, and running times are as claimed. For completeness, suppose that $c \in \mathcal{C}$ satisfies $d(c, S) \leq \rho$. Then with probability at least $1 - \eta$ the approximate local list-recovery algorithm A will output some A_j for which $d(w_j, c) \leq \beta$. Consequently, the tolerant local testing algorithm A_{test} will accept w_j with probability at least $1 - \eta$. So we conclude that with probability at least $1 - 2\eta$ the local algorithm $A_{\mathrm{corr}}(w_j)$ will be included in the output list of $\tilde{A}$, and furthermore by the guarantees of A_{corr} it will be the case that $c = A_{\mathrm{corr}}(w_j)$.

For soundness, suppose that $A_{\mathrm{corr}}(w_j)$ is not consistent with some codeword $c \in \mathcal{C}$. Then by properties of A_{corr}, it holds that $d(w_j, \mathcal{C}) > \gamma$. But in this case the tolerant local testing algorithm A_{test} will reject w_j with probability at least $1 - \eta/L$. So by the union bound, with probability at least $1 - \eta$, each local algorithm in the output list of $\tilde{A}$ implicitly computes a codeword of $\mathcal{C}$. □

Next we prove Lemma 7.4.1 based on the above transformation and Lemmas 7.4.2, 7.4.3, and 7.4.4.

Proof of Lemma 7.4.1. By Lemma 7.4.3 the tensor product code $\mathcal{C}^{\otimes t}$ is $((O(Q \log Q))^t, (\delta/2)^t)$-locally correctable, and by Lemma 7.4.4 it is $(n^2 \cdot \delta^{-O(t)}, \delta^{b_0 t}, (\delta/2)^t)$-tolerantly locally testable for some absolute constant b_0. Moreover, by Lemma 7.4.2 the tensor product code $\mathcal{C}^{\otimes t}$ is $(\delta^{b_0 t})$-approximately $(n \cdot (s^{t^3} \log^t L), \rho \cdot s^{-t^3}, \eta/2, \ell, L^{s^{t^3} \cdot \log^t L} \cdot \log(1/\eta))$-locally list-recoverable. Finally, Lemma 7.4.5 implies that $\mathcal{C}^{\otimes t}$ is $(\tilde{Q}, \rho \cdot s^{-t^3}, \eta, \ell, L^{s^{t^3} \cdot \log^t L} \cdot \log(1/\eta))$-locally list-recoverable for

$$\tilde{Q} = n^3 \cdot (Q \log Q)^t \cdot L^{s^{t^3} \cdot \log^t L} \cdot \log^2(1/\eta) \, .$$

Running times follow similarly. □

Local Correction of Tensor Codes: Proof of Lemma 7.4.3

Lemma 7.4.3 can be deduced from the following lemma using induction.

Lemma 7.4.6. *Suppose that $\mathcal{C} \leq \mathbb{F}^n$, $\mathcal{C}' \leq \mathbb{F}^{n'}$ are linear codes that are (Q, ρ), (Q', ρ')-locally correctable, respectively. Then $\mathcal{C} \otimes \mathcal{C}' \leq \mathbb{F}^{n \times n'}$ is $(Q \cdot O(Q' \log Q'), \rho \cdot \rho')$-locally correctable.*

Moreover, if $\mathcal{C}, \mathcal{C}'$ are locally correctable in times T, T', respectively, then the local correction algorithm for $\mathcal{C} \otimes \mathcal{C}'$ runs in time $T \cdot O(T' \log T')$.

Proof. First note that by Remark 7.2.5, we may assume that the local correction algorithm A' for $\mathcal{C}'$ fails with probability at most $1/6$, at the cost of increasing the query complexity and running time by some multiplicative constant b_0. Similarly, we may also assume that the local correction algorithm A for $\mathcal{C}$ fails with probability at most $1/(6 b_0 Q')$, at the cost of increasing the query complexity and running time by a multiplicative factor of $O(\log Q')$.

Let $w \in \mathbb{F}^{n \times n'}$ be a string that is $(\rho \cdot \rho')$-close to some codeword $c \in \mathcal{C} \otimes \mathcal{C}'$. Recall that the local correction algorithm $\tilde{A}$ for $\mathcal{C} \otimes \mathcal{C}'$ is given as input a codeword coordinate $(i, j) \in [n] \times [n']$ in the tensor product code $\mathcal{C} \otimes \mathcal{C}'$, is allowed to query the received word w at every coordinate of $\mathcal{C} \otimes \mathcal{C}'$, and must produce a guess for $c_{i,j}$, the codeword value indexed by (i, j).

To this end, the local correction algorithm $\tilde{A}$ for $\mathcal{C} \otimes \mathcal{C}'$ first runs the local correction algorithm A' for $\mathcal{C}'$ on input $j \in [n']$. Let $J = \{j_1, \ldots, j_m\} \subseteq [n']$ be the set of query locations, where $m := b_0 \cdot Q'$. Next for each query location $j_r \in J$, the algorithm $\tilde{A}$ obtains a guess for the symbol at position (i, j_r) by running the local correction algorithm A for $\mathcal{C}$ on input i with oracle access to the column j_r. Let v_r be the guess for the symbol at position (i, j_r) produced by A. At this point we have candidate symbols $(v_1, \ldots, v_m)$ for all positions in $\{i\} \times J$. Finally, the algorithm $\tilde{A}$ responds with the output of A' on query locations $j_1, \ldots, j_m$ and values $v_1, \ldots, v_m$. The formal description of the local correction algorithm $\tilde{A}$ is given in Algorithm 2.

Algorithm 2 The local correction algorithm $\tilde{A}$ for $\mathcal{C} \otimes \mathcal{C}'$.

function $\tilde{A}((i, j) \in [n] \times [n'])$ ▷ $\tilde{A}$ receives oracle access to a matrix $w \in \mathbb{F}^{n \times n'}$.
  Run the local correction algorithm A' for $\mathcal{C}'$ on input j, let $J = \{j_1, \ldots, j_m\} \subseteq [n']$ be the query locations for $m = b_0 \cdot Q'$.
  for $r = 1, \ldots, m$ **do**
    Run the local correction algorithm A for $\mathcal{C}$ on input i and oracle access to the j_r-th column $w|_{[n] \times \{j_r\}}$.
    Let $v_r \leftarrow A(i)$. ▷ v_r is a candidate for the symbol at position $(i, j_r) \in [n] \times [n']$.
  end for ▷ At this point, we have candidate symbols $(v_1, \ldots, v_m)$ for every position in $\{i\} \times J$.
  Let v be the output of A' on query locations $j_1, \ldots, j_m$ and values $v_1, \ldots, v_m$.
  Return: v
end function

The algorithm $\tilde{A}$ invokes the algorithm A' once, followed by $m = O(Q')$ invocations of the algorithm A. Thus, the query complexity of $\tilde{A}$ is

$$O(Q') + m \cdot O(Q \cdot \log Q') = Q \cdot O(Q' \log Q') \,,$$

and the running time is

$$O(T') + m \cdot O(T \cdot \log Q') = O(T') + T \cdot O(Q' \log Q') = T \cdot O(T' \log T') \,.$$

As for correctness, recall that by assumption the received word w is $(\rho \cdot \rho')$-close to the codeword $c \in \mathcal{C} \otimes \mathcal{C}'$. Let us call a column *good* if w and c are ρ-close on this column, and note that by Markov's inequality, at least $(1 - \rho')$-fraction of the columns are good. Furthermore, by our assumptions on each good column the local correction algorithm A for $\mathcal{C}$ succeeds with probability at least $1 - \frac{1}{6m}$, and so by union bound, with probability at least $5/6$ the values $(v_1, \ldots, v_m)$ will be computed correctly on each good column. Conditioned on this, the local correction algorithm A' for $\mathcal{C}'$ computes v correctly with probability at least $5/6$, so the total success probability is $2/3$. □

Tolerant Local Testing of High-Rate Tensor Codes: Proof of Lemma 7.4.4

The proof of Lemma 7.4.4 relies on the following *robust* local testing procedure for high-rate tensor codes from [Vid15] which is a local testing procedure with the property that local views of words far from the code are, on average, far from an accepting view.

Theorem 7.4.7 (Robust Local Testing of High-Rate Tensor Codes, [Vid15, Theorem 3.1])**.** *Suppose that $\mathcal{C} \leq \mathbb{F}^n$ is a linear code of relative distance δ, and $t \geq 3$. Then for any $w \in \mathbb{F}^{n^t}$, the expected relative distance of w from $\mathcal{C}^{\otimes 2}$ on a random axis-parallel plane is at least $\delta^{O(t)} \cdot d(w, \mathcal{C}^{\otimes t})$.*

Proof of Lemma 7.4.4. Say we are given a string $w \in \mathbb{F}^{n^t}$ and we need to test if it is close to a codeword of $\mathcal{C}^{\otimes t}$. Let $\tau \geq \delta^{O(t)}$ be some threshold parameter to be chosen later. The test is to choose a random axis-parallel plane $\boldsymbol{P}$ in $\mathbb{F}^{n^t}$ and find if there is a codeword $c \in \mathcal{C}^{\otimes 2}$ which is τ-close to $w|_{\boldsymbol{P}}$. If yes, then accept, else reject. Clearly this test makes only n^2 queries. Also by Corollary 7.2.10, when $\tau < (\delta/2)^2$, this can be implemented in $O(T \cdot n)$ time.

To show completeness, let $w \in \mathbb{F}^{n^t}$ be some string which is ρ-close to a codeword $c \in \mathcal{C}^{\otimes t}$ for $\rho \geq \delta^{O(t)}$ to be chosen later. Since individual coordinates on a random axis-parallel plane are marginally uniform over $\mathbb{F}^{n^t}$, by Markov's inequality, the probability that $w|_{\boldsymbol{P}}$ is τ-far from $c|_{\boldsymbol{P}} \in C^{\otimes 2}$ is at most ρ/τ. So the probability that the test rejects w is at most $p_0 := \rho/\tau$.

To show soundness, let $w \in \mathbb{F}^{n^t}$ be some string which is $(\delta/2)^t$-far from any codeword $c \in \mathcal{C}^{\otimes t}$. Then by Theorem 7.4.7, the expected relative distance of $w|_{\boldsymbol{P}}$ from $\mathcal{C}^{\otimes 2}$ is at least $\delta^{O(t)}$. Thus the probability that the test rejects w is at least $p_1 := \frac{\delta^{O(t)} - \tau}{1 - \tau}$.

Next observe that we can choose $\tau \geq \delta^{O(t)}$ and $\rho \geq \delta^{O(t)}$ sufficiently small so that $p_0 < p_1$. Finally to get the acceptance and rejection probabilities to $2/3$ as in the definition of tolerant locally testable codes, we repeat the above local test $\delta^{-O(t)}$ times and accept a string if it is accepted in at least $\frac{p_0+p_1}{2}$-fraction of the tests. By a Chernoff bound, the new test will have the required soundness and completeness. □

7.4.2 Capacity-Achieving Locally List-Recoverable Codes

In this section we prove the following lemma which shows the existence of capacity-achieving locally list-recoverable codes. An analogous lemma was proved in [HRW17b, Lemma 5.3]; however, only local decoding of *message* coordinates was guaranteed, and there was no soundness property. That we are able to locally correct *codeword* coordinates and guarantee the soundness property will be crucial for our GV bound local correction application.

Lemma 7.4.8. *For any constants $R \in [0, 1]$, $\varepsilon > 0$, $\eta > 0$, and $\ell \geq 1$ there exists an infinite family of codes $\{\mathcal{C}_N\}_N$ that satisfy the following.*

- *$\mathcal{C}_N$ is an $\mathbb{F}_2$-vector linear*[6] *code of block length N and alphabet size $N^{o(1)}$.*
- *$\mathcal{C}_N$ has rate R and relative distance at least $1 - R - \varepsilon$.*
- *$\mathcal{C}_N$ is $(N^{o(1)}, 1-R-\varepsilon, \eta, \ell, N^{o(1)})$-locally list-recoverable with preprocessing and running time $N^{o(1)}$.*
- *$\mathcal{C}_N$ is encodable in time $N^{1+o(1)}$.*

As in the proof of Lemma 7.3.10, we first use Lemma 7.4.1 to obtain *high-rate* locally list-recoverable codes, and then use the Alon-Edmonds-Luby (AEL) distance amplification method [AEL95; AL96] to turn these codes into *capacity-achieving* locally list-recoverable codes. However, this time we shall use the following version of the AEL method for *local* list-recovery from [Gop+18].

Lemma 7.4.9 (Distance Amplification for Local List-Recovery, [Gop+18, Lemma 5.4])**.** *There exists an absolute constant b_0 such that the following holds for any $\delta, R, \varepsilon > 0$ and $t \geq (\delta \cdot R \cdot \varepsilon)^{-b_0}$.*

Suppose that $\mathcal{C} \subseteq (\Sigma^{R \cdot t})^n$ is an outer code of rate $1 - \varepsilon$ and relative distance δ that is (Q, ρ, η, ℓ, L)-locally list-recoverable, and $\mathcal{C}' \subseteq \Sigma^t$ is an inner code of rate R and relative distance $1-R-\varepsilon$ that is $(1-R-\varepsilon, \ell', \ell)$-globally list-recoverable. Then there exists a code $\mathcal{C}_{\mathrm{AEL}} \subseteq (\Sigma^t)^n$ of rate $R - \varepsilon$ and relative distance $1 - R - 2\varepsilon$ that is $(Q \cdot \mathrm{poly}(t), 1 - R - 2\varepsilon, \eta, \ell', L)$-locally list-recoverable.

Moreover,

- *If the local list-recovery algorithm for $\mathcal{C}$ has preprocessing time T_{pre} and running time T, and $\mathcal{C}'$ can be globally list-recovered in time T', then the local list-recovery algorithm for C_{AEL} has preprocessing time $T_{\mathrm{pre}} + Q \cdot (T' + \mathrm{poly}(t, \log n))$ and running time $T + Q \cdot \mathrm{poly}(t) \cdot (T' + \mathrm{poly}(\log n))$.*
- *If $\mathcal{C}, \mathcal{C}'$ have encoding times T, T', respectively, then $\mathcal{C}_{\mathrm{AEL}}$ has encoding time $T + n \cdot (T' + \mathrm{poly}(t, \log n))$.*
- *If $\mathcal{C}, \mathcal{C}'$ are $\mathbb{F}$-vector linear then so is $\mathcal{C}_{AEL}$.*

To apply Lemma 7.4.1, we shall also need a high-rate base code that is both globally list-recoverable and locally correctable. We obtain such a code by intersecting the high-rate globally list-recoverable codes given by Theorem 7.3.13 with the high-rate locally correctable codes given by the following lemma.

Lemma 7.4.10 (High-Rate Locally Correctable Codes)**.** *For any $\varepsilon, \beta > 0$, and integer N where $q := N^\beta$ is a prime power, there exists a code $\mathcal{C}_N$ that satisfies the following.*

- *$\mathcal{C}_N$ is an $\mathbb{F}_q$-vector linear code of block length N and alphabet size $N^{(\varepsilon\beta)^{-O(1/\beta)}}$.*
- *$\mathcal{C}_N$ has rate $1 - \varepsilon$ and relative distance $\Omega(\varepsilon \cdot \beta)$.*
- *$\mathcal{C}_N$ is $(N^\beta \cdot (\varepsilon\beta)^{-O(1/\beta)}, \Omega(\varepsilon \cdot \beta))$-locally correctable in time $N^\beta \cdot (\varepsilon\beta)^{-O(1/\beta)}$.*
- *$\mathcal{C}_N$ is encodable in time $\mathrm{poly}(N)$.*

We prove Lemma 7.4.10 in Section 7.4.2, based on the high-rate locally correctable

[6]That is, the alphabet $\Sigma = \mathbb{F}_2^t$ for a positive integer t, and the $\mathcal{C}_N$ is closed under addition (where elements of $(\mathbb{F}_2^t)^N$ are added in the natural way.)

codes of [KSY14]. Next we prove Lemma 7.4.8, based on Lemmas 7.4.1, 7.4.9, and 7.4.10.

Proof of Lemma 7.4.8. The proof is similar to that of Lemma 7.3.10, with the main difference being that now we need to ensure that the base code is locally correctable. Specifically, we shall first apply Lemma 7.4.1 on a suitable high-rate base code $\mathcal{C}$ that is both globally list-recoverable and locally correctable to obtain a high-rate locally list-recoverable code $\mathcal{C}'$, and then use the transformation given by Lemma 7.4.9 to obtain a capacity-achieving locally list-recoverable code $\mathcal{C}''$.

Base code $\mathcal{C}$. The code $\mathcal{C}$ will be the intersection of the efficient high-rate globally list-recoverable code given by Theorem 7.3.13 with the high-rate locally correctable code given by Lemma 7.4.10, in an appropriate setting of parameters.

Specifically, let $\beta := (\log\log N)^{-o(1)}$ (where the $o(1)$ term in the exponent is an arbitrarily slowly decreasing function of N), and we choose the block length of $\mathcal{C}$ to be N^β, and the rate to be $1 - \varepsilon\beta/4$. The code $\mathcal{C}$ will be constructed in turn as $\mathcal{D}_1 \cap \mathcal{D}_2$, where $\mathcal{D}_1$ is the high-rate globally list-recoverable code given by Theorem 7.3.13, and $\mathcal{D}_2$ is obtained using the high-rate locally correctable code given by Lemma 7.4.10, and both codes $\mathcal{D}_1, \mathcal{D}_2$ have block length N^β and rate $1 - \varepsilon\beta/8$. Details follow.

The code $\mathcal{D}_1$. Let $\ell' \geq 1$ be a constant to be fixed later, and let $\mathcal{D}_1$ be the linear code guaranteed by Theorem 7.3.13 of block length N^β, rate $1 - \varepsilon\beta/8$, and relative distance $(\log\log N)^{-o(1)}$, that is $((\log\log N)^{-o(1)}, \ell', \exp\exp((\log\log N)^{o(1)}))$-globally list-recoverable. Note that such a code exists provided that the alphabet size is sufficiently large even power of a prime $q := \exp((\log\log N)^{o(1)})$.

The code $\mathcal{D}_2$. The code $\mathcal{D}_2$ will be constructed in turn as the concatenation of the high-rate locally correctable code $\mathcal{D}_2'$ given by Lemma 7.4.10 with an efficiently encodable and decodable linear code $\mathcal{D}_2''$. To construct $\mathcal{D}_2''$, we will use Reed-Solomon codes (see Example 2.5.1). The purpose of the concatenation is (a) to reduce the alphabet size of $\mathcal{D}_2'$ to that of $\mathcal{D}_1$ and (b) make the code $\mathcal{D}_2'$ linear.

We first describe the code $\mathcal{D}_2'$. Suppose that N^{β^2} is a power of q (which holds for infinite number of N's). Lemma 7.4.10 guarantees the existence of an $\mathbb{F}_q$-vector linear code $\mathcal{D}_2'$ of length $N^\beta \cdot (1 - \varepsilon\beta/16)$, alphabet size q^a for $a = (\log N)^{1+o(1)}$, rate $1 - \varepsilon\beta/16$, and relative distance $(\log\log N)^{-o(1)}$, that is $(N^{O(\beta^2)}, (\log\log N)^{-o(1)})$-locally correctable.

Next we describe the code $\mathcal{D}_2''$. The code $\mathcal{D}_2''$ will be an efficiently encodable and decodable linear code of length $\frac{1}{1-\varepsilon\beta/16} \cdot a$, alphabet size q, rate $1 - \gamma\beta/16$, and relative distance $(\log\log N)^{-o(1)}$. The code $\mathcal{D}_2''$ can be obtained in turn by taking a Reed-Solomon code (Example 2.5.1) of length $\frac{1-\varepsilon\beta/32}{1-\varepsilon\beta/16} \cdot a$, alphabet size $q^{\log\log N}$ (noting that $q^{\log\log N} > \log^2 N > a$), rate $1 - \varepsilon\beta/32$, and relative distance $(\log\log N)^{-o(1)}$, and concatenating it with another Reed-Solomon code of length $\frac{1}{1-\varepsilon\beta/32} \cdot \log\log N$, alphabet size q, rate $1 - \varepsilon\beta/32$, and relative distance $(\log\log N)^{-o(1)}$.

Finally, by concatenating $\mathcal{D}_2'$ with $\mathcal{D}_2''$ we obtain a vector linear code $\mathcal{D}_2$ of length N^β, alphabet size $q = \exp((\log\log N)^{o(1)})$, rate $1 - \varepsilon\beta/8$, and relative distance $(\log\log N)^{-o(1)}$, that is $(N^{O(\beta^2)}, (\log\log N)^{-o(1)})$-locally correctable.

We conclude that $\mathcal{C} := \mathcal{D}_1 \cap \mathcal{D}_2$ is a linear code of block length N^β, alphabet size $\exp((\log\log N)^{o(1)})$, rate $1 - \varepsilon\beta/4$, and relative distance $(\log\log N)^{-o(1)}$, which is both $((\log\log N)^{-o(1)}, \ell', \exp\exp((\log\log N)^{o(1)}))$-globally list-recoverable and $(N^{O(\beta^2)}, (\log\log N)^{-o(1)})$-locally correctable.

High-rate locally list-recoverable code $\mathcal{C}'$. Let $\mathcal{C}'$ be the code obtained by raising $\mathcal{C}$ to a tensor power of $1/\beta = (\log\log N)^{o(1)}$. Then C' has block length N, alphabet size $\exp((\log\log N)^{o(1)})$, rate at least $1 - \varepsilon/4$, and relative distance $\exp(-(\log\log N)^{o(1)})$. Furthermore, by Lemma 7.4.1, it is $(N^{o(1)}, \exp(-(\log\log N)^{o(1)}), \eta, \ell', N^{o(1)})$-locally list-recoverable.

Capacity-achieving locally list-recoverable code $\mathcal{C}''$. Let $\mathcal{C}''$ be the code obtained by applying Lemma 7.4.9 with the outer code being the code $\mathcal{C}'$ constructed so far, and the inner code being a capacity-achieving globally list-recoverable code $\mathcal{D}''$ of rate $R + \varepsilon/4$ and relative distance at least $1 - R - \varepsilon/2$.

Corollaries 7.3.14 and 7.3.15 guarantee the existence of a code $\mathcal{D}''$ as above that is $(1 - R - \varepsilon/2, \ell, \ell')$-globally list-recoverable for some constant ℓ', provided that the alphabet size is a sufficiently large constant prime power, and the block length is sufficiently large. To satisfy the conditions of Lemma 7.4.9, we further require that the block length of $\mathcal{D}''$ is sufficiently large, i.e., on the order $\exp((\log\log N)^{o(1)})$, and that the alphabet size of $\mathcal{C}'$ is $\exp\exp((\log\log N)^{o(1)})$—the size of $\mathcal{D}''$—which can be achieved by grouping together consecutive symbols of $\mathcal{C}'$.

Lemma 7.4.9 then implies that $\mathcal{C}''$ is a code of block length N, alphabet size $N^{o(1)}$, rate R, and relative distance $1 - R - \varepsilon$, that is $(N^{o(1)}, 1 - R - \varepsilon, \eta, \ell, N^{o(1)})$-locally list-recoverable.

Finally, it can be verified that running times are as claimed (using brute-force encoding and decoding of inner code $\mathcal{D}''$), and that all codes in the process can be taken to be $\mathbb{F}_2$-vector linear, and all transformations preserve $\mathbb{F}_2$-vector linearity, so the final code can be guaranteed to be $\mathbb{F}_2$-vector linear as well. □

High-Rate Locally Correctable Codes: Proof of Lemma 7.4.10

Lemma 7.4.10 is a consequence of the following theorem from [KSY14], summarizing the parameters of multiplicity codes.

Theorem 7.4.11 (Multiplicity Codes, [KSY14, Lemmas 3.5 and 3.6] and [Kop15b])**.** *The following holds for any integers s, d, m, and for any prime power $q \geq \max\{10 \cdot m, \frac{d+6\cdot s}{s}, 12 \cdot (s+1)\}$. There exists an $\mathbb{F}_q$-vector linear code $\mathcal{C}$ of block length q^m, alphabet size $q^{\binom{m+s-1}{m}}$, relative*

distance at least $\delta := 1 - \frac{d}{s \cdot q}$*, and rate at least* $\left(1 - \frac{m^2}{s}\right) \cdot (1-\delta)^m$*, that is* $(O(s^m \cdot q), \delta/10)$*-locally correctable.*

Moreover, $\mathcal{C}$ *can be locally corrected in time* $O(q/\delta^m)$ *and encoded in time* $\mathrm{poly}\left(q^m, \binom{m+s-1}{m}\right)$.

Proof of Lemma 7.4.10. We set the code $\mathcal{C}_N$ to be the code given by Theorem 7.4.11 with the following parameters. We choose $q := N^\beta$ to be the field size (which exists whenever q is a prime power), and choose $m = 1/\beta$. Note that indeed $q^m = N$. We choose $s = 2m^2/\varepsilon$, $\delta = \varepsilon/(2m)$, and $d = s \cdot q \cdot (1 - \delta)$.

The alphabet size of the code is

$$q^{\binom{m+s-1}{m}} \leq N^{\beta \cdot (m+s)^m} \leq N^{(\varepsilon\beta)^{-O(1/\beta)}},$$

the relative distance is at least $\delta \geq \Omega(\varepsilon \cdot \beta)$, and the rate is at least

$$\left(1 - \frac{m^2}{s}\right) \cdot (1-\delta)^m = \left(1 - \frac{\varepsilon}{2}\right)\left(1 - \frac{\varepsilon}{2m}\right)^m \; geq 1 - \varepsilon.$$

Furthermore, $\mathcal{C}_N$ is locally correctable from $\Omega(\varepsilon\beta)$-fraction of errors with query complexity

$$O(s^m \cdot q) \leq N^\beta \cdot (\varepsilon\beta)^{-O(1/\beta)}.$$

as required.

Finally, it can be verified that running times are as required. □

7.4.3 Local Correction up to the GV Bound

In this section we prove the following lemma which implies Corollary 7.1.9 from the introduction.

Lemma 7.4.12. *For any constants* $R \in [0, 0.02]$ *and* $\varepsilon > 0$ *there exists an infinite family of binary linear codes* $\{\mathcal{C}_N\}_N$*, where* $\mathcal{C}_N$ *has block length* N *and rate* R*, and is locally correctable from* $\frac{h_2^{-1}(1-R)-\varepsilon}{2}$*-fraction of errors with query complexity* $N^{o(1)}$.

Furthermore,

- *The local correction algorithm for* $\mathcal{C}_N$ *runs in time* $N^{o(1)}$.
- *There exists a randomized algorithm which, on input* N*, runs in time* $N^{1+o(1)}$ *and outputs with high probability a description of a code* $\mathcal{C}_N$ *with the properties above. Given the description, the code* $\mathcal{C}_N$ *can be encoded deterministically in time* $N^{1+o(1)}$.

Similar to Lemma 7.3.16, the proof of the above lemma relies on random concatenation (Lemma 7.3.18), as well as the following lemma that is an analog of Lemma 7.3.19 for the setting of local list-recovery.

Lemma 7.4.13 (Concatenation for Local List-Recovery)**.** *Suppose that* $\mathcal{C} \subseteq (\Sigma^{R' \cdot t})^n$ *is* (Q, ρ, η, ℓ, L)*-locally list-recoverable, and* $\mathcal{C}_{\mathrm{con}} \subseteq \Sigma^{tn}$ *is a code obtained from* $\mathcal{C}$ *by applying a code* $\mathcal{C}^{(i)} \subseteq \Sigma^t$ *of rate* R' *on each coordinate* $i \in [n]$ *of* $\mathcal{C}$*. Suppose furthermore that at*

least a $(1-\varepsilon)$-fraction of the codes $\mathcal{C}^{(i)}$ are (ρ', ℓ', ℓ)-globally list-recoverable. Then $\mathcal{C}_{\mathrm{con}}$ is $(Q \cdot t, (\rho-\varepsilon)\cdot\rho', \eta, \ell', L)$-locally list-recoverable.

Moreover, if the local list-recovery algorithm for $\mathcal{C}$ has preprocessing time T_{pre} and running time T, and each $\mathcal{C}^{(i)}$ can be globally list-recovered in time T', then the local list-recovery algorithm for $\mathcal{C}_{\mathrm{con}}$ has preprocessing time $T_{\mathrm{pre}} + Q \cdot T'$ and running time $T + Q \cdot T'$.

We prove the above lemma in Section 7.4.3. Finally, we shall also use the following lemma which shows that a locally list-decodable code (satisfying the soundness property) is also locally correctable.

Lemma 7.4.14. *Suppose that $\mathcal{C} \subseteq \Sigma^n$ is a code of relative distance δ that is $(Q, \rho, 0.1, L)$-locally list-decodable for $\rho < \delta/2$. Then $\mathcal{C}$ is $\left(O\big(Q \cdot L \cdot \frac{\log^2 n}{(\delta/2-\rho)^2}\big), \rho\right)$-locally correctable.*

Moreover, if the local list-decoding algorithm has preprocessing time T_{pre} and running time T, then the local correction algorithm runs in time $T_{\mathrm{pre}} + O\left(T \cdot L \cdot \frac{\log^2 n}{(\delta/2-\rho)^2}\right)$.

Proof. We first run the local list-decoding algorithm, and then choose a local corrector from the output list that is sufficiently close to the received word (which can be checked via sampling).

Specifically, let A be the local list-decoding algorithm for $\mathcal{C}$. By Remark 7.2.5 we may assume that both the completeness and soundness properties of A hold with success probability $1-\frac{1}{n^{10}}$ instead of $\frac{2}{3}$ at the cost of increasing the query complexity and running time by a multiplicative factor of $O(\log n)$.

On oracle access to $w \in \Sigma^n$ and input coordinate $i \in [n]$, the local correction algorithm A_{corr} for $\mathcal{C}$ first runs the local list-decoding algorithm A for $\mathcal{C}$; let $A_1, \ldots, A_L$ be the local algorithms in the output list of A. Then for each $j = 1, \ldots, L$, the algorithm A_{corr} runs A_j on a random subset $S_j \subseteq [n]$ of $O\left(\frac{\log n}{(\delta/2-\rho)^2}\right)$ coordinates, and computes the fraction δ_j of coordinates in S_j on which the decoded values differ from the corresponding values of w. Finally, the algorithm A_{corr} finds some A_j for which $\delta_j \leq \delta/2$ (if such A_j exists), and uses A_j to locally correct the input coordinate i. Clearly, the query complexity and running time of A_{corr} are as claimed. We proceed to show that A_{corr} satisfies the local correction guarantees.

Let $c \in \mathcal{C}$ be the (unique) codeword which satisfies that $d(w, c) \leq \rho$. We shall show below that with probability $0.9 - o(1)$, there exists some A_j that computes c and satisfies that $\delta_j \leq \delta/2$, and on the other hand, with probability $0.9 - o(1)$, any A_j which does not compute c satisfies that $\delta_j > \delta/2$. This will imply in turn that the algorithm A_{corr} will succeed in decoding the input coordinate correctly with probability $0.8 - o(1) \geq \frac{2}{3}$ as required.

We first show that with probability $0.9 - o(1)$, there exists some A_j that computes c and satisfies that $\delta_j \leq \delta/2$. To see this note that by the completeness property of A, since $d(w, c) \leq \rho$ with probability at least 0.9 over the randomness of A there exists some A_j that computes c. In this case, by a union bound, with probability $1 - o(1)$ it holds that each decoded coordinate of A_j in S_j equals the corresponding coordinate in

c. Furthermore, a Chernoff bound demonstrates that with probability $1 - o(1)$ it holds that w and c differ on S_j in at most a $\frac{\delta}{2}$-fraction of the coordinates. Consequently, with probability $0.9 - o(1)$ it holds that $\delta_j \leq \delta/2$.

Next we show that with probability $0.9 - o(1)$, any A_j which does not compute c satisfies that $\delta_j > \delta/2$. For this note that by the soundness property of A, with probability at least 0.9 over the randomness of A, any such A_j computes some codeword $c' \in \mathcal{C} \setminus \{c\}$. As above, a union bound guarantees that with probability $1 - o(1)$ it holds that for any such A_j, each decoded coordinate of A_j in S_j equals the corresponding coordinate in c'. On the other hand, since $\mathcal{C}$ has relative distance δ and $d(w, c) \leq \rho$, we have that $d(w, c') \geq \delta - \rho = \delta/2 + (\delta/2 - \alpha)$, and so by a Chernoff bound, with probability $1 - o(1)$ for any such A_j it holds that w and c' differ on S_j in more than a $\frac{\delta}{2}$-fraction of the coordinates. Consequently, with probability $0.9 - o(1)$ it holds that $\delta_j > \delta/2$ for any such A_j. □

Next we prove Lemma 7.4.12, based on the above lemma and Lemmas 7.4.8, 7.3.18, and 7.4.13.

Proof of Lemma 7.4.12. The proof is similar to that of Lemma 7.3.16. As in Lemma 7.3.16, we apply random concatenation on the capacity-achieving locally list-recoverable code $\mathcal{C}$ given by Lemma 7.4.8. By Lemma 7.3.18, the resulting code $\tilde{\mathcal{C}}$ will approach the Gilbert-Varshamov bound with high probability, while by Lemma 7.4.13, the code $\tilde{\mathcal{C}}$ will also be locally list-recoverable (and in particular, locally list-decodable) with high probability. Lemma 7.4.14 then implies that whenever the list-decoding radius exceeds the desired local correction radius, then the code $\tilde{\mathcal{C}}$ can also be locally corrected from this radius. Details follow.

The code $\mathcal{C}$. As in Lemma 7.3.16, let b_0 be the absolute constant guaranteed by Lemma 7.3.18, and apply Lemma 7.4.8 with rate $R_0 := \frac{R}{\theta^{-1}(R+\varepsilon/4)}$, proximity parameter $\varepsilon_0 := \varepsilon^2/(4b_0)$, and input list size $\ell_0 := 2^{1/\varepsilon}$. Lemma 7.4.8 then guarantees, for infinitely many N's, the existence of an $\mathbb{F}_2$-vector linear code $\mathcal{C}$ of block length N, alphabet size $N^{o(1)}$, rate R_0, and relative distance $1 - R_0 - \varepsilon_0$, that is $(N^{o(1)}, 1 - R_0 - \varepsilon_0, 0.1, \ell_0, N^{o(1)})$-locally list-recoverable.

The code $\tilde{\mathcal{C}}$. Moreover, by Corollary 7.3.15, each $\boldsymbol{\mathcal{C}}^{(i)}$ is $(h_2^{-1}(1 - R' - \varepsilon), 2^{1/\varepsilon})$-list-decodable with probability $1 - o(1)$, so with probability $1 - \exp(-\Omega(N))$ this property holds for at least $(1 - \varepsilon^2/(4b_0))$-fraction of the $\boldsymbol{\mathcal{C}}^{(i)}$'s. Let $\mathcal{C}_i$, $i \in [n]$ and $\tilde{\mathcal{C}}$ be any linear codes achieving these parameters. Lemma 7.4.13 implies in turn that the code $\tilde{\mathcal{C}}$ is $(N^{o(1)}, \tilde{\rho}, 0.1, N^{o(1)})$-locally list-decodable for

$$\tilde{\rho} = (1 - R_0 - \varepsilon^2/(2b_0)) \cdot h_2^{-1}(1 - R' - \varepsilon) \, .$$

Local correction. Next assume that the local list-decoding radius $\tilde{\rho}$ exceeds the desired local correction radius, i.e.,

$$(1 - R_0 - \varepsilon^2/(2b_0)) \cdot h_2^{-1}(1 - R' - \varepsilon) \geq \frac{h_2^{-1}(1-R) - \varepsilon}{2}\,, \tag{7.5}$$

where $R_0 := \frac{R}{\theta^{-1}(R+\varepsilon/4)}$ and $R' := \theta^{-1}(R + \varepsilon/4)$. It was shown in [Rud07, Section 4.4] that this is indeed the case whenever $R \leq 0.02$ and ε is a sufficiently small constant.

Assuming that (7.5) holds, Lemma 7.4.14 implies that $\tilde{\mathcal{C}}$ is locally correctable from $\frac{h_2^{-1}(1-R)-\varepsilon}{2}$-fraction of errors with query complexity $N^{o(1)}$.

Finally, it can also be verified that running times are as claimed (using brute-force encoding and decoding of inner codes $\mathcal{C}^{(i)}$). □

Concatenation for Local List Recovery: Proof of Lemma 7.4.13

Proof of Lemma 7.4.13. The local list-recovery algorithm $\tilde{A}$ for $\mathcal{C}_{\text{con}}$ will run the local list-recovery algorithm A for $\mathcal{C}$, and answer the queries of A by globally list-recovering the $\mathcal{C}^{(i)}$'s corresponding to the queries of A.

In more detail, on oracle access to a string of input lists $S \in \binom{\Sigma}{\ell'}^{tn}$, the local list-recovery algorithm $\tilde{A}$ for $\mathcal{C}_{\text{con}}$ runs the local list-recovery algorithm A for $\mathcal{C}$, and whenever A asks for some coordinate $i \in [n]$, the algorithm $\tilde{A}$ globally list-recovers the i-th block of S of length t from ρ'-fraction of errors, and feeds the messages corresponding to the first ℓ codewords in the output list as an answer to the query of A. Let $A_1, \ldots, A_L$ be the resulting output local algorithms of A. Then $\tilde{A}$ outputs L local algorithms $\tilde{A}_1, \ldots, \tilde{A}_L$ where each algorithm $\tilde{A}_j$ is defined as follows.

To locally correct the r-th coordinate in the k-th block of $\mathcal{C}_{\text{con}}$ of length t (that is, a coordinate of the form $(k-1)\cdot t + r \in [tn]$ where $1 \leq k \leq n$ and $1 \leq r \leq t$), the algorithm $\tilde{A}_j$ runs the algorithm A_j on input coordinate k. As above, whenever A_j asks for some coordinate $i \in [n]$, the algorithm $\tilde{A}_j$ globally list-recovers the i-th block of S of length t from ρ'-fraction of errors, and feeds the messages corresponding to the first ℓ codewords in the output list as an answer to the query of A_j. Let $a \in \Sigma^{\rho' \cdot t}$ be the output symbol of A_j. Then the algorithm $\tilde{A}_j$ outputs the r-th symbol of $\mathcal{C}^{(k)}(a) \in \Sigma^t$.

Clearly, query complexity, output list size, and running times of $\tilde{A}$ are as claimed. The soundness property also clearly holds. To see that the completeness property holds as well note that if $d(\tilde{c}, S) \leq (\rho - \varepsilon) \cdot \rho'$ for some $\tilde{c} \in \mathcal{C}_{\text{con}}$, then by Markov's inequality for at most $(\rho - \varepsilon)$-fraction of $i \in [n]$ it holds that the i-th block of S of length t is inconsistent with the i-th block of $\tilde{c}$ of length t by more than ρ'-fraction of the coordinates. Moreover, since at least a $(1-\varepsilon)$-fraction of the codes $\mathcal{C}^{(i)}$ are (ρ', ℓ', ℓ)-list-recoverable, list-recovery of the $\mathcal{C}^{(i)}$'s fails on at most a ρ-fraction of the blocks. Completeness then follows since $\mathcal{C}$ is locally list-recoverable from ρ-fraction of errors. □

7.5 Combinatorial Lower Bound on Output List Size

In this section, we first provide a *combinatorial* lower bound on the output list size for list-recovering a high-rate tensor product $C^{\otimes t}$, even in the noiseless setting. In particular, we show that the output list size must be doubly-exponential in t. From this, we are able to deduce certain corollaries demonstrating that our algorithms nearly achieve optimal parameters.

Recall that given vectors $v_1 \in \mathbb{F}^{n_1}, v_2 \in \mathbb{F}^{n_2}, \ldots, v_t \in \mathbb{F}^{n_t}$, their *tensor product* $v_1 \otimes v_2 \otimes \cdots \otimes v_t$ is the t-dimensional box whose value in the $(i_1, i_2, \ldots, i_t) \in [n_1] \times [n_2] \cdots \times [n_t]$ coordinate is given by the product

$$(v_1 \otimes v_2 \otimes \cdots \otimes v_t)_{i_1, i_2, \ldots, i_t} = (v_1)_{i_1} \cdot (v_2)_{i_2} \cdots (v_t)_{i_t} .$$

For the special case of $t = 2$, the tensor product $v \otimes u$ can be thought of as the outer product vu^T.

We also record the following standard fact regarding tensor products.

Fact 7.5.1. *Let $v_1, \ldots, v_{t_1} \in \mathbb{F}^{n_1}$ and $u_1, \ldots, u_{t_2} \in \mathbb{F}^{n_2}$ be sets of linearly independent vectors. Then the collection $\{v_i \otimes u_j \mid i \in [t_1], j \in [t_2]\}$ is linearly independent in $\mathbb{F}^{n_1 \times n_2}$.*

7.5.1 Output List Size for List-Recovering High-Rate Tensor Codes

In this section we prove Theorem 7.1.11 from the introduction, which we restate here for convenience.

Theorem 7.1.11 (Output List Size for List-Recovering High-Rate Tensor Codes)**.** *Let $\varepsilon > 0$. Suppose that $C \leq \mathbb{F}_q^n$ is a linear code of rate $1 - \varepsilon$, and that $C^{\otimes t} \leq \mathbb{F}_q^{n^t}$ is $(0, \ell, L)$-list-recoverable. Then $L \geq \ell^{1/\varepsilon^t}$.*

To prove this theorem, we first prove the following proposition. Informally speaking, we iteratively apply the Singleton bound to conclude that linear codes of rate $1 - \varepsilon$ contain about $1/\varepsilon$ codewords with pairwise disjoint supports. Recall that, for a vector $v \in \mathbb{F}^n$, the *support* of v is $\mathrm{supp}(v) = \{i \in [n] \mid v_i \neq 0\}$.

Proposition 7.5.2. *Let $C \leq \mathbb{F}^n$ be a subspace of dimension k, and let r be a positive integer. Suppose that*

$$\left(1 - \tfrac{1}{r}\right) \cdot n + 1 \leq k . \tag{7.6}$$

Then there exist non-zero vectors $c_1, \ldots, c_r \in C$ such that for all $i \neq j$, $\mathrm{supp}(c_i) \cap \mathrm{supp}(c_j) = \emptyset$.

Proof. Let $m := n - k + 1$, and note that Condition (7.6) is equivalent to

$$(r - 1)m \leq k - 1 .$$

Take a basis for C of the form $(e_1, u_1), \ldots, (e_k, u_k)$, where $e_i \in \mathbb{F}^k$ is the ith standard basis vector, and $u_1, \ldots, u_k \in \mathbb{F}^{n-k}$ are vectors. For $j = 1, \ldots, r - 1$, we can find a

nontrivial linear combination of the vectors $u_{(j-1)\cdot m+1}, \ldots, u_{j\cdot m}$ summing to zero, as they are a (multi-)set of $m = n - k + 1$ vectors lying in $\mathbb{F}^{n-k}$. Taking this linear combination of $(e_{(j-1)\cdot m+1}, u_{(j-1)\cdot m+1}), \ldots, (e_{j\cdot m}, u_{j\cdot m})$, we obtain a nonzero vector whose support is contained in the interval $\{(j-1)\cdot m+1, \ldots, j\cdot m\}$; denote this vector by c_j. In this manner, we obtain $r-1$ nonzero vectors $c_1, \ldots, c_{r-1} \in C$ with pairwise disjoint support. Finally, we may add the vector $c_r := (e_k, u_k)$ to this collection, yielding r vectors, as desired. □

Next we prove Theorem 7.1.11, based on the above proposition.

Proof of Theorem 7.1.11. Let $r := 1/\varepsilon$, and recall wish to come up with ℓ^{r^t} codewords in $\mathcal{C}^{\otimes t}$ that are contained in the output list for appropriately chosen input lists.

In order to accomplish this, we first use Proposition 7.5.2 to obtain a subset $\mathcal{C}' \subseteq \mathcal{C}$ of r nonzero codewords with pairwise disjoint support. We then consider the subset $\mathcal{C}'' \subseteq \mathcal{C}^{\otimes t}$ containing all tensor products $c_1 \otimes c_2 \otimes \cdots \otimes c_t$ of t (not necessarily distinct) codewords $c_1, \ldots, c_t \in \mathcal{C}'$, and our main observation is that all these r^t tensor products are also nonzero with pairwise disjoint support. Finally, we let $B \subseteq \mathbb{F}$ be an arbitrary subset of size ℓ, and consider the subset $\bar{\mathcal{C}} \subseteq \mathcal{C}^{\otimes t}$ containing all linear combinations of codewords in $\mathcal{C}''$ with coefficients in B. Since all codewords in $\mathcal{C}''$ are nonzero with pairwise disjoint support, they are in particular linearly independent, so the set $\bar{\mathcal{C}}$ contains ℓ^{r^t} distinct codewords in $\mathcal{C}^{\otimes t}$.

Moreover, since codewords in $\mathcal{C}''$ have pairwise disjoint support, for each coordinate $(i_1, \ldots, i_t) \in [n]^t$, there is at most one codeword $c \in \mathcal{C}''$ for which $c_{i_1,\ldots,i_t}$ is nonzero.

Therefore this is the only term which can contribute nontrivially to the value in the $(i_1, \ldots, i_t)$ coordinate of a codeword in $\bar{\mathcal{C}}$. So we can let the corresponding input list $S_{i_1,\ldots,i_t}$ contain all the $\beta \cdot c_{i_1,\ldots,i_t}$ for elements $\beta \in B$. Details follow.

The set $\mathcal{C}'$. Since $\mathcal{C}$ has rate $1-\varepsilon$, it has dimension $k = (1-\varepsilon)n$, and so Proposition 7.5.2 guarantees the existence of a subset $\mathcal{C}' \subseteq \mathcal{C}$ of $r = 1/\varepsilon$ nonzero codewords with pairwise disjoint support.

The set $\mathcal{C}''$. Next we let

$$\mathcal{C}'' := \left\{ c_1 \otimes c_2 \otimes \cdots \otimes c_t \mid c_1, c_2, \ldots, c_t \in \mathcal{C}' \right\}$$

be the subset of $\mathcal{C}^{\otimes t}$ containing all tensor products of t (not necessarily distinct) codewords in $\mathcal{C}'$. Since all codewords in $\mathcal{C}'$ are nonzero, their t-wise tensor products are nonzero as well.

To see that all codewords in $\mathcal{C}''$ have pairwise disjoint support, suppose that $c = c_1 \otimes c_2 \otimes \ldots \otimes c_t \in \mathcal{C}''$, and $(i_1, i_2, \ldots, i_t) \in \mathrm{supp}(c)$. Then

$$0 \neq c_{i_1,i_2,\ldots,i_t} = (c_1)_{i_1} \cdot (c_2)_{i_2} \cdots (c_t)_{i_t} ,$$

so we must have that $(c_1)_{i_1}, (c_2)_{i_2}, \ldots, (c_t)_{i_t}$ are all nonzero. We conclude that

$$\mathrm{supp}(c) \subseteq \mathrm{supp}(c_1) \times \mathrm{supp}(c_2) \times \cdots \times \mathrm{supp}(c_t).$$

Now, suppose that $c = c_1 \otimes \ldots \otimes c_t$, $c' = c'_1 \otimes \ldots \otimes c'_t$ are a pair of codewords in $\mathcal{C}''$ with $c_j \neq c'_j$ for some $j \in [t]$. Since all codewords in $\mathcal{C}'$ have pairwise disjoint support it must hold that $\mathrm{supp}(c_j) \cap \mathrm{supp}(c'_j) = \emptyset$, and we conclude that $\mathrm{supp}(c) \cap \mathrm{supp}(c') = \emptyset$.

The set $\bar{\mathcal{C}}$. Now, let $B \subseteq \mathbb{F}_q$ be an arbitrary subset of size ℓ, and let

$$\bar{C} := \left\{ \sum_{c \in C''} \beta_c \cdot c \;\middle|\; \beta_c \in B \text{ for all } c \in \mathcal{C}'' \right\}$$

be the subset of $\mathcal{C}^{\otimes t}$ containing all linear combinations of codewords in $\mathcal{C}''$ with coefficients in B. Since all codewords in $\mathcal{C}''$ are nonzero with pairwise disjoint support, they are in particular linearly independent in $\mathbb{F}^{n^t}$,[7] so the set $\bar{\mathcal{C}}$ contains ℓ^{r^t} distinct codewords in $\mathcal{C}^{\otimes t}$.

Input lists. Finally, we wish to define input lists $S_{i_1,\ldots,i_t}$ for any coordinate $(i_1, \ldots, i_t) \in [n]^t$ so that for any codeword $c \in \bar{\mathcal{C}}$, and for any coordinate $(i_1, \ldots, i_t) \in [n]^t$, it holds that $c_{i_1,\ldots,i_t} \in S_{i_1,\ldots,i_t}$.

To this end, we observe that since codewords in $\mathcal{C}''$ have pairwise disjoint support, for each coordinate $(i_1, \ldots, i_t) \in [n]^t$, there is at most one codeword $c \in \mathcal{C}''$ for which $c_{i_1,\ldots,i_t}$ is nonzero. Therefore this is the only term which can contribute nontrivially to the value in the $(i_t, \ldots, i_t)$ coordinate of a codeword in $\bar{\mathcal{C}}$. So we can define the corresponding input list $S_{i_1,\ldots,i_t}$ as

$$S_{i_1,\ldots,i_t} := \{\beta \cdot c_{i_1,\ldots,i_t} : \beta \in B\}$$

if such a codeword c exists, and as $S_{i_1,\ldots,i_t} = \{0\}$ otherwise. Note that each set $S_{i_1,\ldots,i_t}$ has size at most ℓ, and that they satisfy the required property.

This yields a set of ℓ^{r^t} codewords from $\mathcal{C}^{\otimes t}$ that are contained in the output list for the input list tuple S defined above, proving the theorem. □

7.5.2 Concrete Lower Bound on Output List Size

In this section, we demonstrate a setting of parameters that yields Corollary 7.1.12 from the introduction, restated below.

[7] This also follows from the fact that all codewords in $\mathcal{C}''$ are linearly independent together with Fact 7.5.1.

Corollary 7.1.12. *For any $\delta > 0$ and $\ell > 1$ there exists $L > 1$ such that the following holds for any sufficiently large n. There exists a linear code $\mathcal{C} \leq \mathbb{F}_q^n$ of relative distance δ that is $(\Omega(\delta), \ell, L)$-list-recoverable, but $\mathcal{C}^{\otimes t} \leq \mathbb{F}_q^{n^t}$ is only $(0, \ell, L')$-list-recoverable for $L' \geq \exp((2\delta)^{-(t-3/2)} \cdot \sqrt{\log L})$.*

We use the following result on the list-recoverability of random linear codes from [RW18].

Theorem 7.5.3 ([RW18], Corollary 3.3)**.** *There exists an absolute constant b_0 so that the following holds. For any $\varepsilon > 0$, $\ell \geq 1$, and a prime power $q \geq \ell^{b_0/\varepsilon}$, a random linear code $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ of rate $1 - \varepsilon$ is $(\Omega(\varepsilon), \ell, L)$-list recoverable for*

$$L \leq \left(\frac{q\ell}{\varepsilon}\right)^{(\log \ell)/\varepsilon} \cdot \exp\left(\frac{\log^2 \ell}{\varepsilon^3}\right)$$

with probability $1 - \exp(-\Omega(n))$.

Proof of Corollary 7.1.12. Let $\mathcal{C} \leq \mathbb{F}_q^n$ be a linear code as promised by Theorem 7.5.3 of rate $1 - 2\delta$ and $q = \ell^{O(1/\delta)}$ that is $(\Omega(\delta), \ell, L)$-list-recoverable for $L = \exp((\log^2 \ell)/\delta^3)$, or equivalently, $\ell = \exp(\delta^{3/2} \cdot \sqrt{\log L})$. By Corollary 7.3.14, we may further assume that the code $\mathcal{C}$ has relative distance at least δ. Now, by Theorem 7.1.11 we must have $L' \geq \ell^{(2\delta)^{-t}} = \exp((2\delta)^{-(t-3/2)} \cdot \sqrt{\log L})$. □

7.5.3 Lower Bound for Local List-Recovery

We now prove Corollary 7.1.13 from the introduction, restated below.

Corollary 7.1.13. *For any $\delta > 0$ and sufficiently large n there exists a linear code $\mathcal{C} \leq \mathbb{F}_q^n$ of relative distance δ such that the following holds. Suppose that $\mathcal{C}^{\otimes t} \leq \mathbb{F}_q^N$ is $(\frac{1}{N}, 2, L)$-locally list-recoverable with query complexity Q. Then $Q \cdot L \geq N^{\Omega_\delta(1/\log\log N)}$.*

We first recall Lemma 7.4.14 (restated below in a shorter form) which says that a locally list-decodable (and in particular locally list-recoverable) code with output list size L and query complexity Q is also locally correctable with query complexity roughly $Q \cdot L$.

Lemma 7.5.4. *Suppose that $\mathcal{C} \subseteq \Sigma^n$ is a code of relative distance δ that is $(Q, \alpha, 0.1, L)$-locally list-decodable for $\alpha < \delta/2$. Then $\mathcal{C}$ is $\left(O\big(Q \cdot L \cdot \frac{\log^2 n}{(\delta/2-\alpha)^2}\big), \alpha\right)$-locally correctable.*

So to prove Corollary 7.1.13, it is enough to show a lower bound on the query complexity for local correcting $\mathcal{C}^{\otimes t}$, assuming that the output list for list-recovering $\mathcal{C}^{\otimes t}$ is small. To show such a lower bound, we first observe that for any linear code $\mathcal{C}$, the (absolute) distance of $\mathcal{C}^\perp$ is a lower bound on the query complexity for locally correcting $\mathcal{C}$.

Lemma 7.5.5. *Suppose that $\mathcal{C} \leq \mathbb{F}^n$ is a linear code that is $(Q, \frac{1}{n})$-locally correctable. Then $Q \geq \Delta(C^\perp) - 2$.*

We prove Lemma 7.5.5 in Section 7.5.4. To apply this lemma to $\mathcal{C}^{\otimes t}$ we further observe that the tensor product preserves the dual distance of the base code.

Lemma 7.5.6. *Suppose that $\mathcal{C}_1 \leq \mathbb{F}^{n_1}, \mathcal{C}_2 \leq \mathbb{F}^{n_2}$ are linear codes, and that $\mathcal{C}_1^\perp, \mathcal{C}_2^\perp$ have absolute distances Δ_1, Δ_2, respectively. Then $(\mathcal{C}_1 \otimes \mathcal{C}_2)^\perp$ has absolute distance $\min\{\Delta_1, \Delta_2\}$. In particular, if $\mathcal{C} \leq \mathbb{F}^n$ is a linear code and $\mathcal{C}^\perp$ has absolute distance Δ, then $(\mathcal{C}^{\otimes t})^\perp$ has absolute distance Δ for any $t \geq 1$.*

The proof of Lemma 7.5.6 is provided in Section 7.5.5. We now proceed to the proof of Corollary 7.1.13.

Proof of Corollary 7.1.13. Let $\boldsymbol{\mathcal{C}} \leq \mathbb{F}_q^n$ be a random linear code of rate $1 - 2\delta$. By Corollary 7.3.14, for sufficiently large q, the code $\boldsymbol{\mathcal{C}}$ will have relative distance at least δ with high probability. Moreover, since $\boldsymbol{\mathcal{C}}^\perp$ has rate 2δ, by the same corollary we also have that $\boldsymbol{\mathcal{C}}^\perp$ has relative distance at least $1 - 3\delta$ with high probability. We conclude for any sufficiently large n the existence of a linear code $\mathcal{C} \leq \mathbb{F}_q^n$ of rate $1 - 2\delta$ and relative distance at least δ such that $\mathcal{C}^\perp$ has relative distance at least $1 - 3\delta$.

Next observe that for the code $\mathcal{C}^{\otimes t}$ to be $(Q, \frac{1}{N}, 0.1, 2, L)$-locally list-recoverable, it in particular must be $(0, 2, L)$-list-recoverable, so the lower bound from Theorem 7.1.11 implies that $L \geq 2^{1/(2\delta)^t}$. Now, if $2^{1/(2\delta)^t} \geq N$ then we have that $Q \cdot L \geq 2^{1/(2\delta)^t} \geq N$, and we are done. So we may assume that $2^{1/(2\delta)^t} < N$ which implies in turn that $t = O_\delta(\log \log N)$ and $n = N^{1/t} = N^{\Omega_\delta(1/\log \log N)}$.

Moreover, as we have assumed we have a $(Q, \frac{1}{N}, 0.1, 2, L)$-local list-recovery algorithm for $\mathcal{C}^{\otimes t}$, we also have a $(Q, \frac{1}{N}, 0.1, L)$-local list-decoding algorithm for $C^{\otimes t}$. Lemma 7.5.4 then promises that we have a $(O(Q \cdot L \cdot \frac{\log^2 N}{(\delta^t/2 - 1/N)^2}), \frac{1}{N})$-local correction algorithm for $\mathcal{C}^{\otimes t}$.

Now, by Lemma 7.5.6 we have that $(\mathcal{C}^{\otimes t})^\perp$ has (absolute) distance at least $(1 - 3\delta)n$, and consequently Lemma 7.5.5 implies that

$$O\left(Q \cdot L \cdot \frac{\log^2 N}{(\delta^t/2 - \frac{1}{N})^2}\right) \geq (1 - 3\delta)n - 2 = N^{\Omega_\delta(1/\log \log N)}.$$

This implies $Q \cdot L \geq N^{\Omega_\delta(1/\log \log N)}$, as desired. □

7.5.4 Dual Distance is a Lower Bound on Query Complexity: Proof of Lemma 7.5.5

First, we recall the standard fact that (absolute) dual distance Δ implies that the uniform distribution over the code is $(\Delta - 1)$-wise independent.

Fact 7.5.7 ([ABI86])**.** *Let $\mathcal{C} \leq \mathbb{F}_q^n$ be a linear code, and suppose that $\mathcal{C}^\perp$ has (absolute) distance Δ. Then for all $1 \leq i_1 < \cdots < i_s \leq n$ with $s < \Delta$, and all $a_1, \ldots, a_s \in \mathbb{F}_q$,*

$$\mathop{\mathbb{P}}_{\boldsymbol{c} \sim \mathcal{C}}\left(\boldsymbol{c}_{i_1} = a_1 \wedge \cdots \wedge \boldsymbol{c}_{i_s} = a_s\right) = \frac{1}{q^s}.$$

Let $\Delta := \Delta(C^\perp)$. Making use of Yao's principle, it suffices to show a distribution μ over vectors at absolute distance at most 1 from $\mathcal{C}$ such that the following holds. For any

deterministic algorithm making at most $\Delta - 2$ queries to its input $\boldsymbol{w}$ sampled according to μ, the probability that it correctly computes $\boldsymbol{c}_1$ is at most $1/3$, where $\boldsymbol{c}$ is the unique codeword in $\mathcal{C}$ at absolute distance at most 1 from $\boldsymbol{w}$. We will in fact show that no deterministic query algorithm can correctly compute $\boldsymbol{c}_1$ with probability greater than $1/q$.

Let μ denote the distribution that samples $\boldsymbol{c} \in \mathcal{C}$ uniformly at random and then sets $\boldsymbol{c}_1 = 0$. Let A be a deterministic algorithm making at most $\Delta - 2$ queries, and let $j_1, \ldots, j_s \in [n]$ denote the queries made by A, where we assume $s \leq \Delta - 2$. Note that querying $\boldsymbol{w}_1$ does not help A, as it will always read 0. Hence, without loss of generality, $1 \notin \{j_1, \ldots, j_s\}$.

Now, by Fact 7.5.7 and Bayes' rule, for any $b_1, \ldots, b_s, a \in \mathbb{F}_q$, if $\boldsymbol{c} \sim \mathcal{C}$ is distributed uniformly,

$$\mathbb{P}\left(\boldsymbol{c}_1 = a | \boldsymbol{c}_{j_1} = b_1, \ldots, \boldsymbol{c}_{j_s} = b_s\right) = \frac{\mathbb{P}\left(\boldsymbol{c}_1 = a, \boldsymbol{c}_{j_1} = b_1, \ldots, \boldsymbol{c}_{j_s} = b_s\right)}{\mathbb{P}\left(\boldsymbol{c}_{j_1} = b_1, \ldots, \boldsymbol{c}_{j_s} = b_s\right)} = \frac{q^{-(s+1)}}{q^{-s}} = \frac{1}{q} \,.$$

Additionally, observe that the distribution of the tuple $(\boldsymbol{c}_{j_1}, \ldots, \boldsymbol{c}_{j_s})$ is the same if $\boldsymbol{c}$ is a uniformly random codeword from $\mathcal{C}$ or if it is sampled according to μ.

Hence, if we think of the query algorithm as implementing a (deterministic) function $g : \mathbb{F}_q^s \to \mathbb{F}_q$ from the responses to its queries to its guess for $\boldsymbol{c}_1$, regardless of the responses $b_1, \ldots, b_s$ to the queries, we have

$$\mathop{\mathbb{P}}_{\boldsymbol{w} \sim \mu}\left(\boldsymbol{c}_1 = g(b_1, \ldots, b_s) | \boldsymbol{w}_{j_1} = b_1, \ldots, \boldsymbol{w}_{j_s} = b_s\right) = \frac{1}{q} \,,$$

where $\boldsymbol{c}$ is the unique codeword in $\mathcal{C}$ for which $d(\boldsymbol{c}, \boldsymbol{w}) \leq \frac{1}{n}$. That is, the query algorithm will not be able to guess $\boldsymbol{c}_1$ with probability greater than $1/q$, as claimed.

7.5.5 Tensor Product Preserves Dual Distance: Proof of Lemma 7.5.6

First note that we clearly have that $\Delta((\mathcal{C}_1 \otimes \mathcal{C}_2)^\perp) \leq \min\{\Delta_1, \Delta_2\}$: for example, the matrix whose first column is a vector from $\mathcal{C}_1^\perp$ of weight Δ_1 and all other columns are 0 gives a matrix in $(\mathcal{C}_1 \otimes \mathcal{C}_2)^\perp$ of weight Δ_1, and similarly a matrix in $(\mathcal{C}_1 \otimes \mathcal{C}_2)^\perp$ of weight Δ_2 can be constructed. We now establish the opposite inequality of $\Delta((C_1 \otimes C_2)^\perp) \geq \min\{\Delta_1, \Delta_2\}$.

It is well-known (and not hard to show) that the (absolute) distance of a code $\mathcal{C}$ is the minimum number of linearly dependent columns in a parity-check matrix for $\mathcal{C}$. Furthermore, if G is a generating matrix for $\mathcal{C}$ then G^T is a parity-check matrix for $\mathcal{C}^\perp$. We conclude that the distance of $\mathcal{C}^\perp$ is the minimum number of linearly dependent rows in a generating matrix for $\mathcal{C}$.

Let G_1, G_2 be generating matrices for $\mathcal{C}_1, \mathcal{C}_2$, respectively, and note that by the above, any collection of $t_1 < \Delta_1, t_2 < \Delta_2$ rows of G_1, G_2, respectively, are linearly independent. Next recall that $G_1 \otimes G_2$ is a generating matrix for $\mathcal{C}_1 \otimes \mathcal{C}_2$, and so it suffices to show that for any $t < \min\{\Delta_1, \Delta_2\}$, any collection of t rows of $G_1 \otimes G_2$ are linearly independent.

Let $u_1, u_2, \ldots, u_{n_1}$ and $v_1, v_2, \ldots, v_{n_2}$ denote the rows of G_1, G_2, respectively, and note that each row in $G_1 \otimes G_2$ is of the form $u_i \otimes v_j$ for some $i \in [n_1], j \in [n_2]$. Fix $t < \min\{\Delta_1, \Delta_2\}$, and suppose that $u_{i_1} \otimes v_{j_1}, u_{i_2} \otimes v_{j_2}, \ldots, u_{i_t} \otimes v_{j_t}$ is a collection of t rows of $G_1 \otimes G_2$. Then by the above we have that both collections $u_{i_1}, u_{i_2}, \ldots, u_{i_t}$ and $v_{j_1}, v_{j_2}, \ldots, v_{j_t}$ are linearly independent (ignoring duplications). Fact 7.5.1 implies in turn that the collection $u_{i_1} \otimes v_{j_1}, u_{i_2} \otimes v_{j_2}, \ldots, u_{i_t} \otimes v_{j_t}$ is also linearly independent. This concludes the proof of the lemma.

Chapter 8

Dimension Expanders: An Application of List-Decodable Codes

In this chapter, we show that techniques developed in the context of list-decoding rank metric codes can be used to provide a construction of dimension expanders, which are a linear-algebraic analog of expander graphs.

8.1 Introduction

The field of *pseudorandomness* is concerned with efficiently constructing objects that share desirable properties with random objects while using no or little randomness. The ideas developed in pseudorandomness have found broad applications in areas such as complexity theory, derandomizaton, coding theory, cryptography, high-dimensional geometry, graph theory, and additive combinatorics. Due to much effort on the part of many researchers, nontrivial constructions of expander graphs, randomness extractors and condensers, Ramsey graphs, list-decodable codes,[1] compressed sensing matrices, Euclidean sections, and pseudorandom generators and functions have been presented. Interestingly, while these problems may appear superficially to be unrelated, many of the techniques developed in one context have been useful in others, and the deep connections uncovered between these pseudorandom objects have led to a unified theory of "Boolean pseudoranomness". See for instance this survey by Vadhan [Vad12] for more discussion of this phenomenon.

More recently, there is a developing theory of "algebraic pseudorandomness," wherein the pseudorandom objects of interest now have "algebraic structure" rather than a purely combinatorial structure. In these scenarios, instead of studying the size of subsets or min-entropy, we consider the dimension of subspaces. Many analogs of classical pseudorandom objects have been defined, such as dimension expanders, subspace-

[1] Indeed, the star of this thesis can be naturally considered a pseudorandom object; recall that the best known constructions of list-decodable codes are uniformly random codes.

evasive sets, subspace designs, rank-preserving condensers, and list-decodable rank metric codes. Beyond being interesting in their own rights, these algebraic pseudorandom objects have found many applications: for example, subspace-evasive sets have been used in the construction of Ramsey graphs [PR04] and list-decodable codes [GX12; GW13]; subspace designs have been used to list-decode codes over the Hamming metric and the rank metric [GW14; GWX16]; and rank-preserving condensers have been used in affine extractors [GR08a] and polynomial identity testing [KS11; FS12].

In this chapter, we focus upon providing explicit constructions of dimension expanders over finite fields. A *dimension expander* is a collection of d linear maps $\Gamma_j : \mathbb{F}^n \to \mathbb{F}^n$ such that, for any subspace $U \subseteq \mathbb{F}^n$ of sufficiently small dimension, the sum of the images of U under all the maps $\Gamma_1(U) + \cdots + \Gamma_d(U)$ has dimension which is a constant factor larger than $\dim U$. As suggested by their name, dimension expanders may be viewed as a linear-algebraic analog of expander graphs. Indeed, imagine creating a graph with vertex set $\mathbb{F}^n$, and then adding an edge from a vertex $u \in \mathbb{F}^n$ to the vertices $\Gamma_j(u)$.[2] Alternatively, consider the bipartite graph with left and right partition given by $\mathbb{F}^n$, and attach a vertex $u \in \mathbb{F}^n$ in the left partition to $\Gamma_j(u)$ in the right partition for each j. For this reason, d is referred to as the *degree* of the dimension expander. The property of being a dimension expander then says that, given any (sufficiently small) *subspace*, the span of the neighborhood will have appreciably larger dimension. Indeed, we use the notation Γ_j for the linear maps in analogy with the "neighborhood function" of a graph. Just as with expander graphs, we seek dimension expanders with constant degree, and moreover we would like to be able to expand subspaces of dimension at most ηn by a multiplicative factor of β, where $\eta = \Omega(1)$ and $\beta = 1 + \Omega(1)$. We refer to such an object as an (η, β)-dimension expander. If $\beta = \Omega(d)$, we deem the dimension expander *degree-proportional*. If moreover $\beta = (1 - \varepsilon)d$, we deem the dimension expander *lossless*. Via a probabilistic argument, it is a simple exercise to show that constant-degree lossless dimension expanders exist over every field.

Finally, we indicate that *unbalanced* bipartite expander graphs play a key role in constructions of extractors and other Boolean pseudorandom objects. In this scenario, the left partition is significantly larger than the right partition, but we still have that sufficiently small subsets U of the left partition expand significantly, with $(1 - \varepsilon)d|U|$ neighbors in the right partition in the lossless case. Such unbalanced expanders are closely related to *randomness condensers*, which preserve all or most of the min-entropy of a source while compressing its length. The improved min-entropy *rate* at the output makes subsequent *extraction* of nearly-uniform random bits easier. Indeed, the extractors in [GUV09] were obtained via this paradigm, once lossless expanders based on list-decodable codes were constructed. Inspired by this, we consider the challenge of constructing *unbalanced* dimension expanders: for N and n not necessarily equal, we would like a collection of maps $\Gamma_1, \ldots, \Gamma_d : \mathbb{F}^N \to \mathbb{F}^n$ that expand sufficiently small subspaces by a factor of $\approx d$. We quantify the "unbalancedness" of the dimension expander by $b = \frac{N}{n}$, and we refer to it as a *b-unbalanced dimension expander in $\mathbb{F}^n$*. Again, if the ex-

[2] In general, this yields a directed graph. However, we may assume the maps Γ_j are invertible and then add the maps Γ_j^{-1} to the collection, which makes the graph undirected.

pansion factor is $\Omega(d)$ we deem the unbalanced dimension expander *degree-proportional*, while if the expansion factor is $(1-\varepsilon)d$ we deem it *lossless*.

8.1.1 Our results

We provide various explicit constructions of dimension expanders. More precisely, we have a family of sets of matrices $\{\{\Gamma_1^{(n_k)}, \ldots, \Gamma_d^{(n_k)}\}\}_{k\in\mathbb{N}}$ for an infinite sequence of integers $n_1 < n_2 < \cdots$, where each $\Gamma_j^{(n_k)}$ is an $n_k \times n_k$ matrix (or $n_k \times bn_k$ matrix in the case of b-unbalanced expanders). The family is deemed *explicit* if there is an algorithm outputting the list of matrices $\Gamma_1^{(n_k)}, \ldots, \Gamma_d^{(n_k)}$ in $\mathrm{poly}(n_k)$ field operations.

First of all, we provide the first explicit construction of a lossless dimension expander. Moreover we emphasize that the η parameter is optimal as well, as one cannot hope to expand subspaces of dimension more than $\frac{n}{d}$ by a factor of $\approx d$.

Theorem 8.1.1 (Informal Statement; cf. Theorem 8.5.2)**.** *For all $\varepsilon > 0$ constant, there exists an integer $d = d(\varepsilon)$ sufficiently large such that there is an explicit family of $(\frac{1-\varepsilon}{d}, (1-\varepsilon)d)$-dimension expanders of degree d over $\mathbb{F}^n$ when $|\mathbb{F}| \geq \Omega(n)$.*

As a final remark, we comment that the dependence of d on ε is quite modest: we obtain $d = O(1/\varepsilon^3)$. This compares favorably with the degree achievable by a randomized construction, which guarantees $d = O(1/\varepsilon^2)$; see Proposition 8.2.5 and the subsequent discussion.

The main drawback of the above result is the constraint on the field size. Our next result allows for smaller field sizes, but we are only able to guarantee degree-proportional expansion. We remark that prior to this work, no explicit constructions of degree-proportional dimension expanders were known.

Theorem 8.1.2 (Informal Statement; cf. Theorem 8.5.1)**.** *For all $\delta > 0$ constant, there exists an integer $d = d(\delta)$ sufficiently large such that there is an explicit family of $\left(\Omega\left(\frac{1}{\delta d}\right), \Omega(\delta d)\right)$-dimension expanders of degree d over $\mathbb{F}^n$ when $|\mathbb{F}| \geq n^\delta$.*

Moreover, our paradigm is flexible enough to allow for the construction of unbalanced dimension expanders. We remark that while the results of Forbes and Guruswami [FG15] could be adapted to obtain nontrivial constructions of unbalanced expanders, our work is the first to explicitly state this. Furthermore, our work is the first to achieve lossless expansion, or even degree-proportionality. First, we provide a construction of a lossless unbalanced dimension expander, again over fields of linear size.

Theorem 8.1.3 (Informal Statement; cf. Theorem 8.6.7)**.** *For all $\varepsilon > 0$ and integer $b \geq 1$, there exists an integer $d = d(\varepsilon, b)$ sufficiently large such that there is an explicit family of b-unbalanced $(\frac{1-\varepsilon}{db}, (1-\varepsilon)d)$-dimension expanders of degree d over $\mathbb{F}^n$ when $|\mathbb{F}| \geq \Omega(n)$.*

Again, the dependence of d is $O(b/\varepsilon^3)$, which is only a factor of $1/\varepsilon$ larger than the randomized construction (Proposition 8.2.5). This result is again complemented by a construction of degree-proportional unbalanced dimension expanders over fields of arbitrarily small polynomial size.

Theorem 8.1.4 (Informal Statement; cf Theorem 8.6.6)**.** *For all $\delta > 0$ and integer $b \geq 1$, there exists an integer $d = d(\delta, b)$ sufficiently large such that there is an explicit family of b-unbalanced $\left(\Omega\left(\frac{1}{\delta b d}\right), \Omega(\delta d)\right)$-dimension expanders of degree d over $\mathbb{F}^n$ when $|\mathbb{F}| \geq n^{\delta}$.*

Our final contribution is to define *subspace evasive subspaces*, and observe that they yield degree-proportional dimension expanders. Informally, a subspace evasive subspace H is an $\mathbb{F}_q$-subspace that has small intersection with any subspace of bounded dimension defined over an *extension field*. To properly define this notion, it is best to identify $\mathbb{F}_q^n$ with $\mathbb{F}_{q^n}$, and then consider $\mathbb{F}_{q^d}$-subspaces of $\mathbb{F}_{q^n}$ for $d|n$. The subspace $H \subseteq \mathbb{F}_{q^n}$ is then said to be (s, A, d)-subspace evasive if for every $\mathbb{F}_{q^d}$-linear subspace $W \subseteq \mathbb{F}_{q^n}$ of dimension s, $\dim_{\mathbb{F}_q}(H \cap W) \leq As$.

Proposition 8.1.5 (Informal Statement; cf. Proposition 8.7.4)**.** *There exists a $(s, 1+O(1/k), d)$-subspace evasive subspace for all $s \leq O(n/d)$. Moreover, given an explicit subspace evasive subspace achieving these parameters, there is an explicit construction of a degree-proportional dimension expander.*

8.1.2 Interlude: Rank Metric Codes

Before describing our approach in detail, we take a brief detour to discuss rank metric codes [Gab85]. While we introduced these objects in Chapter 5, we did not discuss explicit constructions. As our dimension expanders are inspired by recent explicit constructions of list-decodable rank metric codes, the time is ripe for their introduction.

Just as a discussion of list-decodable codes over the Hamming metric must start with Reed-Solomon codes, a discussion a list-decodable codes over the rank metric much start with Gabidulin codes.

Example 8.1.6 (Gabidulin Codes, [Gab85])**.** In analogy with Reed-Solomon codes which are defined by evaluations of low degree polynomials, Gabidulin codes are defined by evaluations of low degree *linearized polynomials*. A *q-linearized polynomial* is any polynomial of the form

$$f(X) = \sum_{i=0}^{k-1} f_i X^{q^i}$$

where the coefficients $f_i \in \mathbb{F}_{q^m}$. In justification of their name, observe that the identities $(\alpha + \beta)^q = \alpha^q + \beta^q$[3] for all $\alpha, \beta \in \mathbb{F}_{q^m}$ and $a^q = a$ for all $a \in \mathbb{F}_q$ demonstrate that a linearized polynomial, when viewed as a map from $\mathbb{F}_{q^m} \to \mathbb{F}_{q^m}$, is linear over $\mathbb{F}_q$ in the sense that $f(a\alpha + b\beta) = af(\alpha) + bf(\beta)$ for all $a, b \in \mathbb{F}_q$ and $\alpha, \beta \in \mathbb{F}_{q^m}$. The maximum i for which $f_i \neq 0$ is the *q-degree* of f, and we let $\mathbb{F}_{q^m}[X; (\cdot)^q]_{<k}$ denote the space of all linearized polynomials over $\mathbb{F}_{q^m}$ of q-degree less than k, which narturally forms a k-dimensional vector space over $\mathbb{F}_{q^n}$.

Thus, as $f \in \mathbb{F}_{q^m}[X; (\cdot)^q]_{<k}$ can be viewed as a linear map, we can ask about the dimension of its kernel. Note that as f is a degree q^{k-1} polynomial, it can have at most q^{k-1} roots, and thus $\dim(\ker f) \leq k - 1$.

[3]Which is informally referred to as the "freshman's dream".

Now, for some $n \leq m$, let $\alpha_1, \ldots, \alpha_n \in \mathbb{F}_{q^m}$ be linearly indepdent over $\mathbb{F}_q$. Next, let $\omega_1, \ldots, \omega_m$ be a basis for $\mathbb{F}_{q^m}/\mathbb{F}_q$, and for $\alpha \in \mathbb{F}_{q^m}$ let $\omega(\alpha) \in \mathbb{F}_q^m$ denote the vector $(a_1, \ldots, a_m)^\top$, where $\alpha = \sum_{i=1}^m a_i \omega_i$. We extend this notation to vectors $v \in \mathbb{F}_{q^m}^n$, so that $\omega(v) \in \mathbb{F}_q^{m \times n}$ is a matrix. Finally, we define

$$\mathrm{Gab}[n, m, k, q] := \{(\omega(f(\alpha_1), f(\alpha_2), \ldots, f(\alpha_n)) : f \in \mathbb{F}_{q^m}[X; (\cdot)^q]_{<k}\} .$$

(We remark that the resulting code parameters do not depend on the specific choice of sets $\{\alpha_i\}$ and $\{\omega_i\}$, so we omit their dependence in the above notation.) Now, since every nonzero $f \in \mathbb{F}_{q^m}[X; (\cdot)^q]_{<k}$ satisfies $\dim(\ker f) \leq k-1$, any matrix in $\mathrm{Gab}[n, m, k, q]$ has rank at least $n - k + 1$. Moreover, the rate of $\mathrm{Gab}[n, m, k, q]$ is $\frac{k}{n}$. Thus, we conclude that Gabidulin codes achieve the rank metric Singleton bound (Theorem 5.1.1).

Finally, we remark that Gabidulin Codes may naturally be viewed as a subset of $\mathbb{F}_{q^m}^n$. Moreover, they are in fact linear over the extension field $\mathbb{F}_{q^m}$, in the sense that the set $\mathrm{Gab}[n, m, k, q]$ is closed under multiplication by scalars from $\mathbb{F}_{q^m}$.

Thus, Gabidulin codes are essentially an optimal rank metric code: they do not even suffer from the alphabet restriction that plagues RS codes. However, as mentioned in Section 5.1, when it comes to list-decoding Gabidulin codes most results are negative. In response to this state-of-affairs, Guruswami, Wang and Xing [GWX16] carefully constructed subcodes of Gabidulin codes and showed that they are list-decodable up to radius $1 - R - \varepsilon$ with lists of size $q^{O(1/\varepsilon^4)}$. The code may also readily be seen to be "list-recoverable" in the following sense: given vector spaces $V_1, \ldots, V_n \subseteq \mathbb{F}^m$ of bounded dimension, the number of matrices in $A \in \mathcal{C}$ with $A_{*,i} \in V_i$ for all $i \in [n]$ is bounded, where $A_{*,i}$ denotes the i-th column of A. In brief, the authors use a pseudorandom object called a *subspace design* to prune the space of linearized polynomials in $\mathbb{F}_{q^m}[X; (\cdot)^q]_{<k}$ that they evaluate. Our construction of dimension expanders, which we outline next, is very much inspired by the rank metric code of [GWX16].

8.1.3 Our approach

In the case of Boolean pseudorandomness, not long after the construction of Parvaresh-Vardy codes and folded Reed-Solomon codes [PV05; GR08b], the techniques used to prove list-recoverability of these codes were adapted to show expansion of unbalanced bipartite expanders built from these codes [GUV09]. Our approach is strongly inspired by the connection between list-recovery and expansion that drives [GUV09] and its instantiation with algebraic codes shown to achieve optimal redundancy for list-recovery. Indeed, our methodology can be viewed as an adaption of the GUV approach to the "linearized world". Various challenges arise in attempting to adapt the GUV framework to the setting of Gabidulin-like codes. For instance, we are no longer able to "append the seed" (in our context, the field element α_j) to the output of the neighborhood functions as is done in [GUV09], as that will prevent the maps from being linear.[4] More signif-

[4]One could instead try tensoring the output with the seed, but it is unclear to us how to make this approach work without significantly degrading the expansion factor.

icantly, we also need to perform a careful "pruning" of subspaces which arise in the analysis by exploiting the extra structure possessed by these subspaces. Fortunately for us, the rank metric codes of [GWX16] demonstrate that the solution is to use subspace designs. However, are required to provide a new constructions of subspace designs, as none of the results present in the literature a suitable for our purposes. Broadly speaking, our approach necessitates the use of more sophisticated ideas from linear-algebraic list-decoding than were present in [GUV09].

We now describe our approach in more detail. Recall that $\mathbb{F}_{q^n}[X;(\cdot)^q]_{<k}$ denotes the space of all linearized polynomials of q-degree less than k. We fix a subspace $\mathcal{F} \leq \mathbb{F}_{q^n}[X;(\cdot)^q]_{<k}$ of dimension n over $\mathbb{F}_q$, and then each Γ_j is simply the evaluation of $f \in \mathcal{F}$ at a point $\alpha_j \in \mathbb{F}_{q^n}$, i.e., $\Gamma_j(f) = f(\alpha_j)$. We will in fact choose $\alpha_1, \ldots, \alpha_d$ to span a degree d field extension $\mathbb{F}_h$ over $\mathbb{F}_q$; this is much like what is done in [GWX16].

The analysis of this construction mirrors the proof of the list-decodability of the codes from [GWX16] and we sketch it here. In contrapositive, the dimension expander property amounts to showing that for every subspace $V \leq \mathbb{F}_{q^n}$ of bounded dimension, the space of $f \in \mathcal{F}$ such that $f(\alpha_j) \in V$ for all $j \in [d]$ has dimension about a factor d smaller. So we study the structure of the space of polynomials $f \in \mathbb{F}_{q^n}[X,(\cdot)^q]_{<k}$ which, for some fixed subspace V, have $f(\alpha_j) \in V$ for all $j \in [d]$, and show that it forms a *periodic subspace* (cf. Definition 8.2.8). Thus, the challenge at this point is to find an appropriate subspace $\mathcal{F} \leq \mathbb{F}_{q^n}[X;(\cdot)^q]_{<k}$ that has small intersection with *every* periodic subspace.

We accomplish this by using an appropriate construction of a *subspace design* (cf. Definition 8.2.6). Briefly, subspace designs are collections of subspaces $\{H_i\}_{i=1}^k$ such that, for any subspace W of bounded dimension, the total intersection dimension $\sum_{i=1}^k \dim(H_i \cap W)$ is small. In fact, we will be interested in a slightly more general object: we are only required to have small intersection with $\mathbb{F}_h$-subspaces W, where we recall that $\mathbb{F}_h$ is an extension field of $\mathbb{F}_q$. Once we have a good subspace design, it will suffice to define $\mathcal{F} = \left\{ f(X) = \sum_{i=0}^{k-1} f_i X^{q^i} : f_i \in H_{i+1} \right\}$.

Thus, we have reduced the task of constructing dimension expanders to the task of constructing subspace designs. We provide two constructions, yielding our two claimed constructions of dimension expanders. Both use an explicit subspace design given in [GK16] as a black box (see Lemma 8.4.1). We remark that in this work the authors only considered the $d = 1$ case, i.e., the H_i's were required to have small intersection with all $\mathbb{F}_q$-subspaces, and not just $\mathbb{F}_h$-subspaces. Thus, our task is easier in the sense that we only require intersection with $\mathbb{F}_h$-subspaces to be small. However, for our purposes, we will require a better bound on the total intersection dimension than that which is guaranteed by [GK16]. We also remark that this construction requires linear-sized fields which is the source of our restrictions on field size.

The subspace design which yields our degree-proportional expander is more elementary so we describe it first. Essentially, we take the subspace design of [GK16] and define it over an "intermediate field" $\mathbb{F}_\ell$, i.e., $\mathbb{F}_q \subseteq \mathbb{F}_\ell \subseteq \mathbb{F}_h$. By appropriately choos-

ing the degree of the extension we are able to guarantee smaller intersections with $\mathbb{F}_h$-subspaces and also allow q to be smaller (as it is now only ℓ that must be linear in n, and we can take $\ell \approx q^{1/\delta}$).

Our construction which yields lossless dimension expanders is more involved. We take the construction of [GK16] and embed the subspaces isomorphically into $\mathbb{F}_q[Y]_{<\delta n}$ (for an appropriately chosen constant $\delta > 0$), where $\mathbb{F}_q[Y]_{<\delta n}$ denotes the $\mathbb{F}_q$-vector space of polynomials of degree $< \delta n$. We in turn map each of these subspaces into $\mathbb{F}_h^{n/d}$ by evaluating the polynomials at a tuple of correlated degree d places (recall that $h = q^d$). Concretely, evaluating a polynomial at a degree d place corresponds to reducing the polynomial modulo an irreducible degree d polynomial $g(Y)$, and then identifying $\mathbb{F}_q[Y]/(g(Y)) \cong \mathbb{F}_{q^d}$. Identifying $\mathbb{F}_h^{n/d}$ with $\mathbb{F}_{q^n}$ completes the construction. Ideas similar to the linear algebraic list-decoding of folded Reed-Solomon codes [Gur11; GW13] are used to prove the final bound on intersection dimension, which with a careful choice of parameters is good enough to guarantee lossless expansion. For technical reasons, in order to explicitly construct the degree d place we require $n = q - 1$.

Lastly, while we are able to use explicit constructions of subspace designs to obtain degree-proportional dimension expanders, we observe that with high probability a random $\mathbb{F}_q$-subspace H of dimension n/k will have small intersection with every $\mathbb{F}_h$-subspace W of bounded dimension. We refer to such an H as a *subspace evasive subspace* (cf. Definition 8.7.1). Then, instantiating our approach with $\mathcal{F} = \left\{ f(X) = \sum_{i=0}^{k-1} f_i X^{q^i} : f_i \in H \right\}$ will provide a *degree-proportional* dimension expander. Thus, an explicit construction of a subspace evasive subspace with parameters matching the probabilistic construction would yield an explicit degree-proportional dimension expander. We leave the construction of such an H, which seems like an interesting object in its own right, for future work.

8.1.4 Previous Work

We now survey previous work on dimension expanders. Previous constructions have followed one of three main approaches: the first uses Cayley graphs of groups satisfying Kazhdan's property T, the second uses monotone expanders, and the third uses rank condensers.

Property T. The problem of constructing dimension expanders was originally proposed by Wigderson [Wig04; Bar+04]. Along with the definition, he conjectured that dimension expanders could be constructed with Cayley graphs. This is in analogy with expander graphs, where such approaches have been very successful. To construct an expanding Cayley graph, one uses a group G with generating set S satisfying *Kazhdan's property T*. Wigderson conjectured (see Dvir and Wigderson [DS11, Conjecture 7.1]) that an expanding Cayley graph would automatically yield a dimension expander. More precisely, if one takes any irreducible representation $\rho : G \to \mathrm{GL}_n(\mathbb{F})$ of the group G,

then $\rho(S) = \{\rho(g) : g \in S\}$ would provide a dimension expander.

In characteristic 0, Lubotzky and Zelmanov [LZ08] succeeded in proving Wigderson's conjecture. In an independent work, Harrow [Har08] proved the same result in the context of *quantum expanders*, which imply dimension expanders in characteristic zero. Unfortunately, their approaches intrinsically use the notion of unitarity which does not possess a meaningful definition over positive characteristic. Lubotzky and Zelmanov also provided an example of an expanding group with an irreducible representation over a finite field that does *not* yield a dimension expander. [5] The following theorem summarizes this discussion.

Theorem 8.1.7 ([LZ08; Har08])**.** *Let $\mathbb{F}$ be a field of characteristic zero, $n \geq 1$ an integer. There exists an explicit $(1/2, 1 + \Omega(1))$-dimension expander over $\mathbb{F}^n$ of constant degree.*

Unfortunately, this approach is inherently unable to construct unbalanced dimension expanders. Moreover, it is unclear to us if it is possible to obtain expansion proportional to the degree via this strategy.

Monotone expanders. Consider a bipartite graph G with left and right partition given by $[n]$ and maximum left-degree d, and let $\Gamma_1, \ldots, \Gamma_d : [n] \to [n]$ denote the neighbor (partial)[6] functions of the graph, i.e., each left vertex $i \in [n]$ is connected to $\Gamma_j(i)$ whenever it's defined. One can then define the linear maps $\Gamma'_1, \ldots, \Gamma'_d$ which map $e_i \mapsto e_{\Gamma_j(i)}$ whenever $\Gamma_j(i)$ is defined and then extending linearly, where the e_i are the standard basis vectors. It is easily seen that if G is an expander, the corresponding collection $\{\Gamma'_j\}_{j=1}^d$ will expand subspaces of the form $\mathrm{span}\{e_i : i \in S\}$ for $S \subseteq [n]$. To expand all subspaces (and hence obtain dimension expanders), Dvir and Shpilka [DS11] implicitly observed that it is sufficient for the maps Γ_j to be *monotone* (this observation is made explicit in [DW10]). Note that the matrices Γ'_j have entries in $\{0, 1\}$, and they form a dimension expander over *every* field.

Thus, in order to construct dimension expanders, it suffices to construct monotone expander graphs. Unfortunately, constructing monotone expander graphs is a highly non-trivial task: indeed, probabilistic arguments seem to be insufficient to even establish the *existence* of monotone expanders (see [DW10; BY13]). Nonetheless, Dvir and Shpilka [DS07] succeeded in constructing monotone expanders with logarithmic degree, as well as constant-degree expanders with inverse-logarithmic expansion. Later, using the zig-zag product of Reingold, Vadhan and Wigderson [RVW02], Dvir and Wigderson [DW10] constructed monotone expanders of degree $\log^{(c)} n$ (the c-th iterated logarithm) for any constant c. Moreover, given any constant-degree monotone expander as a starting point (which is not known to exist via the probabilistic method), their method is capable of constructing a constant degree monotone expander graph. Lastly, by a sophisticated analysis of expansion in the group $\mathrm{SL}_2(\mathbb{R})$, Bourgain and Yehudayoff [BY13] were able to construct explicit monotone expanders of constant degree. Thus, we have

[5] In the example the characteristic of the field divides the order of the group; it could be the case that assuming this does not occur, any such irreducible representation yields a dimension expander.

[6] That is, Γ_j need only be defined on a *subset* of $[n]$.

the following theorem.

Theorem 8.1.8 ([BY13]). *Let $n \geq 1$ be an integer. There exists an explicit $(1/2, 1 + \Omega(1))$-dimension expander of degree $O(1)$ over $\mathbb{F}^n$, for every field $\mathbb{F}$.*

Unfortunately, just as with the previous approach, it is unclear to us if this argument could be adapted to yield degree-proportional dimension expanders.

Rank condensers. This final approach to constructing dimension expanders, developed by Forbes and Guruswami [FG15], uses *rank condensers*. Unlike the constructions of the previous sections, it inherently uses properties of finite fields and ideas from algebraic pseudorandomness more broadly, and thus is most in the spirit of our work. The construction proceeds in two steps. First, one "trivially" expands the subspaces by a factor of d by defining $T_j : \mathbb{F}^n \to \mathbb{F}^n \otimes \mathbb{F}^d$ mapping $v \mapsto v \otimes e_j$. The challenge is then to map $\mathbb{F}^n \otimes \mathbb{F}^d \cong \mathbb{F}^{nd}$ back to $\mathbb{F}^n$ such that subspaces do not decrease in dimension too much. This is precisely the problem of *lossy rank condensing*, namely, of constructing a small collection of linear maps $S_k : \mathbb{F}^{nd} \to \mathbb{F}^n$ such that, for any subspace U of bounded degree, there exists some S_k such that $\dim S_k(U) \geq (1 - \varepsilon) \dim U$. To complete the construction, one takes the set of $S_k T_j$ for all k, j. We remark that the construction of the rank condenser from this work uses the subspace designs of [GK16], providing more evidence for the interrelatedness of the objects studied in algebraic pseudorandomness. Unfortunately, the construction of subspace designs used in this work require polynomially large fields. The authors are able to decrease the field size using techniques reminiscent of code-concatenation at the cost of certain logarithmic penalties.

The following theorem was obtained.

Theorem 8.1.9 ([FG15]).

1. *Let $n, d \geq 1$. Assume $|\mathbb{F}| \geq \Omega(n^2)$. There exists an explicit $(\Omega(1/\sqrt{d}), \Omega(\sqrt{d}))$-dimension expander in $\mathbb{F}^n$ of degree d.*
2. *Let $\mathbb{F}_q$ be a finite field, $n, d \geq 1$. There exists an explicit $(\Omega(1/d \log_q(dn)), \Omega(d))$-dimension expander in $\mathbb{F}_q^n$ of degree $O(d^2 \log_q(dn))$.*

In order to improve the dependence on the field size, improved subspace designs over small fields were constructed by Guruswami, Xing and Yuan [GXY18]. These subspace designs yield a family of explicit $(\Omega(1/\log_q \log_q n), 1 + \Omega(1))$-dimension expander of degree $O(\log_q n)$ over $\mathbb{F}_q^n$.

8.1.5 Organization

In Section 8.2 we set notation and define the various pseudorandom objects that we use in our construction. We also provide probabilistic arguments ascertaining the existence of good dimension expanders in order to set expectations. In Section 8.3 we prove that the problem of constructing dimension expanders can be reduced to that of constructing appropriate subspace designs, which is the task we address in Section 8.4. In Section 8.5, we put all of the pieces together to deduce our main theorems for balanced dimension

Parameter	Meaning	Comments
n	the dimension of the expander	growing
q	a prime power	expanders will be $\mathbb{F}_q$-linear
d	the degree of the expander	$d \mid n$
h	a power of q	evaluation points span $\mathbb{F}_h/\mathbb{F}_q$; $h = q^d$
k	q-degree bound for linearized polynomials	$1 \leq k \leq d$, $k \mid d$
$\mathbb{F}_{q^n}[X, (\cdot)^q]_{<k}$	q-linearized polynomials of q-degree $< k$	domain of expanders is a subspace
$\mathbb{F}_{q^n}$	degree n extension of $\mathbb{F}_q$	image space of expander
m	degree of $\mathbb{F}_{q^n}/\mathbb{F}_h$	$m = \frac{n}{d}$
N	dimension of domain for unbalanced expanders	$k \mid N$
b	the "unbalancedness"; assume $\in \mathbb{Z}$	$b = \frac{N}{n}$

Table 8.1: Regularly used parameters and notations for Chapter 8.

expanders. In Section 8.6 we show that all our results readily adapt to the case of unbalanced expanders. Section 8.7 contains a discussion of subspace evasive subspaces. We list open problems in Section 8.8. Proofs deferred from the main body are provided in Section 8.9.

8.2 Background

Terminology. First, we introduce a piece of terminology specific to this chapter. Let d be an integer dividing n and let $h = q^d$. Recall that this gurantees the inclusions $\mathbb{F}_q \subseteq \mathbb{F}_h \subseteq \mathbb{F}_{q^n}$. We will often have subspaces of $W \subseteq \mathbb{F}_{q^n}$ that are linear over $\mathbb{F}_h$, i.e., for all $w \in W$ and $\alpha \in \mathbb{F}_h$ we have $\alpha w \in W$. When we wish to emphasize this, we will say that W is an $\mathbb{F}_h$-subspace. Moreover, we will write $\dim_{\mathbb{F}_q} W$ or $\dim_{\mathbb{F}_h} W$ if we need to emphasize that the dimension is computed when viewing W as an $\mathbb{F}_q$-subspace or as an $\mathbb{F}_h$-subspace, respectively. We remark that in this case, $\dim_{\mathbb{F}_q} W = d \cdot \dim_{\mathbb{F}_h} W$.

8.2.1 Dimension Expanders

We now formally define dimension expanders and provide an alternate characterization that we find easier to reason about.

Definition 8.2.1 (Dimension Expander)**.** Let $n, d \geq 1$ be an integer, $\eta > 0$ and $\beta > 1$. Let $\Gamma_1, \ldots, \Gamma_d : \mathbb{F}^n \to \mathbb{F}^n$ be linear maps. The collection $\{\Gamma_j\}_{j=1}^d$ forms a (η, β)-*dimension expander* if for all subspaces $U \leq \mathbb{F}^n$ of dimension at most ηn,

$$\dim \left(\sum_{j=1}^{d} \Gamma_j(U) \right) \geq \beta \dim U \ .$$

The *degree* of the dimension expander is d.

When clear from context we refer to a dimension expander just as an *expander*. The following proposition follows easily from the definitions.

Proposition 8.2.2 (Contrapositive Characterization)**.** *Let $n \geq 1$ be an integer, $\eta > 0$ and $\beta > 1$. Let $\Gamma_1, \ldots, \Gamma_d : \mathbb{F}^n \to \mathbb{F}^n$ be linear maps. Suppose that for all $V \leq \mathbb{F}^n$ of dimension at most $\eta\beta n$,*

$$\dim \{u \in \mathbb{F}^n : \Gamma_j(u) \in V \ \ \forall j \in [d]\} \leq \frac{1}{\beta} \dim V \ .$$

Then $\{\Gamma_j\}_{j=1}^d$ forms an (η, β)-dimension expander.

Proof. Let $U \leq \mathbb{F}^n$ be a subspace of dimension at most ηn and put $V = \sum_{j=1}^d \Gamma_j(U)$. If $\dim(V) > \eta\beta n$ then we are done, so assume $\dim(V) \leq \eta\beta n$. By the assumption of the proposition, this tells us that

$$\dim \{u \in \mathbb{F}^n : \Gamma_j(u) \in V \ \ \forall j \in [d]\} \leq \frac{1}{\beta} \dim V \ .$$

Since $U \subseteq \{u \in \mathbb{F}^n : \Gamma_j(u) \in V \ \ \forall j \in [d]\}$, we have $\dim U \leq \frac{1}{\beta} \dim V$. Rearranging this yields $\dim V \geq \beta \dim U$, as was to be shown. □

Next, we define a slight generalization of dimension expanders, wherein the domain and codomain need not have the same dimension. That is, the linear maps Γ_j map $\mathbb{F}^N \to \mathbb{F}^n$, where N and n may not be equal. We parametrize the "unbalancedness" of the dimension expander by $b = \frac{N}{n}$. In our construction we will assume $b \in \mathbb{Z}$, although this is not a fundamental restriction.

Definition 8.2.3 (Unbalanced Dimension Expanders)**.** Let $N, n, d \geq 1$ be integers, $\eta > 0$ and $\beta > 1$. Let $\Gamma_1, \ldots, \Gamma_d : \mathbb{F}^N \to \mathbb{F}^n$ be linear maps. Set $b = \frac{N}{n}$. The collection $\{\Gamma_j\}_{j=1}^d$ forms a *b-unbalanced (η, β)-dimension expander* if for all subspaces $U \leq \mathbb{F}^N$ of dimension at most ηN,

$$\dim \left(\sum_{j=1}^d \Gamma_j(U) \right) \geq \beta \dim U \ .$$

The *degree* of the unbalanced dimension expander is d.

The appropriate generalization of Proposition 8.2.2 is as follows.

Proposition 8.2.4 (Contrapositive Characterization)**.** *Let $N, n \geq 1$ be integers, $\eta > 0$ and $\beta > 1$. Put $b = \frac{N}{n}$. Let $\Gamma_1, \ldots, \Gamma_d : \mathbb{F}^N \to \mathbb{F}^n$ be linear maps. Suppose that for all $V \leq \mathbb{F}^n$ of dimension at most $\eta\beta N$,*

$$\dim \left\{u \in \mathbb{F}^N : \Gamma_j(u) \in V \ \ \forall j \in [d]\right\} \leq \frac{1}{\beta} \dim V \ .$$

Then $\{\Gamma_j\}_{j=1}^d$ forms a b-unbalanced (η, β)-dimension expander.

We now quote the parameters achievable by a random construction of unbalanced dimension expanders. This sets the stage and ultimate target to aim for with explicit constructions. We prove this proposition in Section 8.9.2, and we remark that our argument is completely analogous to that given in Section C.3 of [FG15].

Proposition 8.2.5 (Generalization of [FG15, Proposition C.10])**.** *Let $\mathbb{F}_q$ be a finite field, N, n positive integers and put $b := \frac{N}{n}$. Let $\beta > 1$ and $\eta \in (0, \frac{1}{b\beta})$. Then, assuming*

$$d \geq \beta + \frac{b}{1 - b\beta\eta} + \log_q 16 \, ,$$

there exists a collection of linear maps $\Gamma_1, \ldots, \Gamma_d : \mathbb{F}_q^N \to \mathbb{F}_q^n$ forming a (η, β)-unbalanced dimension expander.

Thus, for $b = 1$, if we wish to have $\beta = (1-\varepsilon)d$ and $\eta = \frac{1-\varepsilon}{d}$ we may take $d = O(1/\varepsilon^2)$. We remark that in Theorem 8.5.2, we obtain $d = O(1/\varepsilon^3)$. Similarly, for the b-unbalanced case, if we would like $\beta = (1-\varepsilon)d$ and $\eta = \frac{1-\varepsilon}{bd}$ we may take $d = O(b/\varepsilon^2)$, while in Theorem 8.6.7 we obtain $d = O(b/\varepsilon^3)$.

8.2.2 Subspace Designs

A crucial ingredient in our construction of dimension expanders are subspace designs. They were originally introduced by Guruswami and Xing [GX13] in order to obtain algebraic codes which are list-decodable up to the Singleton bound. As in [GWX16], we will be concerned with a slight weakening of this notion, where we are only concerned with having small intersection with subspaces which are linear over an extension of the base field, although we will also require the intersection dimension to be smaller.

Definition 8.2.6 (Subspace Design)**.** Let V be a $\mathbb{F}_{q^d}$-vector space. A collection $H_1, \ldots, H_k \subseteq V$ of $\mathbb{F}_q$-subspaces is called a (s, A, d)*-subspace design in V* if for every $\mathbb{F}_{q^d}$-subspace $W \subseteq V$ of $\mathbb{F}_{q^d}$-dimension s,

$$\sum_{i=1}^{k} \dim_{\mathbb{F}_q}(H_i \cap W) \leq As \, .$$

We call a subspace design *explicit* if there is an algorithm outputting $\mathbb{F}_q$-bases for each subspace H_i in $\mathrm{poly}(n)$ field operations.

Remark 8.2.7. In previous works, what we have termed a (s, A, d)-subspace design would have been called a (s, As, d)-subspace design. We find it more convenient in this work to remove the multiplicative factor of s from the parameter in the definition.

8.2.3 Periodic Subspaces

We now abstract the kind of structure that will be found in the subspace of $\mathbb{F}_q^n$ which is mapped entirely into a low-dimensional subspace of $\mathbb{F}_q^n$ by the d linear transformations comprising our dimension expander. We note that our definition here is slightly different in form and notation than earlier ones in [GX13; GWX16].

Definition 8.2.8 (Periodic Subspace)**.** For positive integers n, k, s, d with $d|n$, an $\mathbb{F}_q$-subspace T of $\mathbb{F}_{q^n}^k$ is said to be (s, d)*-periodic* if there exists an $\mathbb{F}_{q^d}$-subspace $W \subseteq \mathbb{F}_{q^n}$

of dimension at most s such that for all j, $1 \le j \le k$, and all $\xi_1, \xi_2, \ldots, \xi_{j-1} \in \mathbb{F}_{q^n}$, the $\mathbb{F}_q$-affine subspace

$$\{\xi_j : \exists v \in T \text{ with } v_\iota = \xi_\iota \text{ for } 1 \le \iota \le j\} \subseteq \mathbb{F}_{q^n}$$

belongs to a coset of W. In other words, for every *prefix* $(\xi_1, \ldots, \xi_{j-1})$, the possible extensions ξ_j to the j-th symbol that can belong to a vector in T are contained in a coset of W.

An important property of periodic subspaces is that they have small intersection with subspace designs. This is captured by the following proposition.

Proposition 8.2.9 ([GWX16], Proposition 3.9)**.** *Let T be a (s, d)-periodic $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^n}^k$, and $H_1, \ldots, H_k \subseteq \mathbb{F}_{q^n}$ be $\mathbb{F}_q$-subspaces forming a (s, A, d) subspace design in $\mathbb{F}_{q^n}$. Then $T \cap (H_1 \times \cdots \times H_k)$ is an $\mathbb{F}_q$-subspace of dimension at most As.*

8.3 Construction

As discussed in the introduction (Section 8.1), the construction of our dimension expander is inspired by recent constructions of variants of Gabidulin codes for list-decoding in the rank metric. Indeed, the analysis of our dimension expander proceeds similarly to the analysis of list-decodability of the rank metric codes presented in [GWX16].

Our dimension expanders map $\mathbb{F}_q^n \to \mathbb{F}_q^n$. We view the domain as

$$\mathcal{F} := \left\{ f(X) = \sum_{i=0}^{k-1} f_i X^{q^i} : f_i \in H_i,\ i = 0, \ldots, k-1 \right\}$$

where $H_0, \ldots, H_{k-1}$ give a collection of $\mathbb{F}_q$-subspaces of $\mathbb{F}_{q^n}$, each of $\mathbb{F}_q$-dimension $\frac{n}{k}$ (thus, we assume $k|n$). We will choose $H_0, H_1, \ldots, H_{k-1}$ forming a subspace design. We view the image space as $\mathbb{F}_{q^n}$. Let $h = q^d$, and let $\alpha_1, \ldots, \alpha_d$ give a basis for $\mathbb{F}_h$ over $\mathbb{F}_q$. We assume $d|n$ and write $md = n$. For $j = 1, \ldots, d$, we define

$$\Gamma_j : \mathcal{F} \to \mathbb{F}_{q^n} \quad \text{by} \quad f \mapsto f(\alpha_j)\,. \tag{8.1}$$

That is, each $\Gamma_j(f)$ is just the evaluation of f at the basis element α_j. These maps are clearly linear over $\mathbb{F}_q$.

Analysis. We now prove that the collection $\{\Gamma_j\}_{j=1}^d$ forms a dimension expander.

For positive integers D, s with $s \le m$, we define $\mathcal{L}_{D,s}$ to be the space of polynomials $Q \in \mathbb{F}_{q^n}[Z_0, \ldots, Z_{s-1}]$ of the form $Q(Z_0, \ldots, Z_{s-1}) = A_0(Z_0) + \cdots + A_{s-1}(Z_{s-1})$ with each $A_i \in \mathbb{F}_{q^n}[X; (\cdot)^q]_{<D}$, i.e., each A_i is a q-linearized polynomial of q-degree at most $D-1$.

Lemma 8.3.1. *Let $V \subseteq \mathbb{F}_{q^n}$ be an $\mathbb{F}_q$-subspace of dimension B. If $Ds > B$, there exists a nonzero polynomial $Q \in \mathcal{L}_{D,s}$ such that*

$$\forall v \in V,\ \ Q(v, v^h, \ldots, v^{h^{s-1}}) = 0\,. \tag{8.2}$$

Proof. Let $v_1, \ldots, v_B$ give a basis for V over $\mathbb{F}_q$. Then, since $\gamma \mapsto \gamma^h = \gamma^{q^d}$ is a linear operation over $\mathbb{F}_q$, so long as $Q(v_i, v_i^h, \ldots, v_i^{h^{s-1}}) = 0$ for all $i \in [B]$ we have $Q(v, v^h, \ldots, v^{h^{s-1}}) = 0$ for all $v \in V$. Thus, finding such a Q amounts to solving a homogeneous linear system over $\mathbb{F}_{q^n}$ with B constraints. Since the $\mathbb{F}_{q^n}$-dimension of $\mathcal{L}_{D,s}$ is $Ds > B$, a nonzero $Q \in \mathcal{L}_{D,s}$ meeting Condition (8.2) must exist. □

Given a polynomial $g(X) = g_0 + g_1 X + \cdots + g_r X^r$ and an automorphism τ of $\mathbb{F}_{q^n}$, we write g^τ for the polynomial $g^\tau(X) = \tau(g_0) + \tau(g_1)X + \cdots + \tau(g_r)X^r$, and let $g^{\tau^i} = (g^{\tau^{i-1}})^\tau$. We let $\sigma : \gamma \mapsto \gamma^h$, i.e., σ is the Frobenius automorphism of $\mathbb{F}_{h^m} = \mathbb{F}_{q^n}$ over $\mathbb{F}_h$.

Lemma 8.3.2. *Let $f \in \mathbb{F}_{q^n}[X]$ be a q-linearized polynomial with q-degree at most $k-1$. Let $V \subseteq \mathbb{F}_{q^n}$ be an $\mathbb{F}_q$-subspace, and $Q \in \mathcal{L}_{D,s}$ a polynomial satisfying* (8.2)*. Suppose that $f(\alpha) \in V$ for all $\alpha \in \mathbb{F}_h = \mathbb{F}_{q^d}$ and that $D \leq d - k + 1$. Then*

$$A_0(f(X)) + A_1(f^\sigma(X)) + \cdots + A_{s-1}(f^{\sigma^{s-1}}(X)) = Q(f(X), f^\sigma(X), \ldots, f^{\sigma^{s-1}}(X)) = 0\,. \tag{8.3}$$

Proof. Let $\alpha \in \mathbb{F}_h$. Since $f(\alpha) \in V$ by assumption, we have

$$Q(f(\alpha), f(\alpha)^h, \ldots, f(\alpha)^{h^{s-1}}) = 0$$

as we have assumed Q satisfies Equation (8.2). Now, since $\alpha \in \mathbb{F}_h$, we have $\alpha^h = \alpha$, so

$$f(\alpha)^h = \left(\sum_{i=0}^{k-1} f_i \alpha^{q^i}\right)^h = \sum_{i=0}^{k-1} f_i^h (\alpha^{q^i})^h = \sum_{i=0}^{k-1} f_i^h \alpha^{q^i} = f^\sigma(\alpha)\,,$$

and by iterating we have $f(\alpha)^{h^i} = f^{\sigma^i}(\alpha)$ for all $i = 0, \ldots, s-1$. Thus, we find that for all $\alpha \in \mathbb{F}_h$,

$$Q(f(\alpha), f^\sigma(\alpha), \ldots, f^{\sigma^{s-1}}(\alpha)) = 0\,.$$

Now, the univariate polynomial $R_f(X) := Q(f(X), f^\sigma(X), \ldots, f^{\sigma^{s-1}}(X)) \in \mathbb{F}_{q^n}[X]$ has q-degree at most $(D-1) + (k-1) = D + k - 2$. Thus, if $D \leq d - k + 1$, the q-degree of $R_f(X)$ is at most $d - 1$. Since it vanishes on $\mathbb{F}_h$, an $\mathbb{F}_q$-subspace of dimension d, we conclude that $R_f(X)$ must be the 0 polynomial. □

Lemma 8.3.3. *The set of solutions to Equation* (8.3)*, for any nonzero $Q \in \mathcal{L}_{D,s}$ (for arbitrary D), is an $(s-1, d)$-periodic subspace.*

Proof. First, by replacing $A_0, \ldots, A_{s-1}$ with $A_0^{q^j}, \ldots, A_{s-1}^{q^j}$ for an appropriate j and identifying X^{q^n} with X (which is valid since we only ever evaluate the polynomials on elements of $\mathbb{F}_{q^n}$), we may assume that there exists an $i^* \in \{0, \ldots, s-1\}$ such that A_{i^*} has a nonzero coefficient on X. (Of course, this might increase the q-degree of the A_i.)

Write $A_\iota(X) = a_{\iota,0}X + a_{\iota,1}X^q + a_{\iota,2}X^{q^2} + \cdots$ for $\iota = 0, \ldots, s-1$. Then, for $\ell = 0, 1, \ldots, k-1$, we define

$$B_\ell(X) := a_{0,\ell}X + a_{1,\ell}X^h + \cdots + a_{s-1,\ell}X^{h^{s-1}}\,.$$

Since $a_{i^*,0} \neq 0$, we see that $B_0 \neq 0$. Since $s-1 \leq m-1$, if $W = \ker(B_0)$, we find that W is an $\mathbb{F}_h$-subspace of $\mathbb{F}_{q^n} = \mathbb{F}_{h^m}$ of dimension at most $s-1$.

Condition (8.3) informs us that

$$A_0(f(X)) + A_1(f^\sigma(X)) + \cdots + A_{s-1}(f^{\sigma^{s-1}}(X)) = 0\,. \tag{8.4}$$

The coefficient of X in the left hand size of (8.4) is $B_0(f_0)$; upon equating it to 0, we see $f_0 \in W$.

Now, fix an $i \in \{1, \ldots, k-1\}$. The coefficient of X^{q^i} in the left hand side of (8.4) is

$$B_i(f_0^{q^i}) + B_{i-1}(f_1^{q^{i-1}}) + \cdots + B_1(f_{i-1}^q) + B_0(f_i)\,.$$

Upon equating this coefficient to 0, we see that $f_i \in W + \theta_i$, where $\theta_i \in \mathbb{F}_{q^n}$ is determined by $f_0, f_1, \ldots, f_{i-1}$. Specifically, we can take $\theta_i = -B_i(f_0^{q^i}) - B_{i-1}(f_1^{q^{i-1}}) - \cdots - B_1(f_{i-1}^q)$. Therefore, for each choice of $(f_0, f_1, \ldots, f_{i-1})$, f_i must belong to a coset of the subspace W. This shows that the solutions lie in a $(s-1, d)$-periodic subspace. □

Equipped with these lemmas, we are in a position to deduce our main theorem for this section.

Theorem 8.3.4. *Let $\{H_i\}_{i=0}^{k-1}$ give a $(s-1, A, d)$-subspace design for all $s-1 \leq \mu n$ for some $0 < \mu < 1/d$. Then $\{\Gamma_j\}_{j=1}^d$ is a $(\mu A, \frac{d-k+1}{A})$-dimension expander. Moreover if the subspace design is explicit then the dimension expander is explicit.*

Proof. We will appeal to Proposition 8.2.2. Let $V \subseteq \mathbb{F}_{q^n}$ be an $\mathbb{F}_q$-subspace of dimension $B \leq (d-k+1)\mu n$. Let

$$U := \{f \in \mathcal{F} : \Gamma_j(f) \in V \ \ \forall j \in [d]\}.$$

By the $\mathbb{F}_q$-linearity of the polynomials f and the fact that $\alpha_1, \ldots, \alpha_d$ gives a basis for $\mathbb{F}_h$ over $\mathbb{F}_q$, we may rewrite this as

$$U = \{f \in \mathcal{F} : f(\alpha) \in V \ \ \forall \alpha \in \mathbb{F}_h\}\,.$$

Let $D = d-k+1$ and choose the integer s such that $\frac{B}{D} < s \leq \frac{B}{D} + 1 \leq \mu n + 1$. As $\mu < 1/d$, we have $s \leq n/d = m$. By Lemma 8.3.1, we have a nonzero $Q \in \mathcal{L}_{D,s}$ such that $Q(v, v^h, \ldots, v^{h^{s-1}}) = 0$ for all $v \in V$. We then have that every $f \in \mathcal{F}$ satisfies (8.3), so we conclude that U is contained in a $(s-1, d)$-periodic subspace. Since $s-1 \leq \mu n$, our assumption on $\{H_i\}_{i=0}^{k-1}$ combined with Proposition 8.2.9 tells us that U is contained in an affine subspace over $\mathbb{F}_q$ of dimension at most $A(s-1)$. In particular, $\dim_{\mathbb{F}_q} U \leq A(s-1)$. Recalling $s - 1 \leq \frac{B}{D}$,

$$\dim_{\mathbb{F}_q} U \leq A \cdot \frac{B}{D} = \frac{A}{D} \cdot \dim_{\mathbb{F}_q} V\,.$$

Applying Proposition 8.2.2 with $\eta = \mu A$ and $\beta = \frac{D}{A}$, we conclude that $\{\Gamma_j\}_{j=1}^d$ gives a $(\mu A, \frac{D}{A})$-dimension expander, as was to be shown.

Finally, as for the explicitness, suppose that $H_1, \ldots, H_k$ are explicit. Thus, in $\mathrm{poly}(n)$ field operations we may output $\mathbb{F}_q$-bases $\mathcal{B}_1, \ldots, \mathcal{B}_k$ for $H_1, \ldots, H_k$. Then, we construct the basis $\mathcal{B} = \{f = \sum_{i=0}^{k-1} f_i X^{q^i} : f_i \in \mathcal{B}_i, i \in [k]\}$, and enumerate $\mathcal{B} = \{g_1, \ldots, g_n\}$. Finally, for $j \in [d]$ we output the matrix Γ_j obtained by evaluating $g_1(\alpha_j), \ldots, g_n(\alpha_j)$, writing each $g_i(\alpha_j)$ in an $\mathbb{F}_q$-basis for $\mathbb{F}_{q^n}$, and then putting $g_i(\alpha_j)$ as the i-th column of Γ_j. □

Intuitively, we have that subspaces of dimension As are expanded to subspaces of dimension $(d-k+1)s/A$. This informs what we should hope for from our subspace designs. In particular, obtaining $A = O(1)$ is enough to obtain a degree proportional expander (by setting $k = \Theta(d)$), while if $A \approx 1 + \varepsilon$ and $k \approx \varepsilon d$ we can obtain a *lossless* expander. With these goals in mind, we turn our attention to constructing subspace designs.

8.4 Constructions of Subspace Designs

For the case of $d = 1$, explicit constructions of subspace designs have been given in previous works. The first explicit construction was given in [GK16], using ideas which had been developed in constructions of list-decodable codes. This construction was subsequently improved over fields of small size in [GXY18].

A previous construction of a subspace design for $d > 1$ was given in [GWX16]. In this work, a subspace design over the base field (i.e., for $d = 1$) was intersected with a *subspace evasive set* from [DL12]. However, for our purposes, the size of the intersection dimension (i.e., the product As) of this construction is too large. In that work, the authors were more concerned with ensuring that the H_i's had large dimension; however, we only require that the H_i's have dimension n/k.

We provide two constructions of subspace designs in this work, yielding our two constructions of dimension expanders. The first construction yields a *degree-proportional* dimension expander over fields of size n^δ (for arbitrarily small constant δ). The next yields a *lossless* dimension expander. The only drawback is that it requires a field of size linear in n (for technical reasons, we take $q - 1 = n$). We present our first construction in Section 8.4.1 and our second construction in Section 8.4.2.

Both of our constructions use as a black box a subspace design provided in [GK16]. Specifically, by taking $r = 2$ in Theorem 7 of [GK16], we obtain a subspace design with the following parameters.

Lemma 8.4.1. *For all positive integers s, t, m and prime powers ℓ satisfying $s \le t \le m < \ell$, there is an explicit collection of $M \ge \frac{\ell^2}{4t}$ $\mathbb{F}_\ell$-spaces $V_1, V_2, \ldots, V_M \subseteq \mathbb{F}_\ell^m$, each of codimension $2t$, which forms an $(s, \frac{m-1}{2(t-s+1)}, 1)$ subspace design in $\mathbb{F}_\ell^m$.*

8.4.1 Subspace Designs via an Intermediate Field

This first construction takes the subspace design of Lemma 8.4.1 defined over an intermediate field $\mathbb{F}_\ell$. That is, we fix an integer $1 < c < d$ such that $c|d$ so that, for $\ell = q^c$, $\mathbb{F}_q \subseteq \mathbb{F}_\ell \subseteq \mathbb{F}_h$. Then, if $\omega_1, \ldots, \omega_m$ gives a basis for $\mathbb{F}_{h^m}/\mathbb{F}_h$, define

$$L = \left\{ \sum_{i=1}^{m} a_i \omega_i : a_i \in \mathbb{F}_\ell \right\} .$$

This is an $\mathbb{F}_\ell$-subspace of $\mathbb{F}_{h^m} = \mathbb{F}_{q^n}$ of $\mathbb{F}_\ell$-dimension m, as $\omega_1, \ldots, \omega_m$ are linearly independent over $\mathbb{F}_h$ and so *a fortiori* are linearly independent over the subfield $\mathbb{F}_\ell$. Thus, $L \simeq \mathbb{F}_\ell^m$, and we fix an $\mathbb{F}_\ell$-linear isomorphism $\psi : \mathbb{F}_\ell^m \to L$. Note that an $\mathbb{F}_\ell$-linear map is automatically $\mathbb{F}_q$-linear, so, in particular, the dimension of $\mathbb{F}_q$-subspaces in $\mathbb{F}_\ell^m$ are preserved by ψ. Then, if $V_1, \ldots, V_k$ give the subspace design from Lemma 8.4.1, we define $H_i := \psi(V_i)$ for $i = 1, \ldots, k$.

Our analysis of the subspace design makes use of the following lemma. We defer its proof to Section 8.9.1.

Lemma 8.4.2. *Let W be an $\mathbb{F}_h$-subspace of $\mathbb{F}_{q^n}$ and let $U := W \cap L$. Then U is an $\mathbb{F}_\ell$-subspace of L and $\dim_{\mathbb{F}_\ell} U \le \dim_{\mathbb{F}_h} W$.*

Proposition 8.4.3. *Let $\ell = q^c$ with $c = \frac{d}{k} \cdot \frac{m}{m-2t}$, where $1 \le k < d$. For all $1 \le s < t < \ell$ and $1 \le k < d$ such that $\ell^2 \ge 4kt$, $k|d$, $m|k(m-2t)$ and $k(m-2t)|n$, there is an explicit construction of $\{H_i\}_{i=1}^k$ that forms a $(s, \frac{d}{k} \cdot \frac{m-1}{m-2t} \cdot \frac{m}{2(t-s)}, d)$-subspace design in $\mathbb{F}_{q^n}$. Furthermore $\dim_{\mathbb{F}_q} H_i = \frac{n}{k}$ for all $i = 1, \ldots, k$.*

Proof. The condition that $k|d$ implies $k|n$, so $\frac{n}{k} \in \mathbb{Z}$. The condition that $k(m-2t)|n$ implies that $c \in \mathbb{Z}$. Finally, the condition that $m|k(m-2t)$ implies $c|d$ and so $\mathbb{F}_\ell \subseteq \mathbb{F}_h \subseteq \mathbb{F}_{q^n}$. We take the first k subspaces $\{V_i\}_{i=1}^k$ given in Lemma 8.4.1 (which is valid since $\ell^2/(4t) \ge k$) and define $H_i = \psi(V_i)$ for $i = 1, \ldots, k$. For any $\mathbb{F}_\ell$-subspace $U \subseteq L$ of $\mathbb{F}_\ell$-dimension $u < t$, we have

$$\sum_{i=1}^{k} \dim_{\mathbb{F}_\ell}(U \cap H_i) = \sum_{i=1}^{k} \dim_{\mathbb{F}_\ell}(\psi^{-1}(U) \cap V_i) \le \frac{(m-1)u}{2(t-u+1)} .$$

Now for any $\mathbb{F}_h$-subspace $W \subseteq \mathbb{F}_{q^n}$ of dimension s, Lemma 8.4.2 tells us that the intersection $U := W \cap L$ is an $\mathbb{F}_\ell$-subspace in L of dimension at most s. Let $u \le s$ be the $\mathbb{F}_\ell$-dimension of U. As $W \cap H_i = U \cap H_i$ (since $H_i \subseteq L$), we have

$$\begin{aligned}
\sum_{i=1}^{k} \dim_{\mathbb{F}_q}(W \cap H_i) &= c \sum_{i=1}^{k} \dim_{\mathbb{F}_\ell}(U \cap H_i) \\
&\le \frac{d}{k} \cdot \frac{m-1}{m-2t} \cdot \frac{m}{2(t-u)} u \\
&\le \frac{d}{k} \cdot \frac{m-1}{m-2t} \cdot \frac{m}{2(t-s)} s .
\end{aligned}$$

Note that each H_i has $\mathbb{F}_\ell$ dimension $m-2t$, i.e, it has $\mathbb{F}_q$-dimension $c(m-2t) = \frac{n}{k}$ by our choice of parameters.

As for the explicitness, upon computing the bases $\mathcal{B}_1, \ldots, \mathcal{B}_k$ for $V_1, \ldots, V_k$ we may obtain bases for $H_1, \ldots, H_k$ by applying ψ to each element of the corresponding basis. Thus, assuming the basis for V_i can be computed in $\mathrm{poly}(m)$ field operations we may also compute a basis for H_i in $\mathrm{poly}(m) = \mathrm{poly}(n)$ field operations. □

We now fix parameters in such a way to show that we can obtain a subspace design over fields of size n^δ for any constant $\delta > 0$.

Corollary 8.4.4. *Let $\delta > 0$ be given and choose an integer r such that $\frac{1}{2\delta} < r \leq \frac{1}{\delta}$. Let k, d be integers such that $d = 2k$ and $r|k$. Assume moreover that $2r|m$. Then, assuming $q \geq n^\delta$, there exists an explicit construction of $\{H_i\}_{i=1}^k$ that forms a $(s, \frac{8}{\delta}, d)$-subspace design in $\mathbb{F}_{q^n}$ for all $s \leq \frac{1-2\delta}{4d}n$. Moreover $\dim_{\mathbb{F}_q} H_i = \frac{n}{k}$ for all $i = 1, \ldots, k$.*

Proof. Put $t = \frac{1}{2}(1 - \frac{1}{r})m$, so $m - 2t = \frac{m}{r}$. Our assumptions on m imply that $t \in \mathbb{Z}$. Moreover, $k(m-2t) = km/r$, and so $m|k(m-2t)$ as we assumed $r|k$. We also have $k(m-2t) = (km/r)|md$ as $k|d$ and $(m/r)|m$. Thus, all the divisibility conditions of Proposition 8.4.3 are satisfied, so let $H_1, \ldots, H_k \subseteq \mathbb{F}_{q^n}$ denote the explicit subspace design promised by the proposition, each satisfying $\dim_{\mathbb{F}_q} H_i = \frac{n}{k}$.

Defining c as in Proposition 8.4.3, we have

$$c = \frac{d}{k} \cdot \frac{m}{m-2t} = 2 \cdot \frac{m}{m/r} = 2r \, .$$

Next, assuming $s \leq t/2 = \frac{1}{4}(1 - \frac{1}{r})m$, we have the bound

$$\frac{d}{k} \cdot \frac{m-1}{m-2t} \cdot \frac{m}{2(t-s)} \leq 2r \cdot \frac{m}{\frac{1}{2}(1-\frac{1}{r})m} = \frac{4r}{1-\frac{1}{r}} \leq 8r \leq \frac{8}{\delta} \, ,$$

where the second to last inequality is valid assuming $r \geq 2$ (which is valid assuming δ is sufficiently small). Note further that $\frac{1}{4}(1-\frac{1}{r}) \geq \frac{1}{4}(1-2\delta)$. Thus, we conclude that $H_1, \ldots, H_k$ forms a $(s, \frac{8}{\delta}, d)$-subspace design in $\mathbb{F}_{q^n}$ for all $s \leq \frac{1-2\delta}{4d}n$, as was to be shown.

Lastly, note that $c = 2r > 1/\delta$. To satisfy the conditions of Proposition 8.4.3 we require $\ell = q^c > t = \frac{1}{2}(1-\frac{1}{r})m$ and $\ell^2 \geq 4kt = 2k(1-\frac{1}{r})m$; note that the first condition implies the second for m large. Thus, we just require $q > t^{1/c}$, which is implied by $q \geq n^\delta$ as $t^{1/c} < n^\delta$. □

8.4.2 Construction via Correlated High-Degree Places

The following section uses more sophisticated ideas from the theory of algebraic function fields. For additional background information, we refer the reader to [Sti09, Chapter 1] and [NX09, Chapter 1].

We utilize techniques developed in the context of linear algebraic list-decoding of Folded Reed-Solomon codes [Gur11; GW13]. Briefly, we take a subspace design in the

space of polynomials of bounded degree, and then map it into $\mathbb{F}_h^m$ in a manner reminiscent of the encoding map of a folded Reed-Solomon code. As we are concerned with bounding the intersection dimension with $\mathbb{F}_h$-linear spaces, we in fact evaluate the polynomial at degree d places. Details follow.

Let ζ be a primitive root of the finite field $\mathbb{F}_q$. Choose a real $\delta \in (0,1)$ such that $\delta > \frac{1}{k}$ and $\delta n < q-1$, where we recall $0 < k < d$ and $n = md$. Denote by σ the automorphism of the function field $\mathbb{F}_q(Y)$ sending Y to ζY. The order of σ is $q-1 \geq m$. Given $g \in \mathbb{F}_q(Y)$, we abbreviate $g^\sigma := \sigma(g(Y)) = g(\zeta Y)$.[7]

Denote by $\mathbb{F}_q[Y]_{<\delta n}$ the set of polynomials of degree less than δn. By Lemma 8.4.1, there exist $V_1, V_2, \ldots, V_k$ of $\mathbb{F}_q[Y]_{<\delta n}$, each of codimension $\delta n - \frac{n}{k}$, which forms a $(r, \frac{\delta n - 1}{\delta n - \frac{n}{k} - 2r + 2}, 1)$-subspace design.

Let $P(Y)$ be an irreducible polynomial of degree d such that $P, P^\sigma, \ldots, P^{\sigma^{m-1}}$ are pairwise coprime. Consider the map

$$\pi : \mathbb{F}_q[Y]_{<\delta n} \to \mathbb{F}_{q^d}^m, \quad f \mapsto (f(P), f(P^\sigma), \ldots, f(P^{\sigma^{m-1}})) ,$$

where $f(P^{\sigma^j})$ is the residue of f in the residue field $\mathbb{F}_q[Y]/(P^{\sigma^j}) \cong \mathbb{F}_{q^d} = \mathbb{F}_h$. As the ideals $(P), (P^\sigma), \ldots, (P^{\sigma^{m-1}})$ are pairwise coprime, the Chinese Remainder Theorem guarantees that π is injective. We define

$$\widetilde{H}_i = \pi(V_i) = \left\{ (f(P), f(P^\sigma), \ldots, f(P^{\sigma^{m-1}})) : f \in V_i \right\} \subseteq \mathbb{F}_h^m \tag{8.5}$$

for $i = 1, 2, \ldots, k$.

Before analyzing the subspaces $\tilde{H}_1, \ldots, \tilde{H}_k$, we record the following fact:

Fact 8.4.5. *We have $f(P^\sigma) = (f^{\sigma^{-1}})(P)$.*

Proposition 8.4.6. *If $s < (1-\delta)m = (1-\delta)\frac{n}{d}$, then the subspaces $\widetilde{H}_1, \widetilde{H}_2, \ldots, \widetilde{H}_k$ defined above give an $(s, \frac{\delta}{1-\delta} \cdot \frac{m}{(\delta - \frac{1}{k})m - \frac{2s}{d(1-\delta)}}, d)$-subspace design in $\mathbb{F}_h^m$. Moreover $\dim_{\mathbb{F}_q} \widetilde{H}_i = \frac{n}{k}$ for all $i = 1, \ldots, k$.*

Proof. The claim about the $\mathbb{F}_q$-dimension of the $\widetilde{H}_i$'s follows from the fact that each V_i has $\mathbb{F}_q$-dimension $\frac{n}{k}$ and the injectivity of π.

Let W be an $\mathbb{F}_h$-subspace of $\mathbb{F}_h^m$ of dimension s and let $\{w^i = (w_1^i, \ldots, w_m^i)\}_{i=1}^s$ be an $\mathbb{F}_h$-basis of W. Put $r = \lfloor \frac{s}{1-\delta} \rfloor$ and $D = \lfloor \frac{sd(m-r+1)}{r} \rfloor$. Then one can verify that

$$D + \delta dm < d(m - r + 1) . \tag{8.6}$$

Claim 8.4.7. *There are polynomials $A_0(X), \ldots, A_{r-1}(X) \in \mathbb{F}_q[X]$ of degree at most D that are not all zero such that for all $w = (w_1, w_2, \ldots, w_m) \in W$, we have*

$$A_0(P^{\sigma^j}) w_{j+1} + A_1(P^{\sigma^j}) w_{j+2} + \cdots + A_{r-1}(P^{\sigma^j}) w_{j+r} \tag{8.7}$$

for all $j = 0, 1, \ldots, m - r$.

[7] Note that in Section 8.3 we wrote g^σ to denote the polynomial obtained by applying σ to the coefficients of g. We hope that this notation does not cause any confusion.

Proof of Claim 8.4.7. Consider the interpolation polynomial

$$R(X, Z_1, \ldots, Z_r) := A_0(X)Z_1 + A_1(X)Z_2 + \cdots + A_{r-1}(X)Z_r \ ,$$

where each $A_i(X) \in \mathbb{F}_q[X]$ has degree at most D. Consider the homogeneous system of equations where the coefficients of the $A_i(X)$'s are the variables:

$$A_0(P^{\sigma^j})w_{i,j+1} + A_1(P^{\sigma^j})w_{i,j+2} + \cdots + A_{r-1}(P^{\sigma^j})w_{i,j+r} = 0 \tag{8.8}$$

for $i = 1, 2, \ldots, s$ and $j = 0, 1, \ldots, m - r$. There are $s(m - r + 1)$ equations in $\mathbb{F}_h = \mathbb{F}_{q^d}$ and $r(D + 1)$ coefficients of $A_i(X)$ in $\mathbb{F}_q$ in total. Since $r(D + 1) > sd(m - r + 1)$, we can find polynomials $A_0, A_1, \ldots, A_{r-1} \in \mathbb{F}_q[X]$ of degree at most D that are not all zero such that the identities (8.8) hold.

Now, for any $w = (w_1, w_2, \ldots, w_m) \in W$, we write $w = \sum_{i=1}^{s} a_i w^i$ for some $a_i \in \mathbb{F}_h$. By (8.8) we have

$$\begin{aligned} &A_0(P^{\sigma^j})w_{j+1} + A_1(P^{\sigma^j})w_{j+2} + \cdots + A_{r-1}(P^{\sigma^j})w_{j+r} \\ &\qquad = \sum_{i=1}^{s} a_i(A_0(P^{\sigma^j})w_{i,j+1} + A_1(P^{\sigma^j})w_{i,j+2} + \cdots + A_{r-1}(P^{\sigma^j})w_{i,j+r}) = 0 \end{aligned}$$

for $j = 0, 1, \ldots, m - r$, as desired. □

For any $(w_1, w_2, \ldots, w_m) \in W \cap \widetilde{H}_i$, there exists a function $f \in V_i$ such that $(f(P), f(P^\sigma), \ldots, f(P^{\sigma^{m-1}})) = (w_1, w_2, \ldots, w_m)$.

Claim 8.4.8. *We have*

$$A_0(Y)f(Y) + A_1(Y)f(\zeta Y) + \cdots + A_{r-1}(Y)f(\zeta^{r-1}Y) = 0 \ . \tag{8.9}$$

Proof of Claim 8.4.8. By the identities (8.7), we have

$$A_0(P^{\sigma^j})f(P^{\sigma^j}) + A_1(P^{\sigma^j})f(P^{\sigma^{j+1}}) + \cdots + A_{r-1}(P^{\sigma^j})f(P^{\sigma^{j+r-1}}) = 0$$

for $j = 0, 1, \ldots, m - r$. Recalling Fact 8.4.5, this gives

$$(A_0 f + A_1 \cdot (f^{\sigma^{-1}}) + \cdots + A_{r-1} \cdot (f^{\sigma^{-r+1}}))(P^{\sigma^j}) = 0$$

for $j = 0, 1, \ldots, m - r$. As the polynomial $(A_0 f + A_1 \cdot (f^{\sigma^{-1}}) + \cdots + A_{r-1} \cdot (f^{\sigma^{-r+1}}))$ has degree at most $D + \delta dm$ and it has $m - r + 1$ zeros at distinct places of degree d, by (8.6) we must have

$$(A_0 f + A_1 \cdot (f^{\sigma^{-1}}) + \cdots + A_{r-1} \cdot (f^{\sigma^{-r+1}})) = 0 \ .$$

Recalling the definition of σ, this implies

$$A_0(Y)f(Y) + A_1(Y)f(\zeta Y) + \cdots + A_{r-1}(Y)f(\zeta^{r-1}Y) = 0 \ .$$ □

Observe that the solutions $f \in \mathbb{F}_q[Y]_{<\delta n}$ to (8.9) form an $\mathbb{F}_q$-linear space; denote it by U. Our task now is to bound the dimension of U; our argument is analogous to that of Lemma 6 in [GW13]. Write $f(Y) = f_0 + f_1 Y + \cdots + f_{k-1} Y^{k-1}$.

Claim 8.4.9. $\dim_{\mathbb{F}_q}(U) \leq r - 1$.

Proof of Claim 8.4.9. By factoring out common powers of Y we may assume that there exists $i^* \in \{0, 1, \ldots, r-1\}$ such that A_{i^*} has a nonzero constant term. Write $A_i(Y) = a_{i,0} + a_{i,1} Y + \cdots + a_{i,D} Y^D$ for $i = 0, 1, \ldots, r-1$, and define the polynomials

$$B_j(Y) := a_{0,j} + a_{1,j} Y + \cdots + a_{r-1,j} Y^{r-1}$$

for $j = 0, 1, \ldots, k-1$. Note that our assumption on A_{i^*} states that $a_{i^*,0} \neq 0$, so B_0 is a nonzero polynomial of degree $\leq r - 1$. Let $\Lambda(Y) := A_0(Y) f(Y) + A_1(Y) f(\zeta Y) + \cdots + A_{r-1}(Y) f(\zeta^{r-1} Y)$, which is the 0 polynomial by Identity 8.9.

Note that the constant term of Λ is $a_{0,0} f_0 + a_{1,0} f_0 + \cdots + a_{r-1} f_0 = B_0(1) f_0$. Thus, assuming $B_0(1) \neq 0$ we find that $f_0 = 0$; otherwise f_0 can take an arbitrary value in $\mathbb{F}_q$.

Now fix an $\ell \in \{1, 2, \ldots, k-1\}$. The coefficient on Y^ℓ in $\Lambda(Y)$ may be expressed as $f_\ell B_0(\zeta^\ell) + f_{\ell-1} B_1(\zeta^{\ell-1}) + \cdots + f_1 B_{\ell-1}(\zeta) + f_0 B_\ell(1)$. As $\Lambda \equiv 0$, this linear form must equal 0. The crucial observation is that, assuming $B_0(\zeta^\ell) \neq 0$, once $f_0, \ldots, f_{\ell-1}$ are fixed there is a unique choice for $f_\ell \in \mathbb{F}_q$ such that this linear form is 0 (otherwise $f_\ell \in \mathbb{F}_q$ is unconstrained). We therefore obtain that the dimension of U is at most the number of $0 \leq \ell \leq k-1$ for which $B_0(\zeta^\ell) = 0$. As ζ is primitive and $k \leq q$, the elements ζ^ℓ for $\ell = 0, 1, \ldots, k-1$ are distinct. As B_0 is a nonzero polynomial of degree $\leq r - 1$ we find that there can be at most $r - 1$ values of ℓ such that $B_0(\zeta^\ell) = 0$. This implies that $\dim_{\mathbb{F}_q} U \leq r - 1$. □

It is clear that $\pi^{-1}(\widetilde{H}_i \cap W) \subseteq V_i \cap U$ for $i = 1, 2, \ldots, k$. Thus, we have

$$\begin{aligned}\sum_{i=1}^{k} \dim_{\mathbb{F}_q}(\widetilde{H}_i \cap W) \leq \sum_{i=1}^{k} \dim_{\mathbb{F}_q}(V_i \cap U) &\leq \frac{r(\delta dm - 1)}{\delta dm - \frac{dm}{k} - 2r + 2} \\ &\leq \frac{\delta}{1-\delta} \cdot \frac{m}{(\delta - \frac{1}{k})m - \frac{2s}{d(1-\delta)}} \cdot s \,.\end{aligned}$$

This demonstrates that $\widetilde{H}_1, \ldots, \widetilde{H}_k$ provide a subspace design with the claimed parameters. □

Obtaining from Proposition 8.4.6 an *explicit* construction is a bit nontrivial, as there is no known deterministic algorithm to find irreducible polynomials of a given input degree. However, a simple approach is to assume $n = q - 1$ and take the polynomial $P(Y) = Y^d - \zeta^{-1}$, where we recall that ζ^{-1} is a primitive root of $\mathbb{F}_q$. Note that finding such a primitive root can be done in $\mathrm{poly}(q)$ time by brute force. That $P(Y)$ is irreducible follows from the following proposition.

Proposition 8.4.10 ([LN97], Chapter 3). *Let $d \geq 2$ be an integer and $\alpha \in \mathbb{F}_q \setminus \{0\}$. Then the binomial $X^d - \alpha$ is irreducible in $\mathbb{F}_q[X]$ iff the following conditions hold:*

1. *Each prime factor of d divides $\mathrm{ord}_{\mathbb{F}_q}(\alpha)$ and $\gcd(d, \frac{q-1}{\mathrm{ord}_{\mathbb{F}_q}(\alpha)}) = 1$;*
2. *$q \equiv 1 \pmod 4$ if $d \equiv 0 \pmod 4$.*

Moreover the polynomials $P(Y)^{\sigma^j} = P(\zeta^j Y) = (\zeta^j Y)^d - \zeta^{-1} = \zeta^{jd}(Y^d - \zeta^{-(jd+1)})$ are also irreducible and pairwise coprime (as $j < m \leq n/d = (q-1)/d$). Finally evaluating a polynomial f at the place P^{σ^j}, which amounts to reducing the polynomial modulo $Y^d - \zeta^{-(j+1)}$, can be done in $\mathrm{poly}(n)$ field operations. Thus, given bases $\mathcal{B}_1, \ldots, \mathcal{B}_k$ for $V_1, \ldots, V_k$, we obtain the bases for $\widetilde{H}_i$ by evaluating π on each element of $\mathcal{B}_i$, respectively.

Summarizing the above discussion, we conclude:

Proposition 8.4.11. *If $n = q - 1$ and $s < (1-\delta)m = (1-\delta)\frac{n}{d}$, then there exist $\tilde{H}_1, \ldots, \tilde{H}_k$ forming an explicit $(s, \frac{\delta}{1-\delta} \cdot \frac{m}{(\delta - \frac{1}{k})m - \frac{2s}{d(1-\delta)}}, d)$-subspace design in $\mathbb{F}_h^m$. Moreover $\dim_{\mathbb{F}_q} \widetilde{H}_i = \frac{n}{k}$ for all $i = 1, \ldots, k$.*

Setting parameters. By choosing k and d appropriately we obtain the following corollary.

Corollary 8.4.12. *Let $\delta > 0$ be such that $1/\delta \in \mathbb{Z}$ and put $k = 1/\delta^2$, $d = 1/\delta^3$. Assume that $q - 1 = n$. There exist $H_1, \ldots, H_k$ which form an explicit $(s, \frac{1}{1-2\delta-\delta^2+2\delta^3}, d)$-subspace design in $\mathbb{F}_{q^n}$ for all $s \leq \frac{1-2\delta}{d} n$. Moreover $\dim_{\mathbb{F}_q} H_i = \frac{n}{k}$ for all $i = 1, \ldots, k$.*

Proof. Fix an $\mathbb{F}_h$-linear isomorphism $\varphi : \mathbb{F}_h^m \to \mathbb{F}_{q^n}$ and define $H_i = \varphi(\widetilde{H}_i)$ for $i = 1, 2, \ldots, k$, where $\widetilde{H}_1, \widetilde{H}_2, \ldots, \widetilde{H}_k \subseteq \mathbb{F}_h^m$ form the subspace design promised in Proposition 8.4.11. Since φ is also $\mathbb{F}_q$-linear, the dimensions of $\mathbb{F}_q$-subspaces are also preserved by φ. Then, if $W \subseteq \mathbb{F}_{q^n}$ is an $\mathbb{F}_h$-subspace,

$$\sum_{i=1}^{k} \dim_{\mathbb{F}_q}(H_i \cap W) = \sum_{i=1}^{k} \dim_{\mathbb{F}_q}(\widetilde{H}_i \cap \varphi^{-1}(W))$$

so $H_1, \ldots, H_k$ forms a subspace design in $\mathbb{F}_{q^n}$ with the same parameters as $\widetilde{H}_1, \ldots, \widetilde{H}_k$. That $H_1, \ldots, H_k$ are explicit follows easily from the explicitness of $\widetilde{H}_1, \ldots, \widetilde{H}_k$.

Assuming $s \leq \frac{1-2\delta}{d} n < (1-\delta)m$, we find

$$\begin{aligned} \frac{\delta}{1-\delta} \cdot \frac{m}{(\delta - \frac{1}{k})m - \frac{2s}{d(1-\delta)}} &\leq \frac{\delta}{1-\delta} \cdot \frac{m}{\delta(1-\delta)m - \frac{2(1-\delta)\delta^3 m}{1-\delta}} \\ &= \frac{1}{(1-\delta)^2} \cdot \frac{1}{1 - \frac{2\delta^2}{1-\delta}} = \frac{1}{(1-\delta)^2 - 2\delta^2(1-\delta)} = \frac{1}{1 - 2\delta - \delta^2 + 2\delta^3}. \end{aligned}$$

The result now follows from Proposition 8.4.11. □

8.5 Explicit Instantiations of Dimension Expanders

As outlined in Section 8.3, our approach for obtaining explicit constructions of dimension expanders is by reducing to the construction of subspace designs. Specifically, we will will apply Theorem 8.3.4 with the constructions of Section 8.4. These results yield Theorems 8.1.2 and 8.1.1, respectively.

First, using the subspace design constructed in Corollary 8.4.4, we obtain a degree-proportional dimension expander over fields of arbitrarily small polynomial size.

Theorem 8.5.1. *Let $\delta > 0$ be given and assume $|\mathbb{F}_q| \geq n^{\delta}$. Let r be an integer satisfying $\frac{1}{2\delta} \leq r < \frac{1}{\delta}$, let k be a multiple of r, and let $d = 2k$. There exists an explicit construction of a (η, β)-dimension expander of degree d over $\mathbb{F}_q^n$ whenever $2dr|n$, where $\eta = \Omega\left(\frac{1}{\delta d}\right)$ and $\beta = \Omega(\delta d)$.*

Proof. Using Corollary 8.4.4, we have an explicit (s, A, d)-subspace design $\{H_i\}_{i=1}^k$ for all $s \leq \mu n$, where $\mu = \frac{1-2\delta}{4d}$ and $A = \frac{8}{\delta}$. Moreover $\dim_{\mathbb{F}_q} H_i = \frac{n}{k}$ for all $i = 1, \ldots, k$. Recall that $d = 2k$, so $d - k + 1 \geq d/2$. Thus, Theorem 8.3.4 implies that we have an explicit (η, β)-dimension expander for

$$\eta = \mu A = \frac{1-2\delta}{4d} \cdot \frac{8}{\delta} = 2(1-2\delta) \cdot \frac{1}{\delta d} = \Omega\left(\tfrac{1}{\delta d}\right)$$

and

$$\beta = \frac{d-k+1}{A} \geq \frac{d/2}{8/\delta} = \frac{1}{16} \cdot \delta d = \Omega(\delta d) \; . \qquad \square$$

Next, we use the subspace design constructed in Corollary 8.4.12 to obtain an explicit construction of a lossless dimension expander. We remark that the construction achieves $d = O(1/\varepsilon^3)$ while the probabilistic argument demonstrates that $d = O(1/\varepsilon^2)$ suffices. Thus, we are just a factor of ε away from the randomized construction.

Theorem 8.5.2. *Fix $\varepsilon > 0$, and choose $\delta = \Theta(\varepsilon)$ sufficiently small and such that $1/\delta \in \mathbb{Z}$. Let $d = 1/\delta^3$ and $k = 1/\delta^2$ and assume that $q - 1 = n$ and $d|n$. Then there exists an explicit construction of a $(\frac{1-\varepsilon}{d}, (1-\varepsilon)d)$-dimension expander with degree d over $\mathbb{F}_q^n$.*

Proof. Using Corollary 8.4.12, there exists a collection $\{H_i\}_{i=1}^k$ forming a (s, A, d) subspace design for all $s \leq (1-2\delta)m = \frac{1-2\delta}{d}n$, where

$$A = \frac{1}{1 - 2\delta - \delta^2 + 2\delta^3} \; .$$

Hence, by Theorem 8.3.4, using the fact that $d - k \geq d(1-\delta)$ we obtain the expansion factor

$$\beta = \frac{d-k+1}{A} \geq d(1-\delta)(1 - 2\delta - \delta^2 + 2\delta^3) \; .$$

By assuming $\delta \leq \varepsilon/4$, this is $\geq (1-\varepsilon)d$, as desired. The lower bound on η is obtained by plugging in $(1-2\delta)/d$ for μ in Theorem 8.3.4:

$$\eta = \mu A \geq \mu = \frac{1-2\delta}{d} \geq \frac{1-\varepsilon}{d} \; . \qquad \square$$

8.6 Unbalanced Expanders

For clarity's sake, we have presented all our results in the context of balanced dimension expanders. However, as remarked earlier, our techniques are flexible enough to produce *unbalanced* dimension expanders. In this section, we state the appropriate generalizations of our results that are required to construct unbalanced dimension expanders. As the proofs are extremely similar to those given before, we do not provide full proofs, but merely indicate the details that need to be changed.

We recall Definition 8.2.3: a b-unbalanced (η, β)-dimension expander of degree d is a collection $\Gamma_1, \ldots, \Gamma_d : \mathbb{F}^N \to \mathbb{F}^n$ of linear maps such that for any $V \leq \mathbb{F}^N$ of dimension at most ηN, $\dim \sum_j \Gamma_j(V) \geq \beta \dim V$. We also recall that $b = \frac{N}{n}$, which we assume to be an integer.

8.6.1 Unbalanced Dimension Expander Construction

In this subsection, we provide the appropriate generalizations of the results of Section 8.3.

Construction. Recall that the dimension expanders map $\mathbb{F}_q^N \to \mathbb{F}_q^n$. We view the domain as

$$\mathcal{F} = \left\{ f(X) = \sum_{i=0}^{k-1} f_i X^{q^i} : f_i \in H_i,\ i = 0, \ldots, k-1 \right\}$$

where $H_0, \ldots, H_{k-1}$ give a collection of $\mathbb{F}_q$-subspaces of $\mathbb{F}_{q^n}$, each of $\mathbb{F}_q$-dimension $\frac{N}{k}$. Thus, we now require $k|N$. As before, $H_0, \ldots, H_{k-1}$ will form a subspace design. We view the image space as $\mathbb{F}_{q^n}$. Again $h = q^d$ and $\alpha_1, \ldots, \alpha_d$ gives a basis for $\mathbb{F}_h/\mathbb{F}_q$. The definition of Γ_j is just as before:

$$\Gamma_j : \mathcal{F} \to \mathbb{F}_{q^n};\ \ f \mapsto f(\alpha_j)\ .$$

Analysis. The statements of Lemmas 8.3.1, 8.3.2 and 8.3.3 remain valid. Thus, we may conclude:

Theorem 8.6.1. *Let $\{H_i\}_{i=0}^{k-1}$ give a (s, A, d)-subspace design in $\mathbb{F}_{q^n}$ for all $s \leq \mu N$ for some $\mu \in (0, \frac{1}{bd})$. Then $\{\Gamma_j\}_{j=1}^{d}$ is a b-unbalanced $(\mu A, \frac{d-k+1}{A})$-dimension expander.*

The only detail which has changed from Theorem 8.3.4 is that now $\mu < \frac{1}{bd}$, rather than just $\mu < \frac{1}{d}$. Asides from this, the proof proceeds identically to before, appealing now to Proposition 8.2.4 instead of Proposition 8.2.2.

8.6.2 Higher-Dimensional Subspace Designs

In this section we construct subspace designs $H_1, \ldots, H_k \subseteq \mathbb{F}_{q^n}$, where the H_i's now have $\mathbb{F}_q$-dimension $\frac{N}{k}$.

Subspace Designs via an Intermediate Field

First, note the proof of Proposition 8.4.3 still applies in this scenario. Essentially, it suffices to adjust the definition of the t parameter so as to ensure that the subspaces have dimension $\frac{N}{k}$.

Proposition 8.6.2. *Let $\ell = q^c$ with $c = \frac{d}{k} \cdot \frac{bm}{m-2t}$, where $1 \leq k < d$. For all $1 \leq s < t < \ell$ such that $\ell^2 \geq 4kt$, $k|d$, $mb|(m-2t)k$ and $k(m-2t)|N$, there is an explicit construction of $\{H_i\}_{i=1}^k$ that forms a $(s, \frac{d}{k} \cdot \frac{bm}{m-2t} \cdot \frac{m-1}{2(t-s)}, d)$-subspace design in $\mathbb{F}_{q^n}$. Moreover $\dim_{\mathbb{F}_q} H_i = \frac{N}{k}$ for all $i = 1, \ldots, k$.*

We now fix parameters to obtain subspace designs over fields of size n^δ.

Corollary 8.6.3. *Let $\delta > 0$ be given and choose an integer r such that $\frac{1}{2\delta} \leq r < \frac{1}{\delta}$. We assume $\delta > 0$ is sufficiently small so that $r \geq \max\{b, 2\}$. Let k, d be integers such that $d = 2k$ and $r|k$. Assume moreover that $2r|mb$. Then, assuming $q \geq n^\delta$, there exists an explicit construction of $\{H_i\}_{i=1}^k$ that forms a $(s, \frac{8}{\delta}, d)$-subspace design in $\mathbb{F}_{q^n}$ for all $s \leq \frac{1-2\delta b}{4bd} N$. Moreover $\dim_{\mathbb{F}_q} H_i = \frac{N}{k}$ for all $i = 1, \ldots, k$.*

Proof Sketch. The proof proceeds very similarly to the proof of Corollary 8.4.4; we just define the appropriate parameters. Set $t = \frac{1}{2}(1 - \frac{b}{r})m = \frac{1}{2db}(1 - \frac{b}{r})N$ and assume $s \leq t/2$. Thus we may take $\mu = \frac{1-2\delta b}{4db} \leq \frac{1-b/r}{4db}$ and A is bounded by

$$\frac{d}{k} \cdot \frac{bm}{m-2t} \cdot \frac{m}{2(t-s)} \leq 2r \frac{2}{1-\frac{1}{r}} \leq \frac{8}{\delta} .$$ □

Subspace Designs via Correlated High-Degree Places

The results in this section follow from the same arguments as those provided in Section 8.4.2, except now we will set $\delta = \sqrt{\frac{b}{k}}$ and insist that the $V_1, \ldots, V_k \subseteq \mathbb{F}_q[X]_{<\delta n}$ are chosen to have codimension $\delta n - \frac{N}{k}$.

Proposition 8.6.4. *If $n = q - 1$ and $s < (1-\delta)m = (1-\delta)\frac{N}{bd}$, then there exists $\tilde{H}_1, \ldots, \tilde{H}_k$ forming an explicit $(s, \frac{\delta}{1-\delta} \cdot \frac{m}{(\delta - \frac{1}{bk})m - \frac{2s}{d(1-\delta)}}, d)$-subspace design in $\mathbb{F}_h^m$. Moreover $\dim_{\mathbb{F}_q} \tilde{H}_i = \frac{N}{k}$ for all $i = 1, \ldots, k$.*

Corollary 8.6.5. *Let $\delta > 0$ be such that $1/\delta \in \mathbb{Z}$ and put $k = b/\delta^2$, $d = b/\delta^3$. Assume that $n = q - 1$ and $d|n$. There exist $H_1, \ldots, H_k$ which form an explicit $(s, \frac{1}{1-2\delta-\delta^2+\delta^3}, d)$-subspace design in $\mathbb{F}_{q^n}$ for all $s \leq \frac{1-2\delta}{db} N$. Moreover $\dim_{\mathbb{F}_q} H_i = \frac{N}{k}$ for all $i = 1, \ldots, k$.*

8.6.3 Explicit Instantiations

Finally, we provide the analogous results to those obtained in Section 8.5. These yield Theorems 8.1.4 and 8.1.3, respectively.

First, instantiating Theorem 8.6.1 with Corollary 8.6.3 yields the following.

Theorem 8.6.6. *Let $\delta > 0$ (sufficiently small) be given and assume $q \geq n^{\delta}$. Let r be an integer in the range $(\frac{1}{2\delta}, \frac{1}{\delta})$, choose a multiple k of r, and let $d = 2k$. Let b be an integer. There exists an explicit construction of a b-unbalanced (η, β)-dimension expander of degree d over $\mathbb{F}_q^n$ whenever $2dr|nb$, where $\eta = \Omega\left(\frac{1}{\delta bd}\right)$ and $\beta = \Omega(\delta d)$.*

Secondly, appealing to Corollary 8.6.5 instead, we obtain the following.

Theorem 8.6.7. *Fix $\varepsilon > 0$ sufficiently small, and choose $\delta = \Theta(\varepsilon)$ sufficiently small and such that $1/\delta \in \mathbb{Z}$. Let $k = b/\delta^2$ and $d = b/\delta^3$. Suppose that $n = q - 1$ and $d|n$. Then there exists an explicit construction of a $(\frac{1-\varepsilon}{bd}, (1-\varepsilon)d)$-dimension expander of degree d over $\mathbb{F}_q$.*

As before, we remark that $d = O(b/\varepsilon^3)$, whereas the randomized construction achieves $d = O(b/\varepsilon^2)$. Moreover, η is again optimal: subspaces of dimension greater than $\frac{N}{bd} = \frac{n}{d}$ cannot be expanded by a fact of $\approx d$.

8.7 Subspace Evasive Subspaces

Recall the discussion from Section 8.1.3: we wish to find a subspaces $H_1, \ldots, H_k$ which have small total intersection with subspaces W which are linear over $\mathbb{F}_{q^d} =: \mathbb{F}_h$. While we have the freedom of choosing the H_i's to be distinct subspaces, we observe that there exists a single subspace that has small intersection with all such subspaces W! That is, it is possible to take $H_1 = \cdots = H_k =: H$, and still obtain a good subspace design. We call such an H a *subspace evasive subspace*.

Moreover, we show that by taking a subspace evasive subspace with parameters matching those achievable by a random subspace, we may obtain *degree-proportional* dimension expanders. We find this observation rather surprising, and also demonstrative of the efficiency of our reduction from dimension expanders to subspace designs.

We begin with the definition germane to this section.

Definition 8.7.1 (Subspace Evasive Subspace)**.** An $\mathbb{F}_q$-subspace $H \subseteq \mathbb{F}_{q^n}$ is called a (s, A, d)-subspace evasive subspace if for every $\mathbb{F}_{q^d}$-linear subspace $W \subseteq \mathbb{F}_{q^n}$ of dimension s,

$$\dim_{\mathbb{F}_q}(H \cap W) \leq As \; .$$

We first observe that subspace evasive subspaces naturally yield subspace designs, although the A parameter degrades by a factor of k.

Observation 8.7.2. *Suppose H is (s, A, d)-evasive. The tuple $(H, H, \ldots, H)$, repeated k times, forms a (s, kA, d)-subspace design.*

The following proposition demonstrates that good subspace evasive subspaces exist.

Proposition 8.7.3. *Let $k, d, n > 2$ be positive integers such that $q^{n/4} \geq m = n/d$ and $k < d$. Let $\boldsymbol{H}$ be a random $\mathbb{F}_q$-subspace of $\mathbb{F}_{q^n} \cong \mathbb{F}_q^n$ of dimension n/k. Then, with probability at least $1 - q^{-\Omega(n)}$, for every $\mathbb{F}_h$-subspace W of $\mathbb{F}_{q^n}$ with $\dim_{\mathbb{F}_h}(W) \leq \frac{m}{4} = \frac{n}{4d}$,*

$$\dim_{\mathbb{F}_q}(W \cap \boldsymbol{H}) \leq \frac{\dim_{\mathbb{F}_h}(W)}{1 - 2/k} \; .$$

That is, $\boldsymbol{H}$ is $(s, \frac{1}{1-2/k}, d)$-evasive for all $s \leq \frac{m}{4} = \frac{n}{4d}$.

Proof. The probability that a fixed set of L vectors that are linearly independent over $\mathbb{F}_q$ belong to $\boldsymbol{H}$ is at most

$$\left(\frac{q^{n/k}}{q^n}\right)^L = q^{-n(1-1/k)L} .$$

By a union bound, the probability that some $\mathbb{F}_h$-subspace of dimension s has at least L such vectors belong to $\boldsymbol{H}$ is at most

$$h^{ms} \cdot h^{sL} \cdot q^{-n(1-1/k)L} = q^{sn} \cdot q^{-L(n(1-1/k)-sd)} . \tag{8.10}$$

Assuming $s \leq \frac{n}{4d}$ and taking $L \geq s/(1-2/k)$, recalling that $k \geq 3$, we have

$$q^{-L(n(1-1/k)-sd)} \leq q^{-\frac{s}{1-2/k}(n(1-1/k)-n/4)} \leq q^{-3s(2n/3-n/4)} = q^{-\frac{5}{4}ns} .$$

Thus, (8.10) is at most $q^{ns}q^{-\frac{5}{4}ns} = q^{-\frac{1}{4}ns}$. Summing up over all s, $1 \leq s \leq m/4$, we get that the desired claim holds for all subspaces $W \subseteq \mathbb{F}_{h^m}$ with $\dim_{\mathbb{F}_h}(W) \leq m/4$ except with probability at most $m \cdot q^{-n/4} \leq q^{-n/8}$. □

We now set $k = 3$ and apply Theorem 8.3.4 with this H to obtain a degree-proportional dimension expander. We remark that we may even have $\eta \geq 1/d$.

Proposition 8.7.4. *Let n, d be integers with $3|n$, $d|n$ and $3 < d$. Let H be the subspace evasive subspace promised by Proposition 8.7.3, and let $\{\Gamma_j\}_{j=1}^d$ denote the dimension expander constructed as in Section 8.3 with each $H_i = H$. Then $\{\Gamma_j\}_{j=1}^d$ forms a degree-proportional dimension expander.*

Proof. By combining Proposition 8.7.3 and Observation 8.7.2, we have that (H, H, H) forms a $(s, 3A, d)$-subspace design for all $s \leq \mu n$, for $A = 3$ and $\mu = \frac{1}{4d}$. Applying Theorem 8.3.4, this implies that $\Gamma_1, \ldots, \Gamma_d$ form an (η, β)-dimension expander for $\eta = \mu \cdot 3 \cdot A = \frac{1}{4d} \cdot 3 \cdot 3 = \frac{9}{4d}$ and $\beta = \frac{d-k+1}{3A} = \frac{d-2}{9}$. □

8.8 Conclusion and Open Problems

In this work we provide the first explicit construction of a lossless dimension expander. Our construction uses ideas from recent constructions of list-recoverable rank-metric codes, which is in analogy with the approach taken by [GUV09] in the "Boolean" world. Our approach is sufficiently general to achieve lossless expansion even in the case that the expander is "unbalanced", i.e., when the codomain has dimension smaller than the domain.

The main open problem that remains is to achieve similar constructions over fields of smaller size. Our construction of lossless expanders requires fields of size $q > n$, whereas our construction of degree-proportional expanders requires fields of size n^δ for arbitrarily small (constant) δ. The constraints on the field size arise largely from the

constructions of subspace designs that we employed. Thus, we believe that a fruitful avenue of attack on this problem would be to obtain constructions of subspace designs over smaller fields.[8]

The authors of [GXY18] addressed precisely this challenge. In this work the authors do manage to construct subspace designs over all fields, but the intersection size now grows with $\log_q n$. If $q = O(1)$, then instantiating our approach with these subspace designs only guarantees expansion if the degree is logarithmic. One could also have q grow polynomially with n and achieve degree-proportional expanders, but as this does not improve over the intermediate fields approach of Section 8.4.1 we have not included it.

Lastly, we recall that our construction of a $(\frac{1-\varepsilon}{d}, (1-\varepsilon)d)$-dimension expander had degree $d = \Theta(1/\varepsilon^3)$, while the probabilistic argument shows $d = O(1/\varepsilon^2)$ is sufficient. Moreover if one is satisfied with a $(\frac{1}{2d}, (1-\varepsilon)d)$-dimension expander then it is sufficient to have $d = O(1/\varepsilon)$. Thus, constructing lossless expanders whose degree has even better dependence on ε would also be interesting.

8.9 Deferred Proofs

In this section we provide certain proofs that were deferred from the main body.

8.9.1 Proof of Lemma 8.4.2

In this subsection we provide a proof of Lemma 8.4.2, which we restate for convenience. Recall that $\ell = q^c$ and $c|d$, so $\mathbb{F}_q \subseteq \mathbb{F}_\ell \subseteq \mathbb{F}_h$. Also, $\omega_1, \ldots, \omega_m$ denotes a basis for $\mathbb{F}_{h^m}/\mathbb{F}_h$ and we define

$$L := \left\{ \sum_{i=1}^{m} a_i \omega_i : a_1, \ldots, a_m \in \mathbb{F}_\ell \right\} .$$

Lemma 8.9.1. *Let W be an $\mathbb{F}_h$-subspace of $\mathbb{F}_{q^n}$ and let $U := W \cap L$. Then U is an $\mathbb{F}_\ell$-subspace of L and $\dim_{\mathbb{F}_\ell} U \leq \dim_{\mathbb{F}_h} W$.*

Proof. It is clear that U is an $\mathbb{F}_\ell$ subspace as $\mathbb{F}_h \supseteq \mathbb{F}_\ell$. Suppose $u_1, \ldots, u_t \in U$ are linearly independent over $\mathbb{F}_\ell$; we will show that they are also linearly independent over $\mathbb{F}_h$. Once we have shown this, the lemma follows.

Put $r = d/c$ and let $\gamma_1, \ldots, \gamma_r$ denote a basis for $\mathbb{F}_h/\mathbb{F}_\ell$. Suppose that $\sum_{k=1}^{t} a_k u_k = 0$ with $a_1, \ldots, a_t \in \mathbb{F}_h$; we want to show $a_1 = \cdots = a_t = 0$. Using our bases, we may write

[8] In [GK16] there is also an "extension field" construction that allows for smaller field sizes, but only guarantees the existence of "weak" subspace designs, which does not suffice for the dimension expander application.

$a_k = \sum_{j=1}^{r} b_{jk}\gamma_j$ and $u_k = \sum_{i=1}^{m} c_{ki}\omega_i$ for $b_{jk}, c_{ki} \in \mathbb{F}_\ell$. Thus, we have

$$\sum_{k=1}^{t} \left(\sum_{j=1}^{r} b_{jk}\gamma_j \right) \left(\sum_{i=1}^{m} c_{ki}\omega_i \right) = 0$$

which, upon rearranging, becomes

$$\sum_{i=1}^{m} \left(\sum_{j=1}^{r} \left(\sum_{k=1}^{t} b_{jk}c_{ki} \right) \gamma_j \right) \omega_i = 0 \ .$$

Since $\omega_1, \ldots, \omega_m$ form a basis for $\mathbb{F}_{q^n}/\mathbb{F}_h$ and $\sum_{j=1}^{r} \left(\sum_{k=1}^{t} b_{jk}c_{ki} \right) \gamma_j \in \mathbb{F}_h$ for all $i \in [m]$, we deduce

$$\sum_{j=1}^{r} \left(\sum_{k=1}^{t} b_{jk}c_{ki} \right) \gamma_j = 0 \ \ \forall i \in [m] \ .$$

Next, since $\gamma_1, \ldots, \gamma_r$ form a basis for $\mathbb{F}_h/\mathbb{F}_\ell$ and $\sum_{k=1}^{t} b_{jk}c_{ki} \in \mathbb{F}_\ell$ for all $j \in [r]$, we deduce

$$\sum_{k=1}^{t} b_{jk}c_{ki} = 0 \ \ \forall i \in [m], \ j \in [r] \ .$$

Thus, defining the matrices $B = (b_{jk}) \in \mathbb{F}_\ell^{r \times t}$ and $C = (c_{ki}) \in \mathbb{F}_\ell^{t \times m}$, we find $BC = 0$ (where 0 denotes the $r \times m$ matrix of all zeroes). Moreover, since $u_1, \ldots, u_t$ are assumed to be $\mathbb{F}_\ell$-linearly independent it follows that the matrix C has full-rank, i.e., $\mathrm{rank}(C) = t$. We therefore have $0 = \mathrm{rank}(BC) = \mathrm{rank}(B)$, i.e., B must be the $r \times t$ matrix of zeroes. This shows that $a_1 = \cdots = a_t = 0$, as desired. □

8.9.2 Randomized Construction of an Unbalanced Dimension Expander

In this section, we prove Proposition 8.2.5, demonstrating that good unbalanced lossless dimension expanders exist. Our argument is modeled after Section C.2 in [FG15], wherein it is shown that good (balanced) dimension expanders exist. As is standard in the theory of pseudorandomness, our existential argument uses the probabilistic method.

For easy reference, we state lemmas bounding the probability a random matrix has low rank, as well as a bound on the number of subspaces. These estimates follow from our upper bounds on the size of rank metric balls (Proposition 5.1.4) and the q-nomial coefficient (Lemma 5.5.3), respectively.

Lemma 8.9.2. *Let $\boldsymbol{M}$ be a uniformly random matrix in $\mathbb{F}_q^{n \times N}$. The probability that $\mathrm{rank}(\boldsymbol{M}) \leq r$ is at most*

$$4q^{-(N-r)(n-r)} \ .$$

Lemma 8.9.3. *The number of subspaces $V \leq \mathbb{F}_q^n$ of dimension k is at most*

$$4q^{k(n-k)} \ .$$

Lemma 8.9.4. *Let q be a prime power and assume $N, n \geq t \geq r \geq 1$. Let $\mathbf{\Gamma}_1, \dots, \mathbf{\Gamma}_d$ be independent random matrices, uniformly distributed over $\mathbb{F}_q^{n\times N}$. Then with probability at least $1 - q^r$, for any subspace $V \subseteq \mathbb{F}_q^N$ of dimension r we have*

$$\dim \sum_{j=1}^{d} \mathbf{\Gamma}_j(V) \geq t \,,$$

assuming

$$d \geq \frac{t-1}{r} + \frac{N-r+1}{n-t+1} + \frac{\log_q 16}{r(n-t+1)} \,.$$

Proof. Fix a subspace $V \subseteq \mathbb{F}_q^N$ of dimension r, and let $M \in \mathbb{F}_q^{N\times r}$ be a matrix whose columns give a basis for V. Thus, $\operatorname{rank}(M) = r$ and the column span of M is V. In particular, $\dim \sum_j \mathbf{\Gamma}_j(V) \geq t$ iff the $\mathbb{F}_q^{n\times rd}$ block matrix

$$A(V) := [\mathbf{\Gamma}_1 M | \cdots | \mathbf{\Gamma}_d M]$$

has rank at least t. As M has rank r and the $\mathbf{\Gamma}_j$ are uniformly random, the matrix $A(V)$ is a uniformly random matrix in $\mathbb{F}_q^{n\times rd}$. Thus, the probability it has rank at most $t-1$ is at most

$$4q^{-(rd-t+1)(n-t+1)} \,.$$

Then, taking a union bound over the choice of V, we see that the probability of failure is at most

$$16q^{r(N-r)-(rd-t+1)(n-t+1)} \,.$$

This is at most q^{-r} assuming

$$(n-t+1)(rd-t+1) \geq r(N-r+1) + \log_q 16 \,.$$

Dividing both sides by $r(n-t+1)$ and rearranging, the previous inequality is equivalent to

$$d \geq \frac{t-1}{r} + \frac{N-r+1}{n-t+1} + \frac{\log_q 16}{r(n-t+1)} \,. \qquad \square$$

The existential proof will be complete upon taking a union bound over the choice of r; the following proposition does exactly this.

Proposition 8.9.5. *Let $\mathbb{F}_q$ be a finite field and assume $N, n \geq 1$ and put $b = \frac{N}{n}$. Let $\beta > 1$ and $\eta \in (0, \frac{1}{b\beta})$. Then there exists a collection of matrices $\{\Gamma_1, \dots, \Gamma_d\} \subseteq \mathbb{F}_q^{n\times N}$ forming a (η, β)-dimension expander of degree d, assuming*

$$d \geq \beta + \frac{b}{1 - b\beta\eta} + \log_q 16 \,.$$

Proof. Fix any $r \leq \eta N$ and let $\mathbf{\Gamma}_1, \dots, \mathbf{\Gamma}_d \in \mathbb{F}_q^{n\times N}$ be independent and uniform; we wish to show $\dim \sum_j \mathbf{\Gamma}_j(V) \geq \beta \dim V$ for any $V \subseteq \mathbb{F}_q^N$ of dimension r with positive probility.

That is, we wish to show $\dim \sum_j \Gamma_j(V) \geq \lceil \beta r \rceil$ with positive probability. For any fixed r, Lemma 8.9.4 promises that this occurs with probability $\geq 1 - q^{-r}$ assuming

$$d \geq \frac{\lceil \beta r \rceil - 1}{r} + \frac{N - r + 1}{n - \lceil \beta r \rceil + 1} + \frac{\log_q 16}{r(n - r + 1)} \, .$$

As $\lceil \beta r \rceil - 1 \leq \beta r$ and $r(n - r + 1) \geq 1$, it actually suffices for

$$d \geq \beta + \frac{N}{n - \beta r} + \log_q 16 \, .$$

Recalling $r \leq \eta N = \eta b n$ and $b = \frac{N}{n}$, we see that it suffices to have

$$d \geq \beta + \frac{b}{1 - b\beta\eta} + \log_q 16 \, ,$$

as stated. Now, as $\sum_{r=1}^{\lceil \eta N \rceil} q^{-r} \leq \sum_{r=1}^{\infty} q^{-r} < 1$, we can take a union bound over the choice of r to conclude that there exists a realization of the $\mathbf{\Gamma}_j$'s which indeed forms a (η, β)-dimension expander. $\square$

Chapter 9

Conclusion

In this thesis, we have expanded the frontiers of our knowledge of list-decodable codes and friends. In Chapters 3 through 6, we considered random ensembles of codes, with a particular focus on random linear codes. Some notable punchlines include:

- Every local property, i.e., any property defined by excluding a family of constant-sized types, has a sharp threshold for random linear codes.
- Random LDPC codes achieve list-decoding capacity with high probability.
- To list-decode random linear rank metric codes ε-away from capacity, lists of size $O_{\rho,q}(1/\varepsilon)$ are sufficent.
- For random linear binary codes ε-away from capacity lists of size $O(1/\varepsilon)$ are sufficient for average-radius list-decoding in either the Hamming or rank metrics.

Later, in Chapter 7, we provided an explicit construction of a code with a near-linear time global list-decoding algorithm; and an explicit construction of a code with a sub-linear time *local* list-decoding algorithm.

Finally, in Chapter 8, we showed how to leverage ideas used to algorithmically list-decode rank metric codes to construct *lossless* dimension expanders.

Alas, despite our best efforts, many problems remain open.[1] In the remainder of this chapter we expound upon a few problems that we find particularly stimulating.

9.1 Precisely Computing the Threshold for List-Decodability

Recalling the notations from Chapter 3, if $\mathcal{P}_n$ is the property of (ρ, L)-list-decodability for codes of block length n, we wish to understand $R_{\mathrm{RLC}}(\mathcal{P}_n)$. The Zyablov-Pinsker argument shows $R_{\mathrm{RLC}}(\mathcal{P}_n) \geq 1 - h_q(\rho) - O\left(\frac{1}{\log_q L}\right)$; however, we believe that in fact $R_{\mathrm{RLC}}(\mathcal{P}_n) \geq 1 - h_q(\rho) - O\left(\frac{1}{L}\right)$. Fortunately, we provided a characterization of $R_{\mathrm{RLC}}(\mathcal{P}_n)$:

[1] Although, given that I hope to obtain employment in this field in the future, it is fortuitous that so many problems remain open.

if $\mathcal{P}_n$ is defined by forbidding types in $\mathcal{T}_n$, we know that

$$R_{\mathrm{RLC}}(\mathcal{P}_n) = \min_{\tau \in \mathcal{T}_n} \max_{\tau' \in \mathcal{I}_\tau} \left\{ 1 - \frac{H_q(\tau')}{\mathrm{rank}(\tau')} \right\} \pm o(1) \, .$$

Thus, recalling Example 3.2.8, it will suffice to prove the following.

Conjecture 9.1.1. *There exists a constant $C > 0$ such that the following holds. Let $L \geq 1$ be an integer. Let $\tau \sim \mathbb{F}_q^{L+1}$ be such that for some $\tilde{\tau} \sim \mathbb{F}_q$ and all $i \in [L+1]$,*

$$\mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{z}) \sim (\tau, \tilde{\tau})} (\boldsymbol{u}_i \neq \boldsymbol{z}) \leq \rho \, . \tag{9.1}$$

Moreover, we assume that for all $i \neq j \in [L+1]$,

$$\mathop{\mathbb{P}}_{\boldsymbol{u} \sim \tau} (\boldsymbol{u}_i \neq \boldsymbol{u}_j) > 0 \, .$$

Then there exists $\tau' \in \mathcal{I}_\tau$ for which

$$1 - \frac{H_q(\tau')}{\mathrm{rank}(\tau')} \geq 1 - h_q(\rho) - \frac{C}{L} \, ,$$

i.e.,

$$H_q(\tau') \leq \mathrm{rank}(\tau') \left(h_q(\rho) + \frac{C}{L} \right) \, .$$

Similar conjectures can be made for list-recoverability, etc. Also, note that the conjecture is stated for distributions $\tau \sim \mathbb{F}_q^{L+1}$: while it suffices to prove the conjecture for types $\tau \in \mathcal{D}_{n,L+1}$ (i.e., with the additional assumption that each $\tau(u) \in \{0, 1/n, \ldots, (n-1)/n, 1\}$ for some $n \in \mathbb{N}$), as n is a growing parameter and the desired bound is independent of n, this distinction is inconsequential.

9.1.1 Rephrasing Conjecture With Fourier Analysis

We now highlight an unexpected connection to Fourier analysis.[2] Specifically, we state a conjecture that at first blush appears to have nothing to do with random linear codes or list-decoding, but nonetheless will turn out to be equivalent to Conjecture 9.1.1. (To be pedantic, we will only obtain an equivalence for the average-radius variant of Conjecture 9.1.1.) For simplicity, we restrict attention to $q = 2$. Throughout, we take expectation norms in real space and counting norms in Fourier space, and entropy is always base 2.

Let $f : \mathbb{F}_2^n \to \mathbb{R}$ have $\|f\|_1 = \mathbb{E}_{\boldsymbol{x} \sim \mathbb{F}_2^n} [|f(x)|] = 1$. We think of $|f|$ (i.e., the function $x \mapsto |f(x)|$) as the probability density function of a distribution with respect to the uniform distribution. That is, we write $\boldsymbol{x} \sim |f|$ to denote a random variable in $\mathbb{F}_2^n$ with $\mathbb{P}(\boldsymbol{x} = x) = |f(x)| \cdot 2^{-n}$.

[2]For a refresher on the notation we use, see Section 4.4.1. We note that we will only require the definitions specialized to $q = 2$.

Definition 9.1.2 (Smooth Function)**.** Let $f : \mathbb{F}_2^n \to \mathbb{R}$ and let $\beta \in [0,1]$. We say that f is *β-smooth* if $\|f\|_1 = 1$ and for every $0 \le m \le n$ and every linear surjection $T : \mathbb{F}_2^n \to \mathbb{F}_2^m$,

$$H(T\boldsymbol{x}) > \beta \cdot m \,,$$

where $\boldsymbol{x} \sim |f|$.

Definition 9.1.3 (Fourier-Concentrated Function)**.** Let $f : \mathbb{F}_2^n \to \mathbb{R}$, $\alpha \in (0,1)$, and $m \in \mathbb{N}$.

- If
$$|\{\xi \in \mathbb{F}_2^n : \hat{f}(\xi) > \alpha\}| \le m \,,$$
we say that f is (α, m)*-Fourier-concentrated*.
- If for every $M \subseteq \mathbb{F}_2^n$ with $|M| = m$,
$$\frac{1}{m} \sum_{\xi \in M} \hat{f}(\xi) < \alpha \,,$$
we say that f is (α, m)*-average Fourier-concentrated*.

We now demonstrate that average-radius list-decodability of random linear codes can be formulated in the above language. This can be done quite efficiently with our types formulation.

Proposition 9.1.4. *Let $\rho \in (0, \frac{1}{2})$, $L \in \mathbb{N}$ and $R \in (0,1)$. The following statements are equivalent:*

1. *A random linear code $\boldsymbol{\mathcal{C}} \le \mathbb{F}_2^n$ of rate R is (ρ, L)-average-radius list-decodable with high probability.*
2. *For every $d \le L+1$, every $(1-R)$-smooth function $f : \mathbb{F}_2^d \to \mathbb{R}$ is $(1-2\rho, L+1)$-average Fourier-concentrated.*

Proof. We only prove that the second statement implies the first; this is the only direction that is important if our end-goal is to prove list-decodability of random linear codes, and moreover we feel that it is reasonably clear how to reverse the subsequent argument.

We will prove the contrapositive: namely, the negation of Statement 1 implies the negation of Statement 2. Let ρ, L, R be as above. For Statement 1 to be false, there must be a pair of types $(\tau, \tilde{\tau})$ with $\tau \in \mathcal{D}_{n,L+1}$ and $\tilde{\tau} \in \mathcal{D}_{n,1}$ such that:

$$\frac{1}{L+1} \sum_{i=1}^{L+1} \mathop{\mathbb{P}}_{(\boldsymbol{u},\boldsymbol{z}) \sim (\tau, \tilde{\tau})} (\boldsymbol{u}_i \ne \boldsymbol{z}) \le \rho \,, \tag{9.2}$$

$$\mathop{\mathbb{P}}_{\boldsymbol{u} \sim \tau} (\boldsymbol{u}_i \ne \boldsymbol{u}_j) > 0 \;\; \forall i \ne j \,, \tag{9.3}$$

$$\max_{\tau' \in \mathcal{I}_\tau} \left\{ 1 - \frac{H(\tau')}{\mathrm{rank}(\tau')} \right\} < R \,. \tag{9.4}$$

Furthermore, we may assume that either $(\tau, \tilde{\tau})(u, 0) = 0$ or $(\tau, \tilde{\tau})(u, 1) = 0$ for every $u \in \mathbb{F}_2^{L+1}$, where we have defined $(\tau, \tilde{\tau})(u, z) = \mathbb{P}_{(\boldsymbol{u},\boldsymbol{z}) \sim (\tau, \tilde{\tau})} (\boldsymbol{u} = u, \boldsymbol{z} = z)$. Indeed, suppose

$(\tau, \tilde{\tau})$ satisfies (9.2) and (9.4), and that $(\tau, \tilde{\tau})(u, 0)$ and $(\tau, \tilde{\tau})(u, 1)$ are both positive for some $u \in \mathbb{F}_2^{L+1}$. Consider $(\tau_0, \tilde{\tau}_0)$ and $(\tau_1, \tilde{\tau}_1)$ which are both identical to $(\tau, \tilde{\tau})$ except we have $(\tau_0, \tilde{\tau}_0)(u, 0) = (\tau, \tilde{\tau})(u, 0) + (\tau, \tilde{\tau})(u, 1)$ and $(\tau_0, \tilde{\tau}_0)(u, 1) = 0$, and $(\tau_1, \tilde{\tau}_1)$ is defined similarly. Note that both marginals τ_0 and τ_1 are distributed identically to τ, so (9.4) still holds. Moreover, at least one of $(\tau_0, \tilde{\tau}_0)$ and $(\tau_1, \tilde{\tau}_1)$ satisfies (9.2) as well, since it's impossible that perturbing τ in two opposite directions decreases the left-hand side of (9.2).

Now, let $U = \operatorname{span}(\operatorname{supp}(\tau))$ and denote $d = \dim(U)$ (so $d = \dim(U) = \operatorname{rank}(\tau)$). Let $B : \mathbb{F}_2^d \to U$ be a linear isomorphism. Define the function $f : \mathbb{F}_2^d \to \mathbb{R}$ by

$$f(x) := 2^d \cdot ((\tau, \tilde{\tau})(Bx, 0) - (\tau, \tilde{\tau})(Bx, 1)) .$$

Due to our assumption that either $(\tau, \tilde{\tau})(Bx, 0)$ or $(\tau, \tilde{\tau})(Bx, 1)$ is 0, we find

$$|f(x)| = 2^d \cdot \mathop{\mathbb{P}}_{\boldsymbol{u} \sim \tau} (\boldsymbol{u} = Bx) \ .$$

We claim that f is $(1 - R)$-smooth. Indeed, let $m \leq d$ and let $T : \mathbb{F}_2^d \to \mathbb{F}_2^m$ be a linear surjection. If $\boldsymbol{x} \sim |f|$ and $\boldsymbol{u} \sim \tau$, we know $H(T\boldsymbol{x}) = H(TB^{-1}\boldsymbol{u})$. Let $\tau' \in \mathcal{I}_\tau$ denote the implied type of τ corresponding to the distribution of $TB^{-1}\boldsymbol{u}$. By Eq. (9.4), it follows that $1 - \frac{H(\tau')}{m} < R$, which rearranges to $H(\tau') > (1 - R)m$. That is, $H(T\boldsymbol{x}) > (1 - R)m$, demonstrating that f is indeed $(1 - R)$-smooth.

Thus, to establish the negation of Statement 2, it suffices to prove that f is not $(1 - 2\rho, L + 1)$-average Fourier-concentrated. Let $(\boldsymbol{u}, \boldsymbol{z}) \sim (\tau, \tilde{\tau})$, $\boldsymbol{x} \sim |f|$ and $\boldsymbol{y} \sim \mathbb{F}_2^d$. Let $\xi_1, \ldots, \xi_{L+1}$ denote the rows of B. For each $i \in [L + 1]$, we have

$$\begin{aligned}\mathbb{P}(\boldsymbol{u}_i \neq \boldsymbol{z}) &= \frac{1 - \mathbb{E}\left[(-1)^{\boldsymbol{u}_i + \boldsymbol{z}}\right]}{2} = \frac{1 - \mathbb{E}\left[(-1)^{\langle \xi_i, \boldsymbol{x} \rangle} \cdot \operatorname{sign}(f(\boldsymbol{x}))\right]}{2} \\ &= \frac{1 - \mathbb{E}\left[(-1)^{\langle \xi_i, \boldsymbol{y} \rangle} \cdot f(\boldsymbol{y})\right]}{2} = \frac{1 - \hat{f}(\xi_i)}{2} .\end{aligned}$$

Let $M = \{\xi_1, \ldots, \xi_{L+1}\}$ and note that Condition 9.3 guarantees $|M| = L+1$: indeed, (9.3) establishes that $U = \operatorname{span}(\operatorname{supp}(\tau))$ has the property that for every $i \neq j \in [L + 1]$, there exists a $u \in U$ for which $u_i \neq u_j$. Thus, applying (9.2), we find

$$\frac{1}{L+1} \sum_{i=1}^{L+1} \hat{f}(\xi_i) = \frac{1}{L+1} \sum_{i=1}^{L+1} \left(1 - 2 \mathop{\mathbb{P}}_{(\boldsymbol{u}, \boldsymbol{z}) \sim (\tau, \tilde{\tau})} (\boldsymbol{u}_i \neq \boldsymbol{z})\right) \geq 1 - 2\rho ,$$

establishing the failure of f to be $(1 - 2\rho, L + 1)$-average Fourier-concentrated. □

Remark 9.1.5. This equivalence, or something very similar, should also hold between absolute-radius list-decoding and Fourier-concentrated functions. Unfortunately, it is a bit tricky to deal with the situation where $(\tau, \tau')(u, 0)$ and $(\tau, \tau')(u, 1)$ are both positive when dealing with absolute-radius list-decoding.

Alas, despite coming up with this rephrasing of Conjecture 9.1.1, resolving this conjecture has proved to be quite challenging. In fact, somewhat embarrassingly this thresholds framework has turned out to be more useful for proving *lower bounds* on the list size required by random linear codes.[3] Specifically, subsequent work of mine [Gur+20] has managed to show the following:

- A random linear code of rate $1 - \log_q(\ell) - \varepsilon$ requires lists of size $L \geq \ell^{\Omega(1/\varepsilon)}$ for zero-error list-recovery with input lists of size ℓ.
- A random linear code of rate $1 - h_q(\rho) - \varepsilon$ requires lists of size $L \geq \lfloor h_q(\rho)/\varepsilon + 0.99 \rfloor$ for list-decoding from error fraction ρ.

9.2 An Additive Combinatorics Conjecture

In light of our difficulties in obtaining new upper bounds on the list size for a random linear code via the types formalism, we have also explored other potential avenues for resolving this question. In this section, we propose an adaptation of our argument from Chapter 6 (which we recall builds on [Gur+02; LW18]) to field size $q > 2$. Recall that the argument proceeds as follows: a potential function is defined, and then one studies how it varies as random basis elements from $\mathbb{F}_2^n$ are added to the code. Using pairwise independence, one can demonstrate that it roughly squares. The analysis is greatly aided by the fact that lines in $\mathbb{F}_2^n$ only contain 2 points, which means that pairwise independence is sufficient to understand the behavior of the potential function.

To sketch our proposed approach, we focus on the case $q = 3$. Hopefully, a resolution of the $q = 3$ case should readily generalize to larger q. Furthemore this approach, if successful, should easily adapt to the setting of list-recovery. In the sequel, for a subset $X \subseteq \mathbb{F}_3^n$, 1_X denotes the indicator function of the set X, i.e.,

$$1_X(x) = \begin{cases} 1 & \text{if } x \in X \\ 0 & \text{otherwise.} \end{cases}$$

A conjecture. We first provide an additive combinatorics conjecture. Later, we show why it is sufficient to establish that random linear codes of rate $1 - h_3(\rho) - \varepsilon$ are $(\rho, O(1/\varepsilon))$-list-decodable with high probability.

Conjecture 9.2.1. *Let $\varepsilon > 0$. There exists a $K = o(3^{\varepsilon n})$ such that the following holds. Let $X_0, X_1, X_2 \subseteq \mathbb{F}_3^n$ be subsets satisfying $|X_j| \leq \beta_j 3^n$ for each j, where each $\beta_j \leq 3^{-\varepsilon n/2}$. Then, if $\boldsymbol{b}, \boldsymbol{x} \sim \mathbb{F}_3^n$ are uniformly random and independent vectors, for all sufficiently large n, we have*

$$\mathop{\mathbb{P}}_{\boldsymbol{b}} \left(\mathop{\mathbb{E}}_{\boldsymbol{x}} \left[1_{X_0}(\boldsymbol{x}) 1_{X_1}(\boldsymbol{x} + \boldsymbol{b}) 1_{X_2}(\boldsymbol{x} + 2\boldsymbol{b}) \right] \geq K \beta_0 \beta_1 \beta_2 \right) \leq n^{-3} \,. \tag{9.5}$$

[3] In hindsight, this is perhaps not too surprising. Recall that, for a type τ, it was more difficult to show that the threshold τ-freeness was at most $R_\tau^* = \max_{\tau' \in \mathcal{I}_\tau} R_{\mathrm{RLC}}^{\mathbb{E}}(\tau')$. Hence, it is understandable why this more difficult direction of the theorem would yield improved results.

Essentially, Conjecture 9.2.1 states that the fraction of directions b for which there exist significantly more than a $\beta_0\beta_1\beta_2$ fraction of $x \in \mathbb{F}_3^n$ for which the line in direction b through x is contained in $X_0 \times X_1 \times X_2$ is negligible.[4] And we really mean *significantly more*: suppose $\beta_0 = \beta_1 = \beta_2 = 3^{-\varepsilon n/2}$. Then, for fixed $b \in \mathbb{F}_3^n$, having

$$\mathop{\mathbb{E}}_{\boldsymbol{x}} [1_{X_0}(\boldsymbol{x}) 1_{X_1}(\boldsymbol{x}+b) 1_{X_2}(\boldsymbol{x}+2b)] = 3^{-\varepsilon n/2} = \omega(K\beta_0\beta_1\beta_2)$$

means that whenever $\boldsymbol{x}$ happens to lie in X_0 (which occurs with probability $3^{-\varepsilon n/2}$), the next two points on the line $\{\boldsymbol{x} + jb : j \in \mathbb{F}_3\}$ always lie in X_1 and X_2, respectively. Asking for the fraction of b for which this holds to be n^{-3} appears to be a very modest request given the assumption that the X_j's have exponentially small density.

For concreteness, we now describe two cases. Suppose first $X_0 = X_1 = X_2 = V$, where V is a subspace with $|V| = \beta \cdot 3^n$. We have that $\mathbb{E}_{\boldsymbol{x}} [1_V(\boldsymbol{x}) 1_V(\boldsymbol{x}+b) 1_V(\boldsymbol{x}+2b)] = \beta$ if $b \in V$ and $\mathbb{E}_{\boldsymbol{x}} [1_V(\boldsymbol{x}) 1_V(\boldsymbol{x}+b) 1_V(\boldsymbol{x}+2b)] = 0$ if $b \notin V$. Indeed, $\boldsymbol{x}$ lies in V with probability β, and then for $\boldsymbol{x} + b$ to also lie in V we need $b \in V$. Thus,

$$\mathop{\mathbb{P}}_{\boldsymbol{b}} \left(\mathop{\mathbb{E}}_{\boldsymbol{x}} [1_{X_0}(\boldsymbol{x}) 1_{X_1}(\boldsymbol{x}+\boldsymbol{b}) 1_{X_2}(\boldsymbol{x}+2\boldsymbol{b})] > 0 \right) = \mathop{\mathbb{P}}_{\boldsymbol{b}} (\boldsymbol{b} \in V) = \beta \, .$$

Assuming $\beta \leq 3^{-\varepsilon n/2}$, we see that Conjecture 9.2.1 holds in this case for any $K > 0$, even with exponentially small probability of failure. At the other extreme, suppose that X_0 has very little linear structure in the Fourier-analytic sense that $\widehat{1_{X_0}}(\xi) \leq \beta_0$ for all $\xi \in \mathbb{F}_3^n$.[5] A standard argument (see, e.g., [TV06, Proposition 10.11]) shows that

$$\mathop{\mathbb{E}}_{\boldsymbol{x},\boldsymbol{b}} [1_{X_0}(\boldsymbol{x}) 1_{X_1}(\boldsymbol{x}+\boldsymbol{b}) 1_{X_2}(\boldsymbol{x}+2\boldsymbol{b})] \leq \left(\max_{\xi \in \mathbb{F}_3^n} \widehat{1_{X_0}}(\xi) \right) \cdot \beta_1\beta_2 = \beta_0\beta_1\beta_2 \, ,$$

and so Markov's inequality implies that

$$\mathop{\mathbb{P}}_{\boldsymbol{b}} \left(\mathop{\mathbb{E}}_{\boldsymbol{x}} [1_{X_0}(\boldsymbol{x}) 1_{X_1}(\boldsymbol{x}+\boldsymbol{b}) 1_{X_2}(\boldsymbol{x}+2\boldsymbol{b})] \geq K\beta_0\beta_1\beta_2 \right) \leq \frac{1}{K} \, .$$

So taking $K = n^3$ is sufficient for Conjecture 9.2.1, or we could choose it to be $3^{\varepsilon n/2}$, say, if we want exponentially small failure probability. Our hope is to use techniques from additive combinatorics to decompose our sets X_0, X_1, X_2 into structured and pseudo-random components, which we can then analyze separately.

Establishing the list-decodability of random linear codes. In the remainder of this section we show that this conjecture establishes $(\rho, O(1/\varepsilon))$-list-decodability of rate $1 -$

[4] The choice of n^{-3} is mostly for convenience – see the proof of Lemma 9.2.3. In fact, I would conjecture that we could have failure probability of the form $3^{-\Omega(\varepsilon n)}$.

[5] This holds with high probability for random sets; see, e.g., Theorem 1.13 of [Hay05].

$h_3(\rho) - \varepsilon$ random linear codes. For a code $\mathcal{C}$ and $x \in \mathbb{F}_3^n$ define $L_{\mathcal{C}}(x) = |B(x, \rho n) \cap \mathcal{C}|$,[6] and for $\ell \in \mathbb{N}$, define

$$f_\ell(\mathcal{C}) = \frac{|\{x \in \mathbb{F}_3^n : L_{\mathcal{C}}(x) \geq \ell\}|}{3^n} ,$$

i.e., the fraction of centers which have at least ℓ codewords within Hamming distance ρ.

Consider the effect of adding uniformly random basis elements $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \in \mathbb{F}_3^n$ to the code step-by-step, and define $\boldsymbol{\mathcal{C}}_i = \operatorname{span}\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_i\}$. Our goal is to show that $f_L(\boldsymbol{\mathcal{C}}_k) < 3^{-n}$ whp. To establish this, we make use of the following definition.

Definition 9.2.2. Let $\alpha, \beta \in (0, 1)$ and $L \geq 1$ an integer. We call a linear code $\mathcal{C} \leq \mathbb{F}_3^n$ (α, β, L)-*nice* if for all integers $1 \leq \ell \leq L$, $f_\ell(\mathcal{C}) \leq \beta \cdot \alpha^\ell$.

The following lemma demonstrates that in passing from $\boldsymbol{\mathcal{C}}_i$ to $\boldsymbol{\mathcal{C}}_{i+1}$, if $\boldsymbol{\mathcal{C}}_i$ was (α, β, L)-nice then $\boldsymbol{\mathcal{C}}_{i+1}$ is with high probability roughly $(\alpha, 3\beta, L)$-nice.

Lemma 9.2.3. *Let $\varepsilon > 0$ and assume Conjecture 9.2.1. Let $\beta \leq 3^{-\varepsilon n/2}$ and $\alpha \in (0, 1)$. Suppose that $\mathcal{C} \leq \mathbb{F}_3^n$ is (α, β, L)-nice, and let $\boldsymbol{\mathcal{C}}' = \operatorname{span}\{\mathcal{C}, \boldsymbol{b}\}$ where $\boldsymbol{b} \in \mathbb{F}_3^n$ is uniformly random. Then with probability at least $1 - 2n^{-2}$, $\boldsymbol{\mathcal{C}}'$ is $(\alpha, 3\beta(1 + o(1)), L)$-nice.*

Proof. In what follows, we regularly state inequalities which hold assuming n is large enough compared to the other parameters. Let $1 \leq \ell \leq L$ be an integer; we wish to establish that $f_\ell(\boldsymbol{\mathcal{C}}') \leq 3\beta(1 + o(1))\alpha^\ell$ with probability at least $1 - 2n^{-2}$. Let $X_j = \{x \in \mathbb{F}_3^n : L_{\mathcal{C}}(x) \geq j\}$ denote the set of points with at least j nearby codewords. Our assumption gives $|X_j| = f_j(\mathcal{C}) \cdot 3^n \leq \beta \cdot \alpha^j \cdot 3^n$.

We call $b \in \mathbb{F}_3^n$ *good* if, for each $1 \leq j_0, j_1, j_2 \leq L$ with $j_0 + j_1 + j_2 \leq L$, we have

$$\mathop{\mathbb{E}}_{\boldsymbol{x}} \left[1_{X_{j_0}}(\boldsymbol{x}) 1_{X_{j_1}}(\boldsymbol{x} + b) 1_{X_{j_2}}(\boldsymbol{x} + 2b)\right] \leq K\beta^3\alpha^{j_0+j_1+j_2} , \tag{9.6}$$

where $K = o(3^{\varepsilon n})$ is as in Conjecture 9.2.1. By Conjecture 9.2.1 and a union bound over the choice of (j_0, j_1, j_2), over the randomness of $\boldsymbol{b}$ this fails to hold with probability at most $L^3 n^{-3} \leq n^{-2}$. We now condition on the event that $\boldsymbol{b}$ is good.

For $x \in \mathbb{F}_3^n$, consider the equation

$$L_{\boldsymbol{\mathcal{C}}'}(x) = L_{\mathcal{C}}(x) + L_{\mathcal{C}}(x + \boldsymbol{b}) + L_{\mathcal{C}}(x + 2\boldsymbol{b}) . \tag{9.7}$$

For an integer j, we now consider all the ways that we could have (9.7) at least ℓ.

- Exactly 1 of the terms on the right-hand side of (9.7) is nonzero, and has value at least ℓ. Note that, for any $b \in \mathbb{F}_3^n$, the fraction of points for which this case occurs is

$$\begin{aligned}\mathop{\mathbb{E}}_{\boldsymbol{x}} [1_{X_\ell}(\boldsymbol{x})\ 1_{X_0}(\boldsymbol{x} + b) 1_{X_0}(\boldsymbol{x} + 2b)] &+ \mathop{\mathbb{E}}_{\boldsymbol{x}} [1_{X_0}(\boldsymbol{x}) 1_{X_\ell}(\boldsymbol{x} + b) 1_{X_0}(\boldsymbol{x} + 2b)] \\ &+ \mathop{\mathbb{E}}_{\boldsymbol{x}} [1_{X_0}(\boldsymbol{x}) 1_{X_0}(\boldsymbol{x} + b) 1_{X_\ell}(\boldsymbol{x} + 2b)] \\ &= 3 \mathop{\mathbb{E}}_{\boldsymbol{x}} [1_{X_\ell}(\boldsymbol{x})] \leq 3\beta\alpha^\ell .\end{aligned}$$

[6] One could imagine using the "smoothed-out" list size as defined in Chapter 6 to establish average-radius list-decodability; however, for simplicity, we will stick to this more concrete definition.

(Note that $1_{X_0} = 1_{\mathbb{F}_3^n}$, the constant 1 function.) Thus, with probability 1, the contribution of this term is at most $3\beta\alpha^\ell$.

- Exactly 2 of the terms on the right-hand side of (9.7) are nonzero, and for some $0 < j < \ell$ one term has value at least j and another term has value at least $\ell - j$. If $\boldsymbol{b}$ were uniformly random, the expected fraction of points for which this case occurs is at most $3^2\ell\beta^2\alpha^\ell \le 9L\beta^2\alpha^\ell$; conditioning on $\boldsymbol{b}$ being good, this fraction can increase by at most $(1 - n^{-2})^{-1} \le (1 + 2n^{-2})$. Hence, by Markov's inequality, with probability at least $1 - \beta^{0.5}$ we have that this fraction is at most $9L\beta^{1.5}\alpha^\ell(1 + 2n^{-2}) \le \beta^{1.4}\alpha^\ell$.
- All 3 of the terms on the right-hand side of (9.7) are nonzero, and for some $0 < j_1, j_2 < \ell - 1$ with $j_1 + j_2 < \ell$, one term takes value at least j_1, another takes value at least j_2, and the last takes value at least $\ell - j_1 - j_2$. As we have conditioned on $\boldsymbol{b}$ being good, the fraction of points for which this occurs is at most

$$K\beta^3\alpha^\ell = o(\beta)\alpha^\ell \,.$$

  Hence, the contribution from this term is at most $o(\beta)\alpha^\ell$.

Hence, summing these fractions, we find that except with probability at most $n^{-2} + \beta^{0.5} \le 2n^{-2}$,

$$f_\ell(\boldsymbol{\mathcal{C}}') \le \left(3\beta + \beta^{1.4} + o(\beta)\right)\alpha^\ell \le 3\beta(1 + o(1))\alpha^\ell \,. \qquad \square$$

Now, for the code $\mathcal{C}_0 = \{0\}$, $L_{\mathcal{C}_0}(x) = 1$ for each $x \in B(0, \rho)$ and $= 0$ if $x \notin B(0, \rho)$. Hence, using $|B(0, \rho)| \le 3^{h_3(\rho)n}$, $\{0\}$ satisfies the hypotheses of Lemma 9.2.3 with $\beta = 3^{-(1-h_3(\rho)-\varepsilon/3)n}$ and $\alpha = 3^{-\varepsilon n/3}$. By a union bound, with probability at least $1 - 2n^{-1}$, we have that $\boldsymbol{\mathcal{C}}_k$ for $k = (1 - h_3(\rho) - \varepsilon)n$ is (α, β')-nice for

$$\beta' = 3^k(1 + o(1))^k\beta = 3^{(1-h_3(\rho)-\varepsilon)n-(1-h_3(\rho)-\varepsilon/3)n}(1 + o(1))^k \le 3^{-\varepsilon n/2} \,.$$

Thus, with probability at least $1 - 2n^{-1}$,

$$f_L(\boldsymbol{\mathcal{C}}_k) \le \alpha(\beta')^L = 3^{-\varepsilon n/3} \cdot 3^{-\varepsilon nL/2} < 3^{-n} \,,$$

assuming $L = c_0/\varepsilon$ for a large enough constant c_0.

9.3 Explicit LDPC Codes

As we have now demonstrated that random LDPC codes achieve list-decoding capacity, a tantalizing open problem is to now explicitly construct capacity-achieving list-decodable LDPC codes. Most known constructions of capacity-achieving list-decodable codes are inherently algebraic, and it appears unlikely that these techniques could yield genuinely linear-time decoding algorithms. Furthermore, it appears reasonable to suspect that a successful construction of a list-decodable LDPC code would inherently have constant list and alphabet size. We remark that our work on tensor codes does provide a construction of a $O(n^{1+o(1)})$-list-decoding algorithm, and while the alphabet size is constant the list size is a slowly growing function of n.

In particular, I believe that LDPC codes constructed from expander graphs are a natural candidate for explicit list-decodable codes that (a) have constant alphabet and list size, and (b) admit a linear-time decoding algorithm. As a first step in this direction, Hemenway and Wootters [HW15] demonstrated rate $1 - \varepsilon$ expander codes can be list-recovered from erasures with constant list sizes from a constant fraction ρ of erasures. In this model, one is given a tuple of lists $(S_1, \ldots, S_n)$ such that except for αn choices of $i \in [n]$, $|S_i| \leq \ell$, and the goal is to output all $c \in \mathcal{C}$ satisfying $c_i \in S_i$ for all $i \in [n]$. Plugging this into standard "distance amplification" machinery [AEL95; AL96; GI02; Gop+18] allows them to achieve the optimal tradeoff between any rate and erasure-radius, still with linear-time list-recovery algorithms. When it comes to binary codes, Ron-Zewi, Wootters and Zémor [RZWZ20] have recently demonstrated how to construct erasure list-decodable expander codes whose erasure-decoding radius exceeds the designed distance of the code.

As these "expander codes" are different than the LDPC codes of Gallager's ensemble we defined in Chapter 4, we now provide the definition. Given a graph $G = (V, E)$ of degree d and a linear inner code $\mathcal{C}_0 \leq \mathbb{F}_q^d$, define

$$\mathcal{C}(G; \mathcal{C}_0) := \{x \in \mathbb{F}_q^E : x|_{\delta(v)} \in \mathcal{C}_0 \ \forall v \in V\} ,$$

where $\delta(v)$ is the set of edges incident on vertex v. Informally, this is the code obtained by placing symbols on the edges of the graph G subject to the constraint that the edges incident to each vertex lie in the code $\mathcal{C}_0$.[7]

The algorithms of [HW15] and [RZWZ20] could naturally be adapted to work in the more challenging model of errors; however, the authors were only able to analyze it in the model of erasures. We hope that we can use insights gleaned from the analysis of random LDPC codes to help us understand the performance of these algorithms in the more challenging model of errors.

Next, we indicate that expander graphs can be viewed as a "sparsified" tensor code. In this context, one again slightly tweaks the definition of an expander code:[8] one takes a (c, d)-biregular graph $H = (V \cup U, E)$ and two inner codes $\mathcal{C}_1 \leq \mathbb{F}_q^c$ and $\mathcal{C}_2 \leq \mathbb{F}_2^d$, and defines

$$\mathcal{C}(H; \mathcal{C}_1, \mathcal{C}_2) := \{x \in \mathbb{F}_q^E : x|_{\delta(v)} \in \mathcal{C}_1 \ \forall v \in V \text{ and } x|_{\delta(u)} \in \mathcal{C}_2 \ \forall u \in U\} .$$

For the case of $H = K_{n_1,n_2}$, observe that $\mathcal{C}(H; \mathcal{C}_1, \mathcal{C}_2) = \mathcal{C}_1 \otimes \mathcal{C}_2$. Therefore, my work on the list-decodability of tensor codes can be viewed as a stepping stone towards understanding the list-decodability of expander codes.

Finally, another natural open problem is to develop an efficient (linear-time?) list-decoding/recovery algorithm for the LDPC codes constructed in Chapter 4. This would provide a Monte Carlo construction of a capacity-achieving code over any alphabet which can be efficiently decoded with constant list sizes.

[7] Of course, one should fix an ordering on the edges to avoid any ambiguities.

[8] Apologies for the surfeit of definitions: it seems that expander graphs are so magical that coding theorists have employed them in many different ways...

9.4 Two-Source Rank Condensers

Lastly, having constructed optimal dimension expanders in Chapter 8, we turn our attention to the construction of other algebraic pseudorandom objects. As indicated earlier, dimension expanders can be viewed as the algebraic analog of (unbalanced) bipartite expander graphs, which are themselves equivalent to seeded randomness extractors. For future work, we propose a construction of a two-source rank condenser, which is the algebraic analog of two-source extractors/condensers. Fixing parameters a bit arbitrarily, call a linear function $\varphi : \mathbb{F}^n \otimes \mathbb{F}^n \to \mathbb{F}^m$ a *two-source rank condenser* if for all $V, W \leq \mathbb{F}^n$ of dimension at least $\sqrt{n}$, $\dim(\varphi(V \otimes W)) \geq n/2$. The probabilistic method shows that it suffices to take $m = O(n)$; however, the best known construction requires $m = n^{3/2}$.[9]

We describe a construction of a linear function φ that we believe could constitute a two-source rank condenser. Recall the definition $\mathcal{F} = \left\{ f(X) = \sum_{i=0}^{k-1} f_i X^{q^i} : f_i \in H_{i+1} \right\}$, where $H_1, \ldots, H_k$ form a subspace design. Assume $\dim(\mathcal{F}) = n$. Then, define $\varphi : \mathcal{F} \otimes \mathbb{F}_{q^n} \to \mathbb{F}_{q^n}$ to be the linear map induced by the bilinear mapping $(f, \alpha) \mapsto f(\alpha)$. For context, if $\Gamma_1, \ldots, \Gamma_d : \mathbb{F}^n \to \mathbb{F}^n$ give an (η, β)-dimension expander, then if one defines $\psi : \mathbb{F}^n \otimes \mathbb{F}^d \to \mathbb{F}^n$ as the linear map induced by the bilinear mapping $(v, u) \mapsto \sum_{j=1}^{d} u_j \Gamma_j(v)$, one can show that for any subspace $U \leq \mathbb{F}^n$ of dimension at most ηn, $\dim(\psi(U \otimes \mathbb{F}^d)) \geq \beta \dim(U)$. Instantiating this with our dimension expander from Theorem 8.5.2, we thus see that the function $\psi : \mathcal{F} \otimes \mathbb{F}_{q^d} \to \mathbb{F}_{q^n}$ induced by $(f, \alpha) \mapsto f(\alpha)$ has the property that for any $U \leq \mathcal{F}$ of dimension at most $\approx \frac{n}{d}$, $\dim(\psi(U \otimes \mathbb{F}_{q^d})) \approx d \cdot \dim(U)$.

For this reason, we believe that the map φ defined above is a very viable candidate for a two-source rank condenser. Unfortunately, the analysis of the dimension expander from Theorem 8.5.2 relied crucially on the fact that $\mathbb{F}_{q^d}$ forms a subfield, so showing that $\varphi(U \otimes V)$ has large dimension when $V \leq \mathbb{F}_{q^n}$ is an arbitrary $\mathbb{F}_q$-linear subspace will require new ideas. Nonetheless, we believe this is a fruitful avenue to pursue.

9.5 Miscellaneous Open Problems

Lastly, we collect several other questions that were raised earlier.

- The machinery in Chapter 3 applies to any local property, although we were mostly interested in list-decodability and related notions. Are there other natural properties that are local?
- Can we more precisely define what it means for a local property family $\mathcal{P} = (\mathcal{P}_{n_i})$ to be "uniform"? Here is a natural definition: there is a fixed a polyhedron P contained in the probability simplex of $\mathbb{R}^{q^\ell}$, and for each n_i, $\mathcal{P}_{n_i}$ is defined by forbidding all types $\tau \in \mathcal{D}_{n_i,\ell}$ for which the vector $(\tau(u))_{u \in \mathbb{F}_q^\ell}$ is contained in P. There

[9] This construction actually provides the stronger guarantee that $\dim(\varphi(V \otimes W))geqn$. In fact, the construction is obtained from a Gabidulin code, or any rank metric code achieving the Singleton bound.

are then two natural questions to ask about such "uniform" types.

1. Does the limit $\lim_{i\to\infty}(R_{\mathrm{RLC}}(\mathcal{P}_{n_i}))$ exist?
2. Recall Remark 4.2.5: in general, we cannot prove a converse to Theorem 4.2.3: one can define a property family $(\mathcal{P}_n)$ such that, even if $R_n \geq R_{\mathrm{RLC}}(\mathcal{P}_n) + 0.99$ for all n, the random LDPC code $\boldsymbol{\mathcal{C}}^n_{s\mathrm{LDPC}}(R_n)$ does satisfy the property with probability 1 for all n. However, one can observe that the local property defined in Remark 4.2.5 is not uniform in the polyhedral sense given above. For this reason, we ask the following question: does a converse to Theorem 4.2.3 hold if we restrict attention to these uniform property families?

- For rank metric codes in $\mathbb{F}_{q^m}^n$, do there exist any $\mathbb{F}_{q^m}$-linear codes which are list-decodable beyond half the minimum distance?
- Can we construct a dimension expander with smaller field size? The answer to this question would be yes if we could construct subspace designs with smaller field sizes.

9.6 Final Thoughts

Thank you to everyone who has read this far![10] We hope that we have managed to demonstrate why list-decodable codes are a subject meriting this much study. Not only do they find interesting applications in the real world and the (equally valuable) TCS world, but the mathematical challenges they pose are both stimulating and captivating.

As a final question: may I please have a Ph.D.?

[10]Mom and/or Dad, if you trudged all the way to this point, I'd be very curious to hear what you understood.

Bibliography

[AVZ00] Erik Agrell, Alexander Vardy, and Kenneth Zeger. "Upper bounds for constant-weight codes". In: *IEEE Transactions on Information Theory* 46.7 (2000), pp. 2373–2395.

[ABI86] Noga Alon, László Babai, and Alon Itai. "A fast and simple randomized parallel algorithm for the maximal independent set problem". In: *Journal of algorithms* 7.4 (1986), pp. 567–583.

[ABP18] Noga Alon, Boris Bukh, and Yury Polyanskiy. "List-decodable zero-rate codes". In: *IEEE Transactions on Information Theory* 65.3 (2018), pp. 1657–1667.

[AEL95] Noga Alon, Jeff Edmonds, and Michael Luby. "Linear time erasure codes with nearly optimal recovery". In: *Proceedings of the 36th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE. 1995, pp. 512–519.

[AL96] Noga Alon and Michael Luby. "A linear time erasure-resilient code with nearly optimal recovery". In: *IEEE Transactions on Information Theory* 42.6 (1996), pp. 1732–1736.

[ABL00] Alexei Ashikhmin, Alexander Barg, and Simon Litsyn. "A new upper bound on codes decodable into size-2 lists". In: *Numbers, Information and Complexity*. Springer, 2000, pp. 239–244.

[Bab+91] László Babai, Lance Fortnow, Leonid A Levin, and Mario Szegedy. "Checking computations in polylogarithmic time". In: *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing (STOC)*. 1991, pp. 21–32.

[Bar+04] Boaz Barak, Russell Impagliazzo, Amir Shpilka, and Avi Wigderson. Personal Communication to Dvir-Shpilka [DS11]. 2004.

[BKP18] Nir Bitansky, Yael Tauman Kalai, and Omer Paneth. "Multi-collision resistance: a paradigm for keyless hash functions". In: *Proceedings of the 50th Annual ACM Symposium on Theory of Computing (STOC)*. 2018, pp. 671–684.

[Bli05] Vladimir M Blinovsky. "Code bounds for multiple packings over a nonbinary finite alphabet". In: *Problems of Information Transmission* 41.1 (2005), pp. 23–32.

[Bli86] Volodia M Blinovsky. "Bounds for codes in the case of list decoding of finite volume". In: *Problems of Information Transmission* 22.1 (1986), pp. 7–19.

[Bol01] Béla Bollobás. *Random graphs*. 73. Cambridge university press, 2001.

[BY13] Jean Bourgain and Amir Yehudayoff. "Expansion in $\mathrm{SL}_2(\mathbb{R})$ and monotone expanders". In: *Geometric and Functional Analysis* 23.1 (2013). Preliminary version in the *44th Annual ACM Symposium on Theory of Computing (STOC 2012).*, pp. 1–41. DOI: 10.1007/s00039-012-0200-9.

[CPS99] Jin-Yi Cai, Aduri Pavan, and D Sivakumar. "On the hardness of permanent". In: *Proceedings of the 16th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*. Springer. 1999, pp. 90–99.

[CGV13] Mahdi Cheraghchi, Venkatesan Guruswami, and Ameya Velingker. "Restricted isometry of Fourier matrices and list decodability of random linear codes". In: *SIAM Journal on Computing* 42.5 (2013), pp. 1888–1914.

[CT12] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[CS+04] Imre Csiszár, Paul C Shields, et al. "Information theory and statistics: A tutorial". In: *Foundations and Trends® in Communications and Information Theory* 1.4 (2004), pp. 417–528.

[Del78] Philippe Delsarte. "Bilinear forms over a finite field, with applications to coding theory". In: *Journal of Combinatorial Theory, Series A* 25.3 (1978), pp. 226–241.

[Din14] Yang Ding. "On list-decodability of random rank metric codes and subspace codes". In: *IEEE Transactions on Information Theory* 61.1 (2014), pp. 51–59.

[Dor+19] Dean Doron, Dana Moshkovitz, Justin Oh, and David Zuckerman. *Nearly optimal pseudorandomness from hardness*. Tech. rep. ECCC preprint TR19-099, 2019.

[DL12] Zeev Dvir and Shachar Lovett. "Subspace evasive sets". In: *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC)*. 2012, pp. 351–358.

[DS07] Zeev Dvir and Amir Shpilka. "Locally decodable codes with two queries and polynomial identity testing for depth 3 circuits". In: *SIAM Journal on Computing* 36.5 (2007), pp. 1404–1434.

[DS11] Zeev Dvir and Amir Shpilka. "Towards dimension expanders over finite fields". In: *Combinatorica* 31.3 (2011), pp. 305–320.

[DW10] Zeev Dvir and Avi Wigderson. "Monotone expanders: Constructions and applications". In: *Theory of Computing* 6.1 (2010), pp. 291–308.

[Eli57] Peter Elias. "List decoding for noisy channels". In: (1957).

[Eli91] Peter Elias. "Error-correcting codes for list decoding". In: *IEEE Transactions on Information Theory* 37.1 (1991), pp. 5–12.

[FG15] Michael A Forbes and Venkatesan Guruswami. "Dimension Expanders via Rank Condensers". In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2015)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2015.

[FS12] Michael A Forbes and Amir Shpilka. "On identity testing of tensors, low-rank recovery and compressed sensing". In: *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC)*. ACM. 2012, pp. 163–172.

[FS95] Katalin Friedl and Madhu Sudan. "Some improvements to total degree tests". In: *Proceedings of the 3rd Annual Israel Symposium on the Theory of Computing and Systems (ISTCS)*. IEEE. 1995, pp. 190–198.

[GPT91] EM Gabidulin, AV Paramonov, and OV Tretjakov. "Ideals over a non-commutative ring and their application in cryptology". In: *Workshop on the Theory and Application of of Cryptographic Techniques*. Springer. 1991, pp. 482–489.

[Gab85] Ernst M Gabidulin. "Theory of codes with maximum rank distance". In: *Problemy Peredachi Informatsii* 21.1 (1985), pp. 3–16.

[GR08a] Ariel Gabizon and Ran Raz. "Deterministic extractors for affine sources over large fields". In: *Combinatorica* 28.4 (2008), pp. 415–440.

[GY08] Maximilien Gadouleau and Zhiyuan Yan. "On the decoder error probability of bounded rank-distance decoders for maximum rank-distance codes". In: *IEEE Transactions on Information Theory* 54.7 (2008), pp. 3202–3206.

[Gal62] Robert Gallager. "Low-density parity-check codes". In: *IRE Transactions on Information Theory* 8.1 (1962), pp. 21–28.

[Gil+13] Anna C Gilbert, Hung Q Ngo, Ely Porat, Atri Rudra, and Martin J Strauss. "ℓ_2/ℓ_2-Foreach sparse recovery with low risk". In: *Proceedings of the 40th Annual International Colloquium on Automata, Languages, and Programming (ICALP)*. Springer. 2013, pp. 461–472.

[Gil52] Edgar N Gilbert. "A comparison of signalling alphabets". In: *The Bell System Technical Journal* 31.3 (1952), pp. 504–522.

[Gol11] Oded Goldreich. "A sample of samplers: A computational perspective on sampling". In: *Studies in Complexity and Cryptography. Miscellanea on the Interplay between Randomness and Computation*. Springer, 2011, pp. 302–332.

[GL89] Oded Goldreich and Leonid A Levin. "A hard-core predicate for all one-way functions". In: *Proceedings of the 21st Annual ACM Symposium on Theory of Computing (STOC)*. ACM. 1989, pp. 25–32.

[GRS06] Oded Goldreich, Dana Ron, and Madhu Sudan. "Chinese remaindering with errors". In: *IEEE Transactions on Information Theory* 46.4 (2006), pp. 1330–1338.

[GS06] Oded Goldreich and Madhu Sudan. "Locally testable codes and PCPs of almost-linear length". In: *Journal of the ACM (JACM)* 53.4 (2006), pp. 558–655.

[GGR11] Parikshit Gopalan, Venkatesan Guruswami, and Prasad Raghavendra. "List decoding tensor products and interleaved codes". In: *SIAM Journal on Computing* 40.5 (2011), pp. 1432–1462.

[Gop+18] Sivakanth Gopi, Swastik Kopparty, Rafael Oliveira, Noga Ron-Zewi, and Shubhangi Saraf. "Locally testable and locally correctable codes approaching the Gilbert-Varshamov bound". In: *IEEE Transactions on Information Theory* 64.8 (2018), pp. 5813–5831.

[Gop81] Valerii Denisovich Goppa. "Codes on algebraic curves". In: *Soviet Math. Dokl.* Vol. 24. 1981, pp. 170–172.

[Gur04] Venkatesan Guruswami. *List decoding of error-correcting codes: winning thesis of the 2002 ACM doctoral dissertation competition*. Vol. 3282. Springer Science & Business Media, 2004.

[Gur06] Venkatesan Guruswami. "Iterative decoding of low-density parity check codes (A Survey)". In: *Bulletin of the European Association for Theoretical Computer Science (EATCS)* 90 (2006).

[Gur11] Venkatesan Guruswami. "Linear-algebraic list decoding of folded Reed-Solomon codes". In: *Proceedings of the 26th Annual IEEE Conference on Computational Complexity (CCC)*. IEEE. 2011, pp. 77–85.

[GHK11] Venkatesan Guruswami, Johan Håstad, and Swastik Kopparty. "On the List-Decodability of Random Linear Codes". In: *IEEE Transactions on Information Theory* 2.57 (2011), pp. 718–725.

[GI01] Venkatesan Guruswami and Piotr Indyk. "Expander-based constructions of efficiently decodable codes". In: *Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2001, pp. 658–667.

[GI02] Venkatesan Guruswami and Piotr Indyk. "Near-optimal linear-time codes for unique decoding and new list-decodable codes over smaller alphabets". In: *Proceedings of the 34th Annual ACM Symposium on Theory of Computing (STOC)*. ACM. 2002, pp. 812–821.

[GI03] Venkatesan Guruswami and Piotr Indyk. "Linear time encodable and list decodable codes". In: *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*. ACM. 2003, pp. 126–135.

[GI04] Venkatesan Guruswami and Piotr Indyk. "Efficiently decodable codes meeting Gilbert-Varshamov bound for low rates". In: *Proceedings of the 15th Annual ACM-SIAM Symposium on Discrete Algorithm (SODA)*. SIAM, 2004, pp. 756–757. ISBN: 0-89871-558-X. URL: `http://dl.acm.org/citation.cfm?id=982792`.

[GK16] Venkatesan Guruswami and Swastik Kopparty. “Explicit subspace designs”. In: *Combinatorica* 36.2 (2016), pp. 161–185.

[GN14] Venkatesan Guruswami and Srivatsan Narayanan. “Combinatorial limitations of average-radius list-decoding”. In: *IEEE Transactions on Information Theory* 60.10 (2014), pp. 5827–5842.

[GR18] Venkatesan Guruswami and Nicolas Resch. “On the list-decodability of random linear rank-metric codes”. In: *Proceedings of the 2018 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2018, pp. 1505–1509.

[GRX18] Venkatesan Guruswami, Nicolas Resch, and Chaoping Xing. “Lossless Dimension Expanders via Linearized Polynomials and Subspace Designs”. In: *Proceedings of the 33rd Annual Computational Complexity Conference (CCC)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2018.

[GR08b] Venkatesan Guruswami and Atri Rudra. “Explicit codes achieving list decoding capacity: error-correction with optimal redundancy”. In: *IEEE Transactions on Information Theory* 54.1 (2008), pp. 135–150.

[GR10] Venkatesan Guruswami and Atri Rudra. “The existence of concatenated codes list-decodable up to the Hamming bound”. In: *IEEE Transactions on Information Theory* 56.10 (2010), pp. 5195–5206.

[GRS12] Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. “Essential coding theory”. In: *Draft available at* `http://www.cse.buffalo.edu/~atri/courses/coding-theory/book` (2012).

[GS99] Venkatesan Guruswami and Madhu Sudan. “Improved Decoding of Reed-Solomon and algebraic-geometry codes”. In: *IEEE Transactions on Information Theory* 45.6 (1999), pp. 1757–1767.

[GUV09] Venkatesan Guruswami, Christopher Umans, and Salil Vadhan. “Unbalanced expanders and randomness extractors from Parvaresh-Vardy codes”. In: *Journal of the ACM (JACM)* 56.4 (2009), pp. 1–34.

[GV10] Venkatesan Guruswami and Salil Vadhan. “A lower bound on list size for list decoding”. In: *IEEE Transactions on Information Theory* 56.11 (2010), pp. 5681–5688.

[GW13] Venkatesan Guruswami and Carol Wang. “Linear-algebraic list decoding for variants of Reed-Solomon codes”. In: *IEEE Transactions on Information Theory* 59.6 (2013), pp. 3257–3268.

[GW14] Venkatesan Guruswami and Carol Wang. “Evading subspaces over large fields and explicit list-decodable rank-metric codes”. In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2014)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2014.

[GWX16] Venkatesan Guruswami, Carol Wang, and Chaoping Xing. "Explicit list-decodable rank-metric and subspace codes via subspace designs". In: *IEEE Transactions on Information Theory* 62.5 (2016), pp. 2707–2718.

[GX12] Venkatesan Guruswami and Chaoping Xing. "Folded codes from function field towers and improved optimal rate list decoding". In: *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC)*. ACM. 2012, pp. 339–350.

[GX13] Venkatesan Guruswami and Chaoping Xing. "List decoding Reed-Solomon, algebraic-geometric, and Gabidulin subcodes up to the Singleton bound". In: *Proceedings of the 45th Annual ACM Symposium on Theory of Computing (STOC)*. ACM. 2013, pp. 843–852.

[GXY18] Venkatesan Guruswami, Chaoping Xing, and Chen Yuan. "Subspace designs based on algebraic function fields". In: *Transactions of the American Mathematical Society* 370.12 (2018), pp. 8757–8775.

[Gur+02] Venkatesan Guruswami, Johan Håstad, Madhu Sudan, and David Zuckerman. "Combinatorial bounds for list decoding". In: *IEEE Transactions on Information Theory* 48.5 (2002), pp. 1021–1034.

[Gur+20] Venkatesan Guruswami, Ray Li, Jonathan Mosheiff, Nicolas Resch, Shashwat Silas, and Mary Wootters. "Bounds for list-decoding and list-recovery of random linear codes". In: *arXiv preprint arXiv:2004.13247* (2020).

[Hai+15] Iftach Haitner, Yuval Ishai, Eran Omri, and Ronen Shaltiel. "Parallel hashing via list recoverability". In: *Proceedings of the 35th Annual Cryptology Conference (Crypto)*. Springer. 2015, pp. 173–190.

[Ham50] Richard W Hamming. "Error detecting and error correcting codes". In: *The Bell System Technical Journal* 29.2 (1950), pp. 147–160.

[Har08] Aram W Harrow. "Quantum expanders from any classical Cayley graph expander". In: *Quantum Information & Computation* 8.8 (2008), pp. 715–721.

[Hay05] Thomas P Hayes. "A large-deviation inequality for vector-valued martingales". In: *Combinatorics, Probability and Computing* (2005).

[HRW17a] Brett Hemenway, Noga Ron-Zewi, and Mary Wootters. "Local List Recovery of High-rate Tensor Codes & Applications". In: *Electronic Colloquium on Computational Complexity (ECCC)* 24 (2017), p. 104.

[HRW17b] Brett Hemenway, Noga Ron-Zewi, and Mary Wootters. "Local List Recovery of High-rate Tensor Codes & Applications". In: *Electronic Colloquium on Computational Complexity (ECCC)* 24 (2017), 104 (revision 1).

[HRZW17] Brett Hemenway, Noga Ron-Zewi, and Mary Wootters. "Local list recovery of high-rate tensor codes & applications". In: *Proceedings of the 58th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2017, pp. 204–215.

[HW15] Brett Hemenway and Mary Wootters. "Linear-Time List Recovery of High-Rate Expander Codes". In: *Proceedings of the 42nd Annual International Colloquium on Automata, Languages, and Programming (ICALP)*. Springer. 2015, pp. 701–712.

[INR10] Piotr Indyk, Hung Q Ngo, and Atri Rudra. "Efficiently decodable non-adaptive group testing". In: *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. Society for Industrial and Applied Mathematics. 2010, pp. 1126–1142.

[Jac97] Jeffrey C Jackson. "An efficient membership-query algorithm for learning DNF with respect to the uniform distribution". In: *Journal of Computer and System Sciences* 55.3 (1997), pp. 414–440.

[Joh62] Selmer Johnson. "A new upper bound for error-correcting codes". In: *IRE Transactions on Information Theory* 8.3 (1962), pp. 203–207.

[Joh63] Selmer Johnson. "Improved asymptotic bounds for error-correcting codes". In: *IEEE Transactions on Information Theory* 9.3 (1963), pp. 198–205.

[KS11] Zohar S Karnin and Amir Shpilka. "Black box polynomial identity testing of generalized depth-3 arithmetic circuits with bounded top fan-in". In: *Combinatorica* 31.3 (2011), pp. 333–364.

[KT00] Jonathan Katz and Luca Trevisan. "On the efficiency of local decoding procedures for error-correcting codes". In: *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing (STOC)*. 2000, pp. 80–86.

[KS12] Kazuki Kobayashi and Tomoharu Shibuya. "Generalization of Lu's linear time encoding algorithm for LDPC codes". In: *Proceedings of the 2012 IEEE International Symposium on Information Theory and its Applications*. IEEE. 2012, pp. 16–20.

[KK08] Ralf Koetter and Frank R Kschischang. "Coding for errors and erasures in random network coding". In: *IEEE Transactions on Information Theory* 54.8 (2008), pp. 3579–3591.

[KNY17] Ilan Komargodski, Moni Naor, and Eylon Yogev. "Secret-sharing for NP". In: *Journal of Cryptology* 30.2 (2017), pp. 444–469.

[Kop15a] Swastik Kopparty. "List-decoding multiplicity codes". In: *Theory of Computing* 11.1 (2015), pp. 149–182.

[Kop15b] Swastik Kopparty. "Some remarks on multiplicity codes". In: *AMS Special Session on Discrete Geometry and Algebraic Combinatorics* 625 (2015), pp. 155–176.

[KSY14] Swastik Kopparty, Shubhangi Saraf, and Sergey Yekhanin. "High-rate codes with sublinear-time decoding". In: *Journal of the ACM (JACM)* 61.5 (2014), pp. 1–20.

[Kop+17] Swastik Kopparty, Or Meir, Noga Ron-Zewi, and Shubhangi Saraf. "High-rate locally correctable and locally testable codes with sub-polynomial query complexity". In: *Journal of the ACM (JACM)* 64.2 (2017), pp. 1–43.

[Kop+18] Swastik Kopparty, Noga Ron-Zewi, Shubhangi Saraf, and Mary Wootters. "Improved decoding of folded Reed-Solomon and multiplicity codes". In: *Proceedings of the 59th Annual IEEE symposium on Foundations of Computer Science (FOCS)*. IEEE. 2018, pp. 212–223.

[Kop+19] Swastik Kopparty, Nicolas Resch, Noga Ron-Zewi, Shubhangi Saraf, and Shashwat Silas. "On list recovery of high-rate tensor codes". In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2019)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2019.

[KRU13] Shrinivas Kudekar, Tom Richardson, and Rüdiger L Urbanke. "Spatially coupled ensembles universally achieve capacity under belief propagation". In: *IEEE Transactions on Information Theory* 59.12 (2013), pp. 7761–7813.

[KT14] Margreta Kuijper and Anna-Lena Trautmann. "List-decoding Gabidulin Codes via Interpolation and the Euclidean Algorithm". In: *Proceedings of the International Symposium on Information Theory and its Applications (ISITA)*. IEEE. 2014, pp. 343–347.

[KM93] Eyal Kushilevitz and Yishay Mansour. "Learning decision trees using the Fourier spectrum". In: *SIAM Journal on Computing* 22.6 (1993), pp. 1331–1348.

[Lar+16] Kasper Green Larsen, Jelani Nelson, Huy L Nguyên, and Mikkel Thorup. "Heavy hitters via cluster-preserving clustering". In: *Proceedings of the 57th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2016, pp. 61–70.

[LW18] Ray Li and Mary Wootters. "Improved List-Decodability of Random Linear Binary Codes". In: *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2018)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2018.

[LN97] Rudolf Lidl and Harald Niederreiter. *Finite fields*. Vol. 20. Cambridge university press, 1997.

[LM20] Nati Linial and Jonathan Mosheiff. "On the weight distribution of random binary linear codes". In: *Random Structures & Algorithms* 56.1 (2020), pp. 5–36.

[Lip90] Richard J Lipton. "Efficient checking of computations". In: *Proceedings of the 7th Annual Symposium on Theoretical Aspects of Computer Science (STACS)*. Springer. 1990, pp. 207–215.

[Loi06] Pierre Loidreau. "A Welch-Berlekamp like algorithm for decoding Gabidulin codes". In: *Coding and Cryptography*. Springer, 2006, pp. 36–45.

[Loi10] Pierre Loidreau. “Designing a rank metric based McEliece cryptosystem”. In: *International Workshop on Post-Quantum Cryptography*. Springer. 2010, pp. 142–152.

[Loi17] Pierre Loidreau. “A new rank metric codes based encryption scheme”. In: *International Workshop on Post-Quantum Cryptography*. Springer. 2017, pp. 3–17.

[LK05] Hsiao-feng Lu and P Vijay Kumar. “A unified construction of space-time codes with optimal rate-diversity tradeoff”. In: *IEEE Transactions on Information Theory* 51.5 (2005), pp. 1709–1730.

[LM09] Jin Lu and José MF Moura. “Linear time encoding of LDPC codes”. In: *IEEE Transactions on Information Theory* 56.1 (2009), pp. 233–249.

[LZ08] Alexander Lubotzky and Efim Zelmanov. “Dimension expanders”. In: *Journal of Algebra* 319.2 (2008), pp. 730–738.

[Lub+01] Michael G Luby, Michael Mitzenmacher, Mohammad Amin Shokrollahi, and Daniel A Spielman. “Efficient erasure correcting codes”. In: *IEEE Transactions on Information Theory* 47.2 (2001), pp. 569–584.

[LGB03] Paul Lusina, Ernst Gabidulin, and Martin Bossert. “Maximum rank distance codes as space-time codes”. In: *IEEE Transactions on Information Theory* 49.10 (2003), pp. 2757–2760.

[MV12] Hessam Mahdavifar and Alexander Vardy. “List-decoding of subspace codes and rank-metric codes up to Singleton bound”. In: *Proceedings of the 2012 IEEE International Symposium on Information Theory (ISIT)*. IEEE. 2012, pp. 1488–1492.

[Mos+19] Jonathan Mosheiff, Nicolas Resch, Noga Ron-Zewi, Shashwat Silas, and Mary Wootters. “LDPC Codes Achieve List Decoding Capacity”. In: *arXiv preprint arXiv:1909.06430* (2019).

[NPR12] Hung Q Ngo, Ely Porat, and Atri Rudra. “Efficiently Decodable Compressed Sensing by List-Recoverable Codes and Recursion”. In: *Proceedings of the 29th Annual International Symposium on Theoretical Aspects of Computer Science (STACS)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik. 2012.

[NX09] Harald Niederreiter and Chaoping Xing. *Algebraic geometry in coding theory and cryptography*. Princeton University Press, 2009.

[O’D14] Ryan O’Donnell. *Analysis of boolean functions*. Cambridge University Press, 2014.

[PV05] Farzad Parvaresh and Alexander Vardy. “Correcting errors beyond the Guruswami-Sudan radius in polynomial time”. In: *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. IEEE. 2005, pp. 285–294.

[PR04] Pavel Pudlák and Vojtech Rödl. “Pseudorandom sets and explicit constructions of Ramsey graphs”. In: *Submitted for publication* (2004).

[RWZ16] Netanel Raviv and Antonia Wachter-Zeh. "Some Gabidulin codes cannot be list decoded efficiently at any radius". In: *IEEE Transactions on Information Theory* 62.4 (2016), pp. 1605–1615.

[RWZ17] Netanel Raviv and Antonia Wachter-Zeh. "A correction to "Some Gabidulin codes cannot be list decoded efficiently at any radius"". In: *IEEE Transactions on Information Theory* 63.4 (2017), pp. 2623–2624.

[RS60] Irving S Reed and Gustave Solomon. "Polynomial codes over certain finite fields". In: *Journal of the Society for Industrial and Applied Mathematics* 8.2 (1960), pp. 300–304.

[Ree54] Irving Reed. "A class of multiple-error-correcting codes and the decoding scheme". In: *Transactions of the IRE Professional Group on Information Theory* 4.4 (1954), pp. 38–49.

[RVW02] Omer Reingold, Salil Vadhan, and Avi Wigderson. "Entropy waves, the zig-zag graph product, and new constant-degree expanders". In: *Annals of Mathematics* 155 (2002), pp. 157–187.

[RZWZ20] Noga Ron-Zewi, Mary Wootters, and Gilles Zémor. "Linear-time Erasure List-decoding of Expander Codes". In: *arXiv preprint arXiv:2002.08579* (2020).

[Rot91] Ron M Roth. "Maximum-rank array codes and their application to crisscross error correction". In: *IEEE Transactions on Information Theory* 37.2 (1991), pp. 328–336.

[RS96] Ronitt Rubinfeld and Madhu Sudan. "Robust characterizations of polynomials with applications to program testing". In: *SIAM Journal on Computing* 25.2 (1996), pp. 252–271.

[Rud07] Atri Rudra. *List decoding and property testing of error-correcting codes*. University of Washington, 2007.

[Rud11] Atri Rudra. "Limits to list decoding of random codes". In: *IEEE Transactions on Information Theory* 57.3 (2011), pp. 1398–1408.

[RW14] Atri Rudra and Mary Wootters. "Every list-decodable code for high noise has abundant near-optimal rate puncturings". In: *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*. ACM. 2014, pp. 764–773.

[RW18] Atri Rudra and Mary Wootters. "Average-radius list-recoverability of random linear codes". In: *Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*. SIAM. 2018, pp. 644–662.

[Sau72] Norbert Sauer. "On the density of families of sets". In: *Journal of Combinatorial Theory, Series A* 13.1 (1972), pp. 145–147.

[Sha48] Claude E. Shannon. "A mathematical theory of communication". In: *Bell System Technical Journal* 27 (1948). Monograph B-1598.

[She72] Saharon Shelah. "A combinatorial problem; stability and order for models and theories in infinitary languages". In: *Pacific Journal of Mathematics* 41.1 (1972), pp. 247–261.

[SKK08] Danilo Silva, Frank R Kschischang, and Ralf Koetter. "A rank-metric approach to error control in random network coding". In: *IEEE Transactions on Information Theory* 54.9 (2008), pp. 3951–3967.

[Sin64] Richard Singleton. "Maximum distance q-nary codes". In: *IEEE Transactions on Information Theory* 10.2 (1964), pp. 116–118.

[SS96] Michael Sipser and Daniel A Spielman. "Expander codes". In: *IEEE Transactions on Information Theory* 42.6 (1996), pp. 1710–1722.

[Sti09] Henning Stichtenoth. *Algebraic function fields and codes*. Vol. 254. Springer Science & Business Media, 2009.

[STV01] Madhu Sudan, Luca Trevisan, and Salil Vadhan. "Pseudorandom generators without the XOR lemma". In: *Journal of Computer and System Sciences* 62.2 (2001), pp. 236–266.

[Tan81] R Tanner. "A recursive approach to low complexity codes". In: *IEEE Transactions on Information Theory* 27.5 (1981), pp. 533–547.

[TV06] Terence Tao and Van H Vu. *Additive combinatorics*. Vol. 105. Cambridge University Press, 2006.

[Tho83] Christian Thommesen. "The existence of binary linear concatenated codes with Reed-Solomon outer codes which asymptotically meet the Gilbert-Varshamov bound". In: *IEEE Transactions on Information Theory* 29.6 (1983), pp. 850–853.

[TVZ82] M A Tsfasman, SG Vlădut, and Th Zink. "Modular curves, Shimura curves, and Goppa codes, better than Varshamov-Gilbert bound". In: *Mathematische Nachrichten* 109.1 (1982), pp. 21–28.

[Vad12] Salil P Vadhan. "Pseudorandomness". In: *Foundations and Trends® in Theoretical Computer Science* 7.1–3 (2012), pp. 1–336.

[Var57] RR Varshamov. "Estimate of the number of signals in error correcting codes". In: *Docklady Akad. Nauk, SSSR* 117 (1957), pp. 739–741.

[Vid15] Michael Viderman. "A combination of testability and decodability by tensor products". In: *Random Structures & Algorithms* 46.3 (2015), pp. 572–598.

[WZ12] Antonia Wachter-Zeh. "Bounds on list decoding Gabidulin codes". In: *Proceedings of the 13th Annual International Workshop on Algebraic and Combinatorial Coding Theory (ACCT)*. 2012.

[WZ13] Antonia Wachter-Zeh. "Bounds on list decoding of rank-metric codes". In: *IEEE Transactions on Information Theory* 59.11 (2013), pp. 7268–7277.

[WJ+08] Martin J Wainwright, Michael I Jordan, et al. "Graphical models, exponential families, and variational inference". In: *Foundations and Trends® in Machine Learning* 1.1–2 (2008), pp. 1–305.

[WB86] Lloyd R Welch and Elwyn R Berlekamp. *Error correction for algebraic block codes*. US Patent 4,633,470. 1986.

[Wig04] Avi Wigderson. *Expanders: Old and new applications and problems*. Lecture at the Institute for Pure and Applied Mathematics (IPAM). Feb. 2004.

[Woo13] Mary Wootters. "On the list decodability of random linear codes with large error rates". In: *Proceedings of the 45h Annual ACM Symposium on Theory of Computing (STOC)*. ACM. 2013, pp. 853–860.

[Woz58] John M Wozencraft. "List decoding". In: *Quarterly Progress Report* 48 (1958), pp. 90–95.

[Zém01] Gillés Zémor. "On expander codes". In: *IEEE Transactions on Information Theory* 47.2 (2001), pp. 835–837.

[ZP81] Victor Vasilievich Zyablov and Mark Semenovich Pinsker. "List concatenated decoding". In: *Problemy Peredachi Informatsii* 17.4 (1981), pp. 29–33.